Licence Sciences et Techniques
L2 MATH

UNIVERSITÉ DE TOULON

M33

Analyse numérique

Recueil d'exercices corrigés et aide-mémoire.

Gloria Faccanoni

http://faccanoni.univ-tln.fr/enseignements.html

Année 2015 – 2016

Dernière mise-à-jour
Jeudi 27 août 2015

Ce fascicule est un support au cours d'analyse numérique en deuxième année d'une Licence de Mathématiques. Il aborde : la recherche de racines d'une fonction réelle de variable réelle, l'interpolation polynomiale, l'intégration numériques, l'intégration d'équations différentielles et la résolution de systèmes linéaires. Les applications se font avec le langage Python dont la documentation et les sources peuvent être téléchargées à l'adresse `http://www.python.org`. Les notions supposées connues correspondent au programme des cours de Mathématiques (Analyse mathématique des fonctions réelles d'une variable réelle et Algèbre Linéaire) et Informatiques (Initiation à l'algorithmique et au langage Python) de la première année de Licence.

L'objet de ce aide-mémoire est de proposer une explication succincte des concepts vu en cours. De nombreux livres, parfois très fournis, existent. Ici on a cherché, compte tenu des contraintes de volume horaire, des acquis des étudiants à la première année et des exigences pour la suite du cursus, à dégager les points clés permettant de structurer le travail personnel de l'étudiant voire de faciliter la lecture d'autres ouvrages. Ce polycopiée ne dispense pas des séances de cours et de TD ni de prendre des notes complémentaires. Il est d'ailleurs important de comprendre et apprendre le cours au fur et à mesure. Ce polycopié est là pour éviter un travail de copie qui empêche parfois de se concentrer sur les explications données oralement mais **ce n'est pas un livre auto-suffisant (il est loin d'être exhaustif) !** De plus, ne vous étonnez pas si vous découvrez des erreurs (merci de me les communiquer).

On a inclus dans ce texte nombreux exercices corrigés. Ceux-ci, de difficulté variée, répondent à une double nécessitée. Il est important de jongler avec les différents concepts introduits en cours et même de faire certaines erreurs une fois pour bien identifier les pièges. Les exercices permettent d'orienter les raisonnements vers d'autres domaines (physique, économie, etc.), cela afin d'exhiber l'intérêt et l'omniprésence de l'analyse numérique au sens large (modélisation, analyse mathématique, discrétisation, résolution numérique et interprétation des résultats). Cependant, veuillez noter que vous n'obtiendrez pas grande chose si vous vous limitez à choisir un exercice, y réfléchir une minute et aller vite voir le début de la correction en passant tout le temps à essayer de comprendre la correction qui va paraitre incompréhensible. Pour que la méthode d'étude soit vraiment efficace, il faut d'abord vraiment essayer de chercher la solution. En particulier, il faut avoir un papier brouillon à coté de soi et un crayon. La première étape consiste alors à traduire l'énoncé (pas le recopier), en particulier s'il est constitué de beaucoup de jargon mathématique. Ensuite il faut essayer de rapprocher les hypothèses de la conclusion souhaitée, et pour cela faire quelques calculs ou transformer les hypothèses pour appliquer un théorème dont on aura vérifier que les hypothèses sont bien satisfaites. C'est ici que l'intuition joue un grand rôle et il ne faut pas hésiter à remplir des pages pour s'apercevoir que l'idée qu'on a eu n'est pas la bonne. Elle pourra toujours resservir dans une autre situation. Quand finalement on pense tenir le bon bout, il faut rédiger soigneusement en s'interrogeant à chaque pas sur la validité (logique, mathématique) de ce qu'on a écrit. Si l'étape précédente ne donne rien, il faut chercher de l'aide (voir le début de la correction, en parler à un autre étudiant, etc.).

Gloria FACCANONI

IMATH Bâtiment M-117
Université de Toulon
Avenue de l'université
83957 LA GARDE - FRANCE

☎ 0033 (0)4 83 16 66 72

✉ gloria.faccanoni@univ-tln.fr
🛈 http://faccanoni.univ-tln.fr

Table des matières

Notations

Ensembles usuels en mathématiques

On désigne généralement les ensemble les plus usuels par une lettre à double barre :

$\mathbb{N}$ l'ensemble des entiers naturels

$\mathbb{N}^*$ l'ensemble des entiers strictement positifs

$\mathbb{Z}$ l'ensemble des entiers relatifs (positifs, négatifs ou nuls)

$\mathbb{Z}^*$ l'ensemble des entiers $\neq 0$

$\mathbb{Q}$ l'ensemble des nombres rationnels $\left(\frac{p}{q}, \quad p \in \mathbb{Z}, \quad q \in \mathbb{Z}^*\right)$

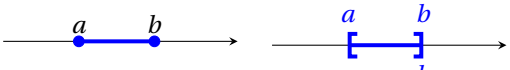
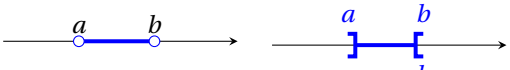
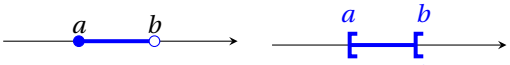
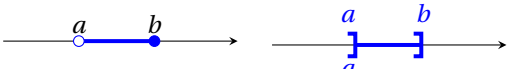
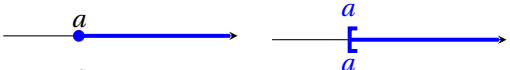
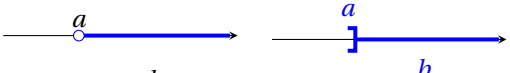
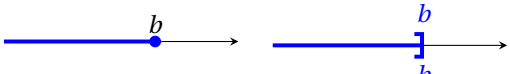
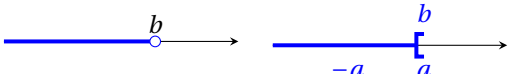
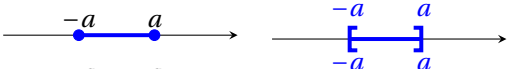
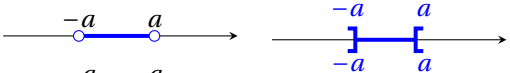
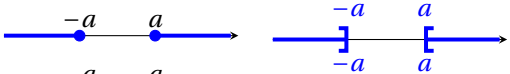
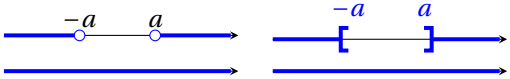
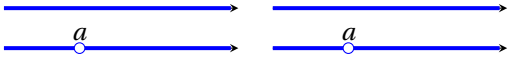
$\mathbb{R}$ l'ensemble des réels

$\mathbb{R}^*$ l'ensemble des réels autres que 0

$\mathbb{C}$ l'ensemble des nombres complexes

$\mathbb{R}_n[x]$ l'espace vectoriel des polynômes en x de degré inférieur ou égal à n

Intervalles

Inégalité	Notation ensembliste	Représentations graphique
$a \leq x \leq b$	$[a, b]$	
$a < x < b$	$]a, b[$	
$a \leq x < b$	$[a, b[$	
$a < x \leq b$	$]a, b]$	
$x \geq a$	$[a, +\infty[$	
$x > a$	$]a, +\infty[$	
$x \leq b$	$]-\infty, b]$	
$x < b$	$]-\infty, b[$	
$\lvert x\rvert \leq a$ avec $a \geq 0$	$[-a, a]$	
$\lvert x\rvert < a$ avec $a \geq 0$	$]-a, a[$	
$\lvert x\rvert \geq a$ avec $a \geq 0$	$]-\infty, -a] \cup [a, +\infty[$	
$\lvert x\rvert > a$ avec $a \geq 0$	$]-\infty, -a[\cup]a, +\infty[$	
$\forall x \in \mathbb{R}$	$]-\infty, +\infty[$	
$x \neq a$	$]-\infty, a[\cup]a, +\infty[= \mathbb{R} \setminus \{a\}$	

Symboles utilisés dans le document

définition

théorème, corollaire, proposition

propriété(s)

astuce

attention

remarque

 méthode, algorithme, cas particulier

 exercice de base

 exercice

 exemple

 curiosité

$\equiv$	égal par définition
$>$	strictement supérieur
$<$	strictement inférieur
$\geq$	supérieur ou égal
$\leq$	inférieur ou égal
$\neq$	différent
$\{ \quad \}$	ensemble
$\mathbb{A} \setminus \mathbb{B}$	ensemble $\mathbb{A}$ privé de l'ensemble $\mathbb{B}$, *i.e.* $C_{\mathbb{A}}(\mathbb{B})$ le complémentaire de $\mathbb{B}$ dans $\mathbb{A}$
$\varnothing$	ensemble vide
$\mid$	tel que
$\in$	appartient
$\notin$	n'appartient pas
$\forall$	pour tout (quantificateur universel)
$\exists$	il existe (quantificateur universel)
$\nexists$	il n'existe pas
$\exists!$	il existe un et un seul
$\subset$	est sous-ensemble (est contenu)
$\cup$	union d'ensembles
$\cap$	intersection d'ensembles
$\Longrightarrow$	si ... alors
$\Longleftrightarrow$	si et seulement si
ssi	si et seulement si
ln	logarithme de base e
$\log_a$	logarithme de base a
∞	infini
$\int$	symbole d'intégrale
$\sum_{i=0}^{n} a_i$	somme par rapport à l'indice i, équivaut à $a_0 + a_1 + \cdots + a_n$
$\prod_{i=0}^{n} a_i$	produit par rapport à l'indice i, équivaut à $a_0 \times a_1 \times \cdots \times a_n$
$n!$	n factoriel, équivaut à $1 \times 2 \times \cdots \times n$
$g \circ f$	f puis g
f', $\frac{df}{dx}$	symboles de dérivée
$\mathscr{D}_f$	domaine de définition d'une fonction f

Conventions pour la présentation du code

Pour l'écriture du code, on utilise deux présentations :

① les instructions précédées de chevrons dans une boite grisée sont à saisir dans une session interactive

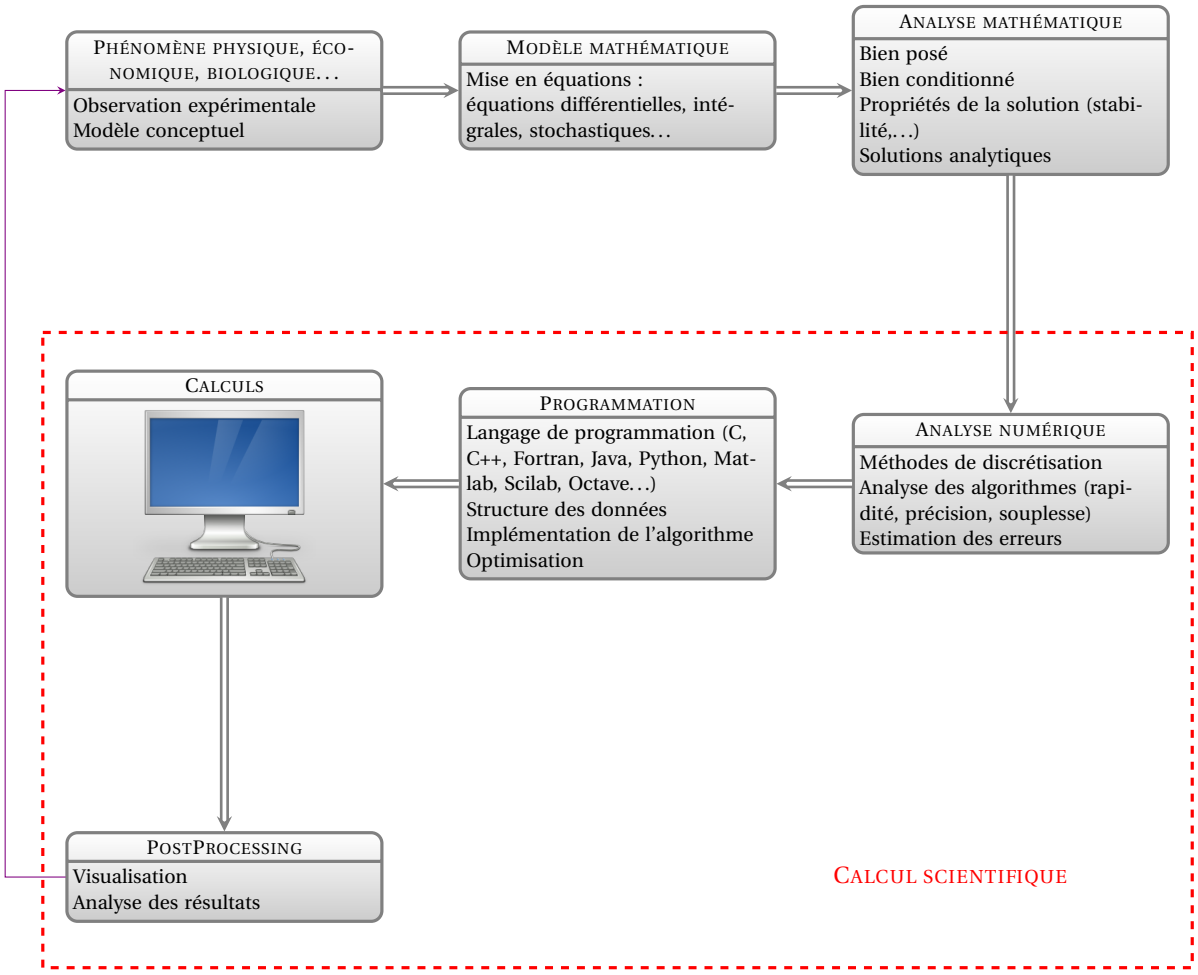
```
>>> 1 + 1
2
```

② les instructions sans chevrons dans une boite grisée sont des bouts de code à écrire dans un fichier

```
print('Coucou!')
```

Introduction au calcul scientifique

On peut définir le CALCUL SCIENTIFIQUE comme la discipline qui permet de reproduire sur un ordinateur un phénomène ou un processus décrit par un modèle mathématique.



L'ordinateur est aujourd'hui un outil incontournable pour simuler et modéliser des systèmes complexes, mais il faut encore savoir exprimer nos problèmes (physiques, économiques, biologiques...) en langage formalisé des mathématiques pures sous la forme d'équations mathématiques (différentielles, intégrales...). Nous sommes habitués à résoudre les problèmes de façon analytique, alors que l'ordinateur ne travaille que sur des suites de nombres. On verra qu'il existe souvent plusieurs approches pour résoudre un même problème, ce qui conduit à des algorithmes différents. Un des objectifs de ce cours est de fournir des bases rigoureuses pour développer quelques algorithmes utiles dans la résolution de problèmes en mathématique, économie, physique...

Un algorithme, pour être utile, doit satisfaire un certain nombre de conditions. Il doit être :

rapide : le nombre d'opérations de calcul pour arriver au résultat escompté doit être aussi réduit que possible ;

précis : l'algorithme doit savoir contenir les effets des erreurs qui sont inhérentes à tout calcul numérique (ces erreurs peuvent être dues à la modélisation, aux données, à la représentation sur ordinateur ou encore à la troncature) ;

souple : l'algorithme doit être facilement transposable à des problèmes différents.

Le choix et l'optimisation des algorithmes numériques mis en pratique sont absolument cruciaux tant pour les calculs de type industriel souvent très répétitifs et devant donc pouvoir être exécutés en un temps très court, que pour les calculs de référence pour lesquels la seule limite est la patience de celui qui les fait. Par exemple, en fluidodynamique, en laissant tourner une station de travail pendant quelques jours, les numériciens résolvent des systèmes frisant le milliard d'inconnues. L'expérience montre qu'entre une approche numérique standard et une approche soigneusement réfléchie

et optimisée un gain de temps de calcul d'un facteur 100, voire davantage, est souvent observé. Il est clair qu'on peut passer ainsi, grâce à cet effort, d'un calcul totalement déraisonnable à un calcul parfaitement banal : tout l'enjeu de l'analyse numériques est là ! C'est dire l'importance pour tous scientifique de bien connaître ces méthodes, leurs avantages et leurs limites.

Exemple *Calcul de* $\sqrt{A}$

Sur ordinateur, l'addition de deux entiers peut se faire de façon exacte mais non le calcul d'une racine carrée. On procède alors par approximations successives jusqu'à converger vers la solution souhaitée. Il existe pour cela divers algorithmes. Le suivant est connu depuis l'antiquité (mais ce n'est pas celui que les ordinateurs utilisent).

Soit A un nombre réel positif dont on cherche la racine carrée. Désignons par x_0 la première estimation de cette racine (généralement le plus grand entier dont le carré est inférieur à A ; par exemple, si $A = 178$, alors $x_0 = 13$ car $13^2 = 169 < 178$ et $14^2 = 196 > 178$) et par ε_0 l'erreur associée :

$$\sqrt{A} = x_0 + \varepsilon_0.$$

Cherchons une approximation de ε_0. On a

$$A = (x_0 + \varepsilon_0)^2 = x_0^2 + 2x_0\varepsilon_0 + \varepsilon_0^2.$$

Supposons que l'erreur soit petite face à x_0, ce qui permet de négliger le terme en ε_0^2 :

$$A \simeq x_0^2 + 2x_0\varepsilon_0.$$

Remplaçons l'erreur ε_0 par un ε_0', qui en est une approximation, de telle sorte que

$$A = x_0^2 + 2x_0\varepsilon_0'.$$

On en déduit que

$$\varepsilon_0' = \frac{A - x_0^2}{2x_0}$$

donc la quantité

$$x_1 \equiv x_0 + \varepsilon_0' = \frac{1}{2}\left(\frac{A}{x_0} + x_0\right)$$

constitue une meilleure approximation de la racine que x_0 (sous réserve que le développement soit convergent). De plus, rien ne nous empêche de recommencer les calculs avec x_1, puis x_2, etc., jusqu'à ce que la précision de la machine ne permette plus de distinguer le résultat final de la véritable solution. On peut donc définir une suite, qui à partir d'une estimation initiale x_0 devrait en principe converger vers la solution recherchée. Cette suite est

$$x_{k+1} = \frac{1}{2}\left(\frac{A}{x_k} + x_k\right), \qquad x_0 > 0.$$

L'algorithme du calcul de la racine carrée devient donc

1. Démarrer avec une première approximation $x_0 > 0$ de $\sqrt{A}$.
2. À chaque itération k, calculer la nouvelle approximation $x_{k+1} = \frac{1}{2}\left(\frac{A}{x_k} + x_k\right)$.
3. Calculer l'erreur associée $\varepsilon_{k+1}' = \frac{A - x_{k+1}^2}{2x_{k+1}}$.
4. Tant que l'erreur est supérieure à un seuil fixé, recommencer au point 2

Le tableau ci-dessous illustre quelques itérations de cet algorithme pour le cas où $A = 5$:

k	x_k	ε_k'
0	2.0000000000	0.2360679775
1	2.2500000000	0.0139320225
2	2.2361111111	0.0000431336
3	2.2360679779	0.0000000004
4	2.2360679775	0.0000000000

On voit que l'algorithme converge très rapidement et permet donc d'estimer la racine carrée d'un nombre moyennant un nombre limité d'opérations élémentaires (additions, soustractions, divisions, multiplications). Il reste encore à savoir si cet algorithme converge toujours et à déterminer la rapidité de sa convergence. L'analyse numérique est une discipline proche des mathématiques appliquées, qui a pour objectif de répondre à ces questions de façon rigoureuse.

Les erreurs

Le simple fait d'utiliser un ordinateur pour représenter des nombres réels induit des erreurs. Par conséquent, plutôt que de tenter d'éliminer les erreurs, il vaut mieux chercher à contrôler leur effet. Généralement, on peut identifier plusieurs niveaux d'erreur dans l'approximation et la résolution d'un problème physique.

Au niveau le plus élevé, on trouve l'erreur qui provient du fait qu'on a réduit la réalité physique à un modèle mathématique. De telles erreurs limitent l'application du modèle mathématique à certaines situations et ne sont pas dans le champ du contrôle du Calcul Scientifique.

On ne peut généralement pas donner la solution explicite d'un modèle mathématique (qu'il soit exprimé par une intégrale, une équation algébrique ou différentielle, un système linéaire ou non linéaire). La résolution par des algorithmes numériques entraîne immanquablement l'introduction et la propagation d'erreurs d'arrondi. De plus, il est souvent nécessaire d'introduire d'autres erreurs liées au fait qu'un ordinateur ne peut effectuer que de manière approximative des calculs impliquant un nombre infini d'opérations arithmétiques. Par exemple, le calcul de la somme d'une série ne pourra être accompli qu'en procédant à une troncature convenable. On doit donc définir un problème numérique, dont la solution diffère de la solution mathématique exacte d'une erreur, appelée erreur de troncature. La somme des erreurs d'arrondis et de troncature constitue l'erreur de calcul. L'erreur de calcul absolue est la différence entre x, la solution exacte du modèle mathématique, et $\tilde{x}$, la solution obtenue à la fin de la résolution numérique, tandis que (si $x \neq 0$) l'erreur de calcul relative est définie par l'erreur de calcul absolue divisé par x. Le calcul numérique consiste généralement à approcher le modèle mathématique en faisant intervenir un paramètre de discrétisation, que nous noterons h et que nous supposerons positif. Si, quand h tend vers 0, la solution du calcul numérique tend vers celle du modèle mathématique, nous dirons que le calcul numérique est convergent. Si de plus, l'erreur (absolue ou relative) peut être majorée par une fonction de Ch^p où C est indépendante de h et où p est un nombre positif, nous dirons que la méthode est convergente d'ordre p. Quand, en plus d'un majorant, on dispose d'un minorant $C_1 h^p$ (C_1 étant une autre constante ($\leq C$) indépendante de h et p), on peut remplacer le symbole $\leq$ par $\simeq$.

Remarque *Pourquoi ne pas faire toujours confiance à une calculatrice*

Voici un exemple d'un calcul pour lequel une calculatrice donne un résultat faux mais qu'un élève sait faire (en un temps raisonnable).

$$\text{Calculer } \sin((1+\sqrt{2})^{200}\pi).$$

Une calculatrice (par exemple celle de Google) renvoie un résultat bidon, alors que n'importe quel élève compétent voit que $(1+\sqrt{2})^{200}+(1-\sqrt{2})^{200}$ est un entier pair, si bien que $\sin((1+\sqrt{2})^{200}\pi) = -\sin((1-\sqrt{2})^{200}\pi)$ est extrêmement petit, et on peut même en faire une estimation de tête : $\sin((1-\sqrt{2})^{200}\pi) < ((1-\sqrt{2})^{200}\pi) < \frac{1}{2^{200}}\pi$, or $2^{10} = 1024 \geq 1000 = 10^3$ donc $\frac{1}{2^{20\times 10}}\pi \leq 10^{-60}\pi$; de tête on peut dire que $\sin((1+\sqrt{2})^{200}\pi) \in]0; -4\times 10^{-60}[$. La valeur correcte est environ -8.75×10^{-77}, donc cette estimation n'est pas terrible, mais c'est tout de même mieux que les 0.97 retournés par Google.

Source : `http://www.madore.org/~david/weblog/2014-06.html#d.2014-06-04.2205`

1. Résolution d'équations non linéaires

Recherche des solutions de l'équation non linéaire $f(x) = 0$ où f est une fonction donnée

Un des problèmes classiques en mathématiques appliquées est celui de la recherche des valeurs pour lesquelles une fonction donnée s'annule. Dans certains cas bien particuliers, comme pour les fonctions $x \mapsto x+1$, $x \mapsto \cos(2x)$ ou encore $x \mapsto x^2 - 2x + 1$, le problème est simple car il existe pour ces fonctions des formules qui donnent les zéros explicitement. Toutefois, pour la plupart des fonctions $f : \mathbb{R} \to \mathbb{R}$ il n'est pas possible de résoudre l'équation $f(x) = 0$ explicitement et il faut recourir à des méthodes numériques. Ainsi par exemple une brève étude de la fonction $f(x) = \cos(x) - x$ montre qu'elle possède un zéro à proximité de 0.7 mais ce zéro ne s'exprime pas au moyen de fonctions usuelles et pour en obtenir une valeur approchée il faut recourir à des méthodes numériques.

Plusieurs méthodes existent et elles différent pas leur vitesse de convergence et par leur robustesse. Lorsqu'il s'agit de calculer les zéros d'une seule fonction, la vitesse de la méthode utilisée n'est souvent pas cruciale. Cependant, dans certains applications il est nécessaire de calculer les zéros de plusieurs milliers de fonctions et la vitesse devient alors un élément stratégique. Par exemple, si on veut représenter graphiquement l'ensemble de points (x, y) du plan pour lequel $x^2 \sin(y) + e^{x+y} - 7 = 0$, on peut procéder comme suit : pour une valeur donnée de x, on cherche l'ensemble des valeurs de y pour lesquelles $x^2 \sin(y) + e^{x+y} - 7 = 0$, *i.e.* l'ensemble des zéros de la fonction $f(y) = x^2 \sin(y) + e^{x+y} - 7 = 0$, et on représente tous les couples obtenus. On choisit une nouvelle valeur pour x et on répète l'opération. Pour obtenir une courbe suffisamment précise, il faut procéder à la recherche des zéros d'un très grand nombre de fonction et il est alors préférable de disposer d'une méthode rapide.

Soit $f : \mathbb{R} \to \mathbb{R}$ une fonction continue donnée dont on veut évaluer numériquement un ou plusieurs zéros $\widehat{x}$, c'est-à-dire qu'on cherche tous les $\widehat{x}$ tels que $f(\widehat{x}) = 0$. Les méthodes numériques pour approcher $\widehat{x}$ consistent à :

① localiser grossièrement le (ou les) zéro(s) de f en procédant à l'étude du graphe de f et/ou à des évaluations qui sont souvent de type graphique ; on note x_0 cette solution grossière ;

② construire, à partir de x_0, une suite $x_1, x_2, x_3, \ldots$ telle que $\lim_{k\to\infty} x_k = \widehat{x}$ où $f(\widehat{x}) = 0$. On dit alors que la méthode est convergente.

Définition *Méthode itérative à deux niveaux*

On appelle *méthode itérative à deux niveaux* un procédé de calcul de la forme

$$x_{k+1} = G(x_k), \qquad k = 0, 1, 2, \ldots$$

dans lequel on part d'une valeur donnée x_0 pour calculer x_1, puis à l'aide de x_1 on calcul x_2 etc. La formule même est dite formule de récurrence. Le procédé est appelé *convergent* si x_k tend vers un nombre fini lorsque k tend vers $+\infty$. Il est bien évident qu'une méthode itérative n'est utile que s'il y a convergence vers les valeurs cherchées.

On peut parfaitement envisager des méthodes itératives multi-niveaux, comme par exemples les schémas à trois niveaux dans lesquels on part de deux valeurs données x_0 et x_1 pour calculer x_2, puis à l'aide de x_1 et x_2 on calcule x_3 etc.

Définition *Ordre de convergence*

Soit p un entier positif. On dit qu'une méthode (à deux niveaux) convergente est d'ordre p s'il existe une constante C telle que

$$|\widehat{x} - x_{k+1}| \leq C|\widehat{x} - x_k|^p.$$

ou encore

$$\lim_{k\to\infty} \frac{x_{k+1} - \widehat{x}}{(x_k - \widehat{x})^p} = C.$$

Si $p = 1$ (et $C < 1$) on parle de convergence linéaire, si $p = 2$ on parle de convergence quadratique.

1.1. Étape ① : localisation des zéros

Définition

Soit $f : \mathbb{R} \to \mathbb{R}$ une fonction. On dit que $\widehat{x}$ est un zéro de f si $f(\widehat{x}) = 0$. Il est dit *simple* si $f'(\widehat{x}) \neq 0$, *multiple* sinon.
Si f est de classe $\mathscr{C}^p$ avec $p \in \mathbb{N}$, on dit que $\widehat{x}$ est un zéro de multiplicité p si

$$\begin{cases} f^{(i)}(\widehat{x}) = 0 & \forall i = 0, \dots, p-1 \\ f^{(p)}(\widehat{x}) \neq 0. \end{cases}$$

Pour localiser grossièrement le (ou les) zéro(s) de f on va d'abord étudier de la fonction f, puis on va essayer d'utiliser un corollaire du théorème des valeurs intermédiaires et le théorème de la bijection afin de trouver un intervalle qui contient un et un seul zéro.

Théorème *des valeurs intermédiaires*

Soit f une fonction continue sur un intervalle $I = [a; b]$ de $\mathbb{R}$. Alors f atteint toutes les valeurs intermédiaires entre $f(a)$ et $f(b)$. Autrement dit :
- ⋆ si $f(a) \leq f(b)$ alors pour tout $d \in [f(a), f(b)]$ il existe $c \in [a; b]$ tel que $f(c) = d$;
- ⋆ si $f(a) \geq f(b)$ alors pour tout $d \in [f(b), f(a)]$ il existe $c \in [a; b]$ tel que $f(c) = d$.

Ce théorème donne alors le corollaire immédiat suivant.

Corollaire *des zéros d'une fonction continue*

Soit une fonction continue $f : [a, b] \to \mathbb{R}$. Si $f(a) \cdot f(b) < 0$, alors il existe (au moins un) $\widehat{x} \in]a, b[$ tel que $f(\widehat{x}) = 0$.

Ce théorème garantit juste l'existence d'un zéro. Pour l'unicité on essayera d'appliquer le théorème de la bijection dont l'énoncé est rappelé ci-dessous.

Théorème *de la bijection*

Soit f une fonction continue et strictement monotone sur un intervalle I de $\mathbb{R}$, alors f induit une bijection de I dans $f(I)$. De plus, sa bijection réciproque est continue sur I, monotone sur I et de même sens de variation que f.

1.2. Étape ② : construction d'une suite convergente

Ayant encadré les zéros de f, la construction de suites qui convergent vers ces zéros peut se faire à l'aide de plusieurs méthodes numériques. Ci-dessous on décrit les méthodes les plus connues et on étudie leurs propriétés (convergence locale *vs* globale, vitesse de convergence, etc.).

1.2.1. Méthodes de dichotomie (ou bissection), de Lagrange (ou *Regula falsi*)

Dans les méthodes de dichotomie et de LAGRANGE, à chaque pas d'itération on divise en deux un intervalle donné et on choisit le sous-intervalle où f change de signe.

Concrètement, soit $f : [a; b] \to \mathbb{R}$ une fonction strictement monotone sur un intervalle $[a, b]$. On suppose que l'équation $f(x) = 0$ n'a qu'une et une seule solution dans cet intervalle. On se propose de déterminer cette valeur avec une précision donnée. Soit $[a_0, b_0]$ un intervalle dans lequel $f(a_0)f(b_0) < 0$ et soit $c_0 \in]a_0, b_0[$. Si $f(a_0)f(c_0) < 0$, alors la racine appartient à l'intervalle $[a_0, c_0]$ et on reprend le procédé avec $a_1 = a_0$ et $b_1 = c_0$. Sinon, c'est-à-dire si $f(a_0)f(c_0) \geq 0$ on pose $a_1 = c_0$ et $b_1 = b_0$. On construit ainsi une suite d'intervalles emboîtés $[a_k, b_k]$. Les suites a_k et b_k sont adjacentes [1] et convergent vers $\widehat{x}$.

Définition *Méthodes de dichotomie et de* LAGRANGE

Soit deux points a_0 et b_0 (avec $a_0 < b_0$) d'images par f de signe contraire (*i.e.* $f(a_0) \cdot f(b_0) < 0$). En partant de $I_0 = [a_0, b_0]$, les méthodes *de dichotomie* et *de* LAGRANGE produisent une suite de sous-intervalles $I_k = [a_k, b_k]$, $k \geq 0$, avec $I_k \subset I_{k-1}$ pour $k \geq 1$ et tels que $f(a_k) \cdot f(b_k) < 0$.
- ⋆ Dans la *méthode de dichotomie*, on découpe l'intervalle $[a_k; b_k]$ en deux intervalles de même longueur, *i.e.* on divise

1. Deux suites $(u_n)_{n\in\mathbb{N}}$ et $(v_n)_{n\in\mathbb{N}}$ sont adjacentes si
   - ⋆ (u_n) est croissante,
   - ⋆ (v_n) est décroissante,
   - ⋆ $\lim_n (u_n - v_n) = 0$.

Si deux suites sont adjacentes, elles convergent et ont la même limite.

$[a_k;b_k]$ en $[a_k;c_k]$ et $[c_k;b_k]$ où c_k est

$$c_k = \frac{a_k+b_k}{2}.$$

⋆ Dans la *méthode de* LAGRANGE, plutôt que de diviser l'intervalle $[a_k;b_k]$ en deux intervalles de même longueur, on découpe $[a_k;b_k]$ en $[a_k;c_k]$ et $[c_k;b_k]$ où c_k est l'abscisse du point d'intersection de la droite passant par $(a_k, f(a_k))$ et $(b_k, f(b_k))$ et l'axe des abscisses, autrement dit c_k est solution de l'équation

$$\frac{f(b_k)-f(a_k)}{b_k-a_k}(c-b_k)+f(b_k)=0$$

qui est

$$c_k = b_k - \frac{b_k-a_k}{f(b_k)-f(a_k)}f(b_k) = \frac{a_kf(b_k)-b_kf(a_k)}{f(b_k)-f(a_k)}.$$

Dans les deux cas, pour l'itération suivante, on pose soit $[a_{k+1};b_{k+1}]=[a_k;c_k]$ soit $[a_{k+1};b_{k+1}]=[c_k;b_k]$ de sorte à ce que $f(a_{k+1})\cdot f(b_{k+1})<0$. La suite $(c_k)_{k\in\mathbb{N}}$ converge vers $\widehat{x}$ puisque la longueur de ces intervalles tend vers 0 quand k tend vers $+\infty$.
Soit ε l'erreur maximale qu'on peut commettre, les algorithmes s'écrivent alors comme suit :

DICHOTOMIE :

Require: $a, b > a, \varepsilon, f : [a,b] \to \mathbb{R}$
  $k \leftarrow 0$
  $a_k \leftarrow a$
  $b_k \leftarrow b$
  $x_k \leftarrow \dfrac{a_k+b_k}{2}$
  while $b_k - a_k > \varepsilon$ **or** $|f(x_k)| > \varepsilon$ **do**
    if $f(a_k)f(x_k) < 0$ **then**
      $a_{k+1} \leftarrow a_k$
      $b_{k+1} \leftarrow x_k$
    else
      $a_{k+1} \leftarrow x_k$
      $b_{k+1} \leftarrow b_k$
    end if
    $x_{k+1} \leftarrow \dfrac{a_{k+1}+b_{k+1}}{2}$
    $k \leftarrow k+1$
  end while

LAGRANGE :

Require: $a, b > a, \varepsilon, f : [a,b] \to \mathbb{R}$
  $k \leftarrow 0$
  $a_k \leftarrow a$
  $b_k \leftarrow b$
  $x_k \leftarrow a_k - \dfrac{b_k-a_k}{f(b_k)-f(a_k)}f(a_k)$
  while $b_k - a_k > \varepsilon$ **or** $|f(x_k)| > \varepsilon$ **do**
    if $f(a_k)f(x_k) < 0$ **then**
      $a_{k+1} \leftarrow a_k$
      $b_{k+1} \leftarrow x_k$
    else
      $a_{k+1} \leftarrow x_k$
      $b_{k+1} \leftarrow b_k$
    end if
    $x_{k+1} \leftarrow a_{k+1} - \dfrac{b_{k+1}-a_{k+1}}{f(b_{k+1})-f(a_{k+1})}f(a_{k+1})$
    $k \leftarrow k+1$
  end while

On n'est pas obligé de stoker tous les intervalles et les itérées, on peut gagner de la mémoire en les écrasant à chaque étape :

DICHOTOMIE :

Require: $a, b > a, \varepsilon, f : [a,b] \to \mathbb{R}$
  $x \leftarrow \dfrac{a+b}{2}$
  while $b - a > \varepsilon$ **or** $|f(x)| > \varepsilon$ **do**
    if $f(a)f(x) < 0$ **then**
      $b \leftarrow x$
    else
      $a \leftarrow x$
    end if
    $x \leftarrow \dfrac{a+b}{2}$
  end while

LAGRANGE :

Require: $a, b > a, \varepsilon, f : [a,b] \to \mathbb{R}$
  $x \leftarrow a - \dfrac{b-a}{f(b)-f(a)}f(a)$
  while $b - a > \varepsilon$ **or** $|f(x)| > \varepsilon$ **do**
    if $f(a)f(x) < 0$ **then**
      $b \leftarrow x$
    else
      $a \leftarrow x$
    end if
    $x \leftarrow a - \dfrac{b-a}{f(b)-f(a)}f(a)$
  end while

Remarque

Avec la méthode de la dichotomie, les itérations s'achèvent à la m-ème étape quand $|x_m - \widehat{x}| \leq |I_m| < \varepsilon$, où ε est une tolérance fixée et $|I_m|$ désigne la longueur de l'intervalle I_m. Clairement $I_k = \frac{b-a}{2^k}$, donc pour avoir une erreur $|x_m - \widehat{x}| < \varepsilon$,

on doit prendre le plus petit m qui vérifie

$$m \geq \log_2\left(\frac{b-a}{\varepsilon}\right) = \frac{\ln(b-a)-\ln(2)}{\ln(2)}.$$

Notons que cette inégalité est générale : elle ne dépend pas du choix de la fonction f.

Exemple *Fond d'investissement*

Un compte d'épargne donne un taux $T \in [0;1]$ d'intérêt par an avec un virement annuel des intérêts sur le compte. Cela signifie que si le premier janvier 2014 on met v euros sur ce compte, à la fin de la n-ème année (*i.e.* au 31 décembre $2014+n$) on en retire $v(1+T)^n$ euros. On décide alors d'ajouter au début de chaque année encore v euros. Cela signifie que si on verse ces v euros le premier janvier $2014+m$ avec $0<m<n$, au 31 décembre $2014+n$ ajoutent $v(1+T)^{n-m}$ euros. Si à la fin de la n-ème année on en retire un capital de $M>v$ euros, quel est le taux d'intérêt annuel de cet investissement ?
À la fin de la n-ème année, le capital versé au premier janvier 2014 est devenu $v(1+T\%)^n$, celui versé au premier janvier 2015 est devenu $v(1+T\%)^{n-1}$ … par conséquente, le capital final M est relié au taux d'intérêt annuel T par la relation

$$M = v\sum_{k=1}^{n}(1+T)^k = v\frac{(1+T)^n-1}{(1+T)-1} = v\frac{1+T}{T}\left((1+T)^n-1\right).$$

On en déduit que T est racine de l'équation algébrique non linéaire $f(T)=0$ où

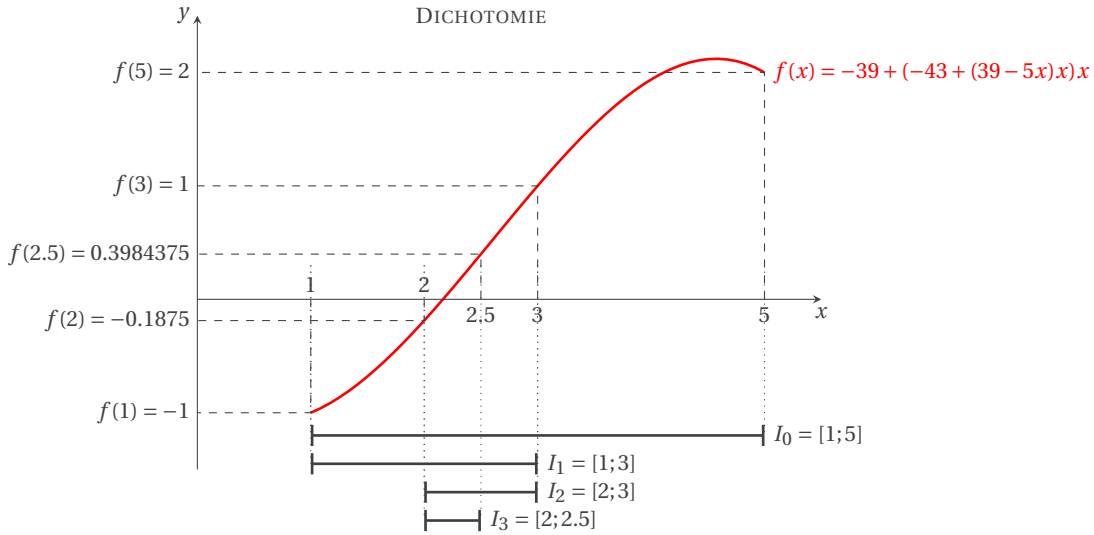
$$f(T) = v\frac{1+T}{T}\left((1+T)^n-1\right) - M.$$

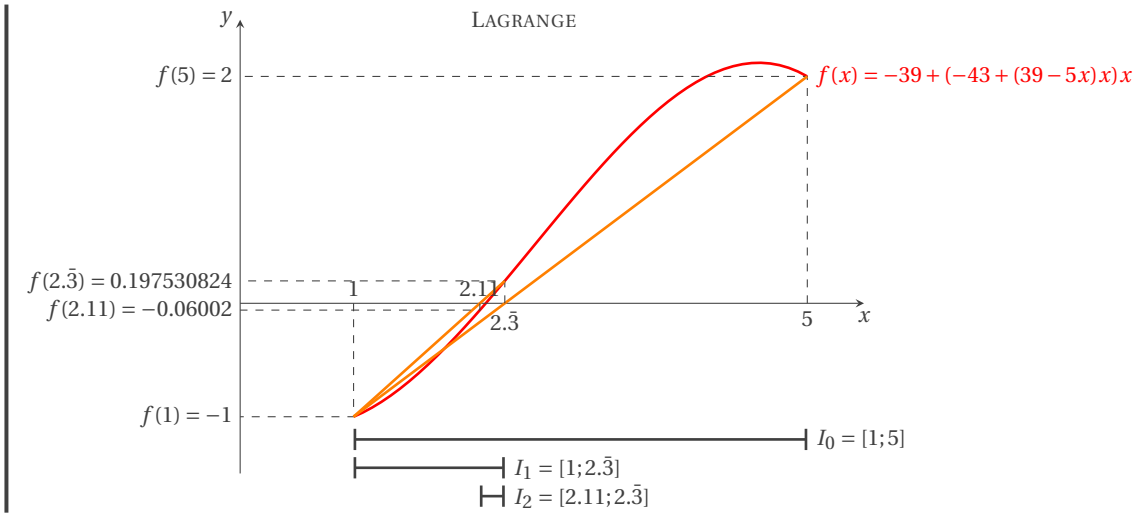
Étudions la fonction f :
⋆ $f(T)>0$ pour tout $T>0$,
⋆ $\lim_{T\to 0^+} f(T) = nv - M < (n-1)v$, $\lim_{T\to+\infty} f(T) = +\infty$,
⋆ $f'(T) = \frac{v}{T^2}\left(1+(1+T)^n(Tn-1)\right) > 0$ pour tout $T>0$ (comparer le graphe de $-1/(1+T)^n$ et de $nT-1$)
En étudiant la fonction f on voit que, comme $nv<M$ dès que $n>1$, elle admet un unique zéro dans l'intervalle $]0,+\infty[$ (on peut même prouver qu'elle admet un unique zéro dans l'intervalle $]0,M[$).
Supposons que $v=1000$€ et qu'après 5 ans M est égal à 6000€. En étudiant la fonction f on voit qu'elle admet un unique zéro dans l'intervalle $]0.01,0.1[$. Si on applique la méthode de la dichotomie avec $\varepsilon = 10^{-12}$, après 37 itérations de la méthode de dichotomie, la méthode converge vers 0.061402411536. On conclut ainsi que le taux d'intérêt T est approximativement égal à 6.14%. Ce résultat a été obtenu en faisant appel à la fonction `dichotomie` définie à la page 24 comme suit :

```
a = 0.01 # T=1%
b = 0.1 # T=10%
tol = 1.0e-12
maxITER = 50
def f(x):
    return 1000.*(1+x)/x*((1+x)**5-1.)-6000.
print dichotomie(f,a,b,tol,maxITER)
```

Exemple

Soit $f(x) = -39-43x+39x^2-5x^3$. On cherche a estimer $x\in[1;5]$ tel que $f(x)=0$.





La méthode de dichotomie est simple mais elle ne garantit pas une réduction monotone de l'erreur d'une itération à l'autre : tout ce dont on est assuré, c'est que la longueur de l'intervalle de recherche est divisée par deux à chaque étape. Par conséquent, si le seul critère d'arrêt est le contrôle de la longueur de I_k, on risque de rejeter de bonnes approximations de $\widehat{x}$. En fait, cette méthode ne prend pas suffisamment en compte le comportement réel de f. Il est par exemple frappant que la méthode ne converge pas en une seule itération quand f est linéaire (à moins que le zéro $\widehat{x}$ ne soit le milieu de l'intervalle de recherche initial).

1.2.2. Méthode de la sécante

Soit $f\colon \mathbb{R} \to \mathbb{R}$ une fonction continue et soit $\widehat{x} \in [a,b]$ un zéro de f.

La méthode de la sécante est une variante de la méthode de LAGRANGE dans laquelle on ne demande plus à ce que le zéro soit entre a_k et b_k. Pour calculer x_{k+1} on prend l'intersection de l'axe des abscisses avec la droite passant par les points $(x_k, f(x_k))$ et $(x_{k-1}, f(x_{k-1}))$, *i.e.* on cherche x solution du système linéaire

$$\begin{cases} y = \frac{f(x_k)-f(x_{k-1})}{x_k - x_{k-1}}(x - x_k) + f(x_k), \\ y = 0, \end{cases}$$

ce qui donne

$$x = x_k - \frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})} f(x_k).$$

Définition *Méthode de la Sécante*

Il s'agit d'une méthode à trois niveaux : approcher les zéros de f se ramène à calculer la limite de la la suite récurrente

$$\begin{cases} x_0 \text{ donné}, \\ x_1 \text{ donné}, \\ x_{k+1} = x_k - \frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})} f(x_k), \end{cases}$$

Cette méthode a ordre de convergence $\frac{1+\sqrt{5}}{2}$.

1.2.3. Méthodes de point fixe

En s'amusant avec une calculatrice de poche ou avec le code python ci-dessous

```
1 import math
2 x = 1
3 for i in range (1,100):
4 ⟶x = math.cos(x)
5 ⟶print("x[{0}]={1}".format(i,x))
```

on peut vérifier qu'en partant de la valeur 1 et en appuyant plusieurs fois de suite sur la touche «cosinus», on obtient cette suite de valeurs :

$$x_0 = 1,$$

$$
\begin{aligned}
x_1 &= \cos(x_0) = 0.540302305868,\\
x_2 &= \cos(x_1) = 0.857553215846,\\
x_3 &= \cos(x_2) = 0.654289790498,\\
&\vdots\\
x_{55} &= 0.739085133171,\\
&\vdots\\
x_{100} &= 0.739085133215
\end{aligned}
$$

qui tend vers la valeur 0.73908513… . En effet, on a par construction $x_{k+1} = \cos(x_k)$ pour $k = 0, 1, \ldots$ (avec $x_0 = 1$). Si cette suite converge, sa limite ℓ satisfait l'équation $\cos(\ell) = \ell$. Pour cette raison, ℓ est appelé point fixe de la fonction cosinus.

Définition *Point fixe*

Soit $\varphi : \mathbb{R} \to \mathbb{R}$ une fonction. Si $\widehat{x} \in \mathbb{R}$ est tel que $\varphi(\widehat{x}) = \widehat{x}$, on dit que $\widehat{x}$ est un point fixe de φ (l'image de $\widehat{x}$ par φ est lui-même).

On peut se demander comment exploiter cette procédure pour calculer les zéros d'une fonction donnée. Remarquons qu'on peut voir ℓ comme un point fixe du cosinus, ou encore comme un zéro de la fonction $f(x) = x - \cos(x)$. La méthode proposée fournit donc un moyen de calculer les zéros de f. Précisons ce principe : soit $f : [a, b] \to \mathbb{R}$ la fonction dont on cherche le zéro. Il est toujours possible de transformer le problème

(Pb-1) "chercher x tel que $f(x) = 0$"

en un problème équivalent (*i.e.* admettant les mêmes solutions)

(Pb-2) "chercher x tel que $x - \varphi(x) = 0$".

Pour que les deux problèmes soient équivalent, la fonction auxiliaire $\varphi : [a, b] \to \mathbb{R}$ doit être choisie de manière à ce que $\varphi(\widehat{x}) = \widehat{x}$ si et seulement si $f(\widehat{x}) = 0$ dans $[a; b]$ (on dit alors que le problème *(Pb-2)* est consistant avec le problème *(Pb-1)*).

Clairement, il existe une infinité de manières pour opérer cette transformation. Par exemple, on peut poser $\varphi(x) = x - f(x)$ ou plus généralement $\varphi(x) = x + \gamma f(x)$ avec $\gamma \in \mathbb{R}^*$ quelconque. On peut même remplacer γ par une fonction de x pour autant qu'elle ne s'annule pas.

Définition *Méthode de point fixe*

Supposons que $\widehat{x} \in \mathbb{R}$ soit un point fixe de φ. La méthode de point fixe consiste en la construction d'une suite $(x_k)_{k\in\mathbb{N}}$ définie par récurrence comme suit :

$$
\begin{cases}
x_0 & \text{donné},\\
x_{k+1} = \varphi(x_k) & \forall k \in \mathbb{N}.
\end{cases}
$$

Naturellement une telle suite n'est pas forcement convergente. Par contre, si elle converge, c'est-à-dire si la suite x_k a une limite que nous notons ℓ, et si φ est continue, alors cette limite est nécessairement un point fixe de φ puisque

$$
\ell = \lim_{k\to\infty} x_{k+1} = \lim_{k\to\infty} \varphi(x_k) = \varphi\left(\lim_{k\to\infty} x_k\right) = \varphi(\ell).
$$

On utilise alors l'algorithme itératif suivant pour construire la suite (comme l'ordinateur ne peux pas construire une infinité de termes, on calcul les premiers termes de la suite et on s'arrête dès que la différence entre deux éléments de la suite est inférieure à une tolérance $\varepsilon > 0$ donnée) :

Require: $x_0, \varepsilon, \varphi : [a, b] \to \mathbb{R}$
  $k \leftarrow 0$
  $x_1 \leftarrow x_0 + 2\varepsilon$
  while $|x_{k+1} - x_k| > \varepsilon$ **do**
    $x_{k+1} \leftarrow \varphi(x_k)$
    $k \leftarrow k + 1$
  end while

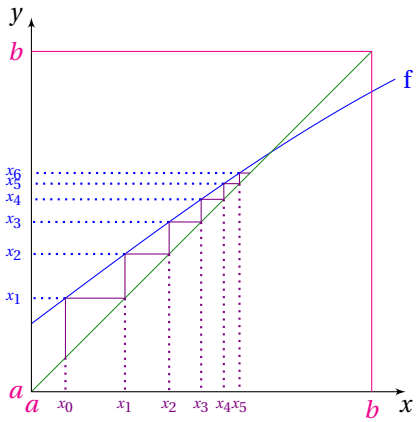
On va maintenant s'intéresser à la convergence de la suite construite par une méthode de point fixe.

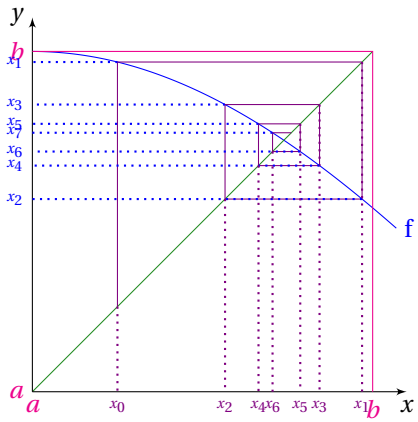
Théorème *Convergence (globale) des itérations de point fixe*

Considérons une fonction $\varphi : [a; b] \to \mathbb{R}$. On se donne $x_0 \in [a; b]$ et on considère la suite $x_{k+1} = \varphi(x_k)$ pour $k \geq 0$. Si les deux conditions suivantes sont satisfaites :

1. condition de stabilité : $\varphi(x) \in [a, b]$ pour tout $x \in [a, b]$

$\varphi([a;b]) \subset [a;b]$
$0 \leq \varphi'(x) < 1$
convergence





$\varphi([a;b]) \subset [a;b]$
$-1 < \varphi'(x) \leq 0$
convergence

FIGURE 1.1.: Interprétation géométrique du théorème de point fixe

2. condition de contraction stricte : il existe $K \in [0;1[$ tel que $|\varphi(x) - \varphi(y)| \leq K|x-y|$ pour tout $x, y \in [a,b]$ alors
- φ est continue,
- φ a un et un seul point fixe $\widehat{x}$ dans $[a,b]$,
- la suite $x_{k+1} = \varphi(x_k)$ converge vers $\widehat{x}$ pour tout choix de x_0 dans $[a,b]$.

Démonstration.

Continuité La condition de contraction stricte implique que φ *est continue* puisque, si on prend une suite $(y_k)_{k\in\mathbb{N}} \in [a,b]$ qui converge vers un élément x de $[a,b]$, alors nous avons $|\varphi(x) - \varphi(y_n)| \leq K|x-y_n|$ et par suite $\lim_{k\to\infty} \varphi(y_k) = \varphi(x)$.

Existence Commençons par prouver l'existence d'un point fixe de φ. La fonction $g(x) = \varphi(x) - x$ est continue dans $[a,b]$ et, grâce à la condition de stabilité, on a $g(a) = \varphi(a) - a \geq 0$ et $g(b) = \varphi(b) - b \leq 0$. En appliquant le théorème des valeurs intermédiaires, on en déduit que g a au moins un zéro dans $[a,b]$, *i.e.* φ a au moins un point fixe dans $[a,b]$.

Unicité L'unicité du point fixe découle de la condition de contraction stricte. En effet, si on avait deux points fixes distincts $\widehat{x}_1$ et $\widehat{x}_2$, alors

$$|\widehat{x}_1 - \widehat{x}_2| = |\varphi(\widehat{x}_1) - \varphi(\widehat{x}_2)| \leq K|\widehat{x}_1 - \widehat{x}_2| < |\widehat{x}_1 - \widehat{x}_2|$$

ce qui est impossible.

Convergence Prouvons à présent que la suite x_k converge vers l'unique point fixe $\widehat{x}$ quand k tend vers $+\infty$ pour toute donnée initiale $x_0 \in [a;b]$. On a

$$0 \leq |x_{k+1} - \widehat{x}| = |\varphi(x_k) - \varphi(\widehat{x})| \leq K|x_k - \widehat{x}|$$

où $K < 1$ est la constante de contraction. En itérant $k+1$ fois cette relation on obtient

$$|x_{k+1} - \widehat{x}| \leq K^{k+1}|x_0 - \widehat{x}|,$$

i.e., pour tout $k \geq 0$

$$\frac{|x_{k+1} - \widehat{x}|}{|x_0 - \widehat{x}|} \leq K^{k+1}.$$

En passant à la limite quand k tend vers $+\infty$ on obtient $|x_{k+1} - \widehat{x}|$ tend vers zéro.

□

Il est important de disposer d'un critère pratique assurant qu'une fonction φ est contractante stricte. Pour cela, rappelons quelques définitions.

Théorème

Si $\varphi\colon [a;b] \to [a;b]$ est de classe $\mathscr{C}^1([a,b])$ et si $|\varphi'(x)| < 1$ pour tout $x \in [a,b]$, alors la condition de contraction stricte est satisfaite avec $K = \max_{[a;b]} |\varphi'(x)|$.

Démonstration. Considérons la fonction affine g qui transforme l'intervalle $[0;1]$ dans l'intervalle $[x;y]$:

$$\begin{aligned} g\colon [0;1] &\to [x;y] \\ t &\mapsto x + t(x-y) \end{aligned}$$

Alors

$$\varphi(x) - \varphi(y) = \int_x^y \varphi'(\zeta)\,\mathrm{d}\zeta = \int_0^1 \varphi'(g(t))g'(t)\,\mathrm{d}t = (x-y)\int_0^1 \varphi'(x+t(x-y))\,\mathrm{d}t$$

et donc

$$\left|\varphi(x) - \varphi(y)\right| = \left|(x-y)\int_0^1 \varphi'(x+t(x-y))\,\mathrm{d}t\right| \leq \left|(x-y)\right| \int_0^1 \left|\varphi'(x+t(x-y))\right|\,\mathrm{d}t \leq K\left|(x-y)\right|.$$

□

Le théorème de convergence globale assure la convergence, avec un ordre 1, de la suite $(x_k)_{k\in\mathbb{N}}$ vers le point fixe $\widehat{x}$ pour tout choix d'une valeur initiale $x_0 \in [a;b]$. Mais en pratique, il est souvent difficile de déterminer à priori l'intervalle $[a;b]$; dans ce cas, le résultat de convergence locale suivant peut être utile.

Théorème *d'*OSTROWSKI *ou de convergence (locale) des itérations de point fixe*

Soit $[a;b] \in \mathbb{R}$ et $\varphi\colon [a;b] \to [a;b]$ une application de classe $\mathscr{C}^1([a;b])$. Soit $\widehat{x} \in [a;b]$ un point fixe de φ. On peut distinguer trois cas :

① Soit $|\varphi'(\widehat{x})| < 1$. Par continuité de φ' il existe un intervalle $[\widehat{x}-\varepsilon;\widehat{x}+\varepsilon] \subset [a;b]$ sur lequel $|\varphi'(\widehat{x})| < K < 1$, donc φ est contractante stricte sur cet intervalle. On a nécessairement $\varphi([\widehat{x}-\varepsilon;\widehat{x}+\varepsilon]) \subset [\widehat{x}-\varepsilon;\widehat{x}+\varepsilon]$ et par conséquent la suite $(x_k)_{k\in\mathbb{N}}$ converge vers $\widehat{x}$ pour tout $x_0 \in [\widehat{x}-\varepsilon;\widehat{x}+\varepsilon]$. On dit que $\widehat{x}$ est un *point fixe attractif*. De plus,
- ⋆ si $0 < \varphi'(\widehat{x}) < 1$ la suite converge de façon monotone, c'est-à-dire l'erreur $x_k - \widehat{x}$ garde un signe constant quand k varie ;
- ⋆ si $-1 < \varphi'(\widehat{x}) < 0$ la suite converge de façon oscillante, c'est-à-dire l'erreur $x_k - \widehat{x}$ change de signe selon la parité de k.

② Si $|\varphi'(\widehat{x})| > 1$, alors il n'existe aucun intervalle $[\widehat{x}-\varepsilon;\widehat{x}+\varepsilon] \subset [a;b]$ tel que la suite $(x_k)_{k\in\mathbb{N}}$ converge vers $\widehat{x}$ pour tout $x_0 \in [\widehat{x}-\varepsilon;\widehat{x}+\varepsilon]$ à l'exception du cas $x_0 = \widehat{x}$. On dit que $\widehat{x}$ est un *point fixe répulsif*. De plus,
- ⋆ si $\varphi'(\widehat{x}) > 1$ la suite diverge de façon monotone,
- ⋆ si $\varphi'(\widehat{x}) < -1$ la suite diverge en oscillant.

③ Si $|\varphi'(\widehat{x})| = 1$ on ne peut en général tirer aucune conclusion : selon le problème considéré, il peut y avoir convergence ou divergence.

Exemple

- ⋆ La fonction $\varphi(x) = \cos(x)$ vérifie toutes les hypothèses du théorème d'OSTROWSKI : elle est de classe $\mathscr{C}^\infty(\mathbb{R})$ et $|\varphi'(\widehat{x})| = |\sin(\widehat{x})| \simeq 0.67 < 1$, donc il existe par continuité un intervalle $[c,d]$ qui contient $\widehat{x}$ tel que $|\varphi'(\widehat{x})| < 1$ pour $x \in [c,d]$.
- ⋆ La fonction $\varphi(x) = x^2 - 1$ possède deux points fixes $\widehat{x}_1 = (1+\sqrt{5})/2$ et $\widehat{x}_2 = (1-\sqrt{5})/2$ mais ne vérifie l'hypothèse du théorème d'OSTROWSKI pour aucun d'eux puisque $|\varphi(\widehat{x}_{1,2})| = |(1\pm\sqrt{5})/2| > 1$. Les itérations de point fixe ne convergent pas.
- ⋆ La fonction $\phi(x) = x - x^3$ admet $\widehat{x} = 0$ comme point fixe. On a $\phi'(\widehat{x}) = 1$ et $x_k \to \widehat{x}$ pour tout $x_0 \in [-1;1]$ car
  - ⋆ si $x_0 = \pm 1$ alors $x_k = \widehat{x}$ pour tout $k \geq 1$,
  - ⋆ si $x_0 \in]-1,1[$ alors on montre que $x_k \in]-1,1[$ pour tout $k \geq 1$. De plus, la suite est monotone décroissante si $0 < x_0 < 1$, monotone croissante si $-1 < x_0 < 0$ donc elle converge vers $\ell \in [-1;1]$. Les uniques candidats limites sont les solutions de l'équation $\ell = \phi(\ell)$ et par conséquente $x_k \to 0$.
- ⋆ La fonction $\phi(x) = x + x^3$ admet aussi $\widehat{x} = 0$ comme point fixe. À nouveau $\phi'(\widehat{x}) = 1$ mais dans ce cas la suite diverge pour tout choix de $x_0 \neq 0$.

Le théorème d'OSTROWSKI dit que, de manière générale, la méthode de point fixe ne converge pas pour des valeurs arbitraires de x_0, mais seulement pour des valeurs suffisamment proches de $\widehat{x}$, c'est-à-dire appartenant à un certain voisinage de $\widehat{x}$. Au premier abord, cette condition semble inutilisable : elle signifie en effet que pour calculer $\widehat{x}$ (qui est inconnu), on devrait partir d'une valeur assez proche de $\widehat{x}$. En pratique, on peut obtenir une valeur initiale x_0 en effectuant quelques itérations de la méthode de dichotomie ou en examinant le graphe de f. Si x_0 est convenablement choisi alors la méthode de point fixe converge.

Proposition *Calcul de l'ordre de convergence d'une méthode de point fixe*

Soit $\widehat{x}$ un point fixe d'une fonction $\varphi \in \mathscr{C}^{p+1}$ pour un entier $p \geq 1$ dans un intervalle $[a;b]$ contenant $\widehat{x}$. Si $\varphi^{(i)}(\widehat{x}) = 0$ pour $1 \leq i \leq p$ et $\varphi^{(p+1)}(\widehat{x}) \neq 0$, alors la méthode de point fixe associée à la fonction d'itération φ est d'ordre $p+1$ et

$$\lim_{k\to+\infty} \frac{x_{k+1} - \widehat{x}}{(x_k - \widehat{x})^{p+1}} = \frac{\varphi^{(p+1)}(\widehat{x})}{(p+1)!}.$$

Démonstration. Écrivons le développement de TAYLOR avec le reste de LAGRANGE de φ en $x = \widehat{x}$:

$$\varphi(x) = \varphi(\widehat{x}) + \sum_{i=1}^{p} \frac{\varphi^{(i)}(\widehat{x})}{i!}(x-\widehat{x})^i + \frac{\varphi^{(p+1)}(\xi)}{(p+1)!}(x-\widehat{x})^{p+1}$$

où ξ est entre x et $\widehat{x}$. Comme $\varphi(\widehat{x}) = \widehat{x}$ et $\varphi^{(i)}(\widehat{x}) = 0$ pour $1 \le i \le p$, cela se simplifie et on a

$$\varphi(x) = \widehat{x} + \frac{\varphi^{(p+1)}(\xi)}{(p+1)!}(x-\widehat{x})^{p+1}.$$

En évaluant l'expression ainsi trouvée en x_k et sachant que $\varphi(x_k) = x_{k+1}$, on a alors

$$x_{k+1} - \widehat{x} = \frac{\varphi^{(p+1)}(\xi)}{(p+1)!}(x_k-\widehat{x})^{p+1}.$$

Lorsque $k \to +\infty$, $x_k$tend vers $\widehat{x}$ et donc ξ, qui se trouve entre x_k et $\widehat{x}$, tend vers $\widehat{x}$ aussi. Alors

$$\lim_{k\to+\infty} \frac{x_{k+1}-\widehat{x}}{(x_k-\widehat{x})^{p+1}} = \lim_{k\to+\infty} \frac{\varphi^{(p+1)}(\xi)}{(p+1)!} = \frac{\varphi^{(p+1)}(\widehat{x})}{(p+1)!}.$$

Pour un ordre p fixé, la convergence de la suite vers $\widehat{x}$ est d'autant plus rapide que $\frac{\varphi^{(p+1)}(\widehat{x})}{(p+1)!}$ est petit. □

Méthodes de point fixe particulièrement connues

Soit $f\colon [a,b] \to \mathbb{R}$ une fonction continue (continûment dérivable pour la méthode de la corde 2 et la méthode de NEWTON) et soit $\widehat{x}$ un zéro de f. Supposons que l'on connaisse une valeur x_0 proche de $\widehat{x}$. Approcher les zéros de f se ramène au problème de la détermination des points fixes de la fonction φ, ce qui se fait en construisant la suite récurrente

$$\begin{cases} x_0 \text{ donné,} \\ x_{k+1} = \varphi(x_k). \end{cases}$$

Pour choisir la fonction φ il est nécessaire de prendre en compte les informations données par les valeurs de f et, éventuellement, par sa dérivée f' ou par une approximation convenable de celle-ci (si f est différentiable). Écrivons pour cela le développement de TAYLOR de f en $\widehat{x}$ au premier ordre : $f(\widehat{x}) = f(x) + (\widehat{x}-x)f'(\xi)$ où ξ est entre $\widehat{x}$ et x. Le problème

> "chercher $\widehat{x}$ tel que $f(\widehat{x}) = 0$"

devient alors

> "chercher $\widehat{x}$ tel que $f(x) + (\widehat{x}-x)f'(\xi) = 0$".

Cette équation conduit à la méthode itérative suivante :

> "pour tout $k \ge 0$, étant donné x_k, déterminer x_{k+1} en résolvant l'équation $f(x_k) + (x_{k+1}-x_k)q_k = 0$, où q_k est égal à $f'(\xi_k)$ (ou en est une approximation) avec ξ_k un point entre x_k et x_{k+1}."

La méthode qu'on vient de décrire revient à chercher l'intersection entre l'axe des x et la droite de pente q_k passant par le point $(x_k, f(x_k))$, ce qui s'écrit sous la forme d'une méthode de point fixe avec

$$x_{k+1} = \varphi(x_k) \equiv x_k - \frac{f(x_k)}{q_k}, \qquad k \ge 0.$$

Considérons maintenant quatre choix particuliers de q_k et donc de φ qui définissent des méthodes célèbres :

Méthode de la Corde 1 :	$q_k = \dfrac{b-a}{f(b)-f(a)}$	$\Longrightarrow$	$\varphi(x) = x - \dfrac{b-a}{f(b)-f(a)}f(x)$
Méthode de la Corde 2 :	$q_k = f'(x_0)$	$\Longrightarrow$	$\varphi(x) = x - \dfrac{f(x)}{f'(x_0)}$
Méthode de NEWTON :	$q_k = f'(x_k)$	$\Longrightarrow$	$\varphi(x) = x - \dfrac{f(x)}{f'(x)}$

Proposition

Si la méthode de la corde converge, elle converge à l'ordre 1 ; si la méthode de NEWTON converge, elle converge à l'ordre 2 si la racine est simple, à l'ordre 1 sinon.

Démonstration.

Méthodes de la Corde $q_k = \frac{f(b)-f(a)}{b-a}$ (méthode 1) ou $q_k = f'(x_0)$ (méthode 2). Si $f'(\widehat{x}) = 0$ alors $\varphi'(\widehat{x}) = 1$ et on ne peut pas assurer la convergence de la méthode. Autrement, la condition $|\varphi'(\widehat{x})| < 1$ revient à demander que $0 < f'(\widehat{x})/q_k < 2$. Ainsi la pente de la corde doit avoir le même signe que $f'(\widehat{x})$ et, pour la méthode 1, l'intervalle de recherche $[a;b]$ doit être tel quel

$$b - a < 2\frac{f(b)-f(a)}{f'(\widehat{x})}.$$

La méthode de la corde converge en une itération si f est affine, autrement elle converge linéairement, sauf dans le cas (exceptionnel) où $f'(\widehat{x}) = \frac{f(b)-f(a)}{b-a}$ (méthode 1) ou $f'(\widehat{x}) = f'(x_0)$ (méthode 2), *i.e.* $\varphi'(\widehat{x}) = 0$ (la convergence est alors au moins quadratique).

Méthode de Newton Soit la méthode de NEWTON pour le calcul de $\widehat{x}$ zéro de f :

$$\varphi(x) = x - \frac{f(x)}{f'(x)}.$$

⋆ Si $f'(\widehat{x}) \neq 0$ (*i.e.* si $\widehat{x}$ est racine simple), on trouve

$$\varphi'(x) = 1 - \frac{(f'(x))^2 - f(x)f''(x)}{(f'(x))^2} = \frac{f(x)f''(x)}{(f'(x))^2}, \qquad \varphi'(\widehat{x}) = 0,$$
$$\varphi''(x) = \frac{f''(x)}{f'(x)} + \frac{f(x)f'''(x)}{(f'(x))^2} - 2\frac{f(x)(f''(x))^2}{(f'(x))^3}, \qquad \varphi''(\widehat{x}) = \frac{f''(\widehat{x})}{f'(\widehat{x})}.$$

La méthode de NEWTON est donc d'ordre 2.

⋆ Si la racine $\widehat{x}$ est de multiplicité $m > 1$, alors la méthode n'est plus du second ordre. En effet, $f(x) = (x-\widehat{x})^m h(x)$ où h est une fonction telle que $h(\widehat{x}) \neq 0$. On a alors

$$\varphi(x) = 1 - \frac{f(x)}{f'(x)} = 1 - \frac{(x-\widehat{x})h(x)}{mh(x) + (x-\widehat{x})h'(x)},$$
$$\varphi'(x) = \frac{h(x)\big(m(m-1)h(x) + 2(x-\widehat{x})h'(x) + (x-\widehat{x})^2 h''(x)\big)}{\big(mh(x) + (x-\widehat{x})h'(x)\big)^2}, \qquad \varphi'(\widehat{x}) = 1 - \frac{1}{m}.$$

Si la valeur de m est connue *a priori*, on peut retrouver la convergence quadratique en modifiant la méthode de NEWTON comme suit :

$$\varphi(x) = x - m\frac{f(x)}{f'(x)}.$$

□

Attention

À noter que même si la méthode de NEWTON permet en général d'obtenir une convergence quadratique, un mauvais choix de la valeur initiale peut provoquer la divergence de cette méthode (notamment si la courbe représentative de f présente au point d'abscisse x_0 un tangente à peu près horizontale). D'où l'importance d'une étude préalable soignée de la fonction f (cette étude est d'ailleurs nécessaire pour toute méthode de point fixe).

Remarque *Interprétation géométrique de la méthode de* NEWTON *et des méthodes de la corde*

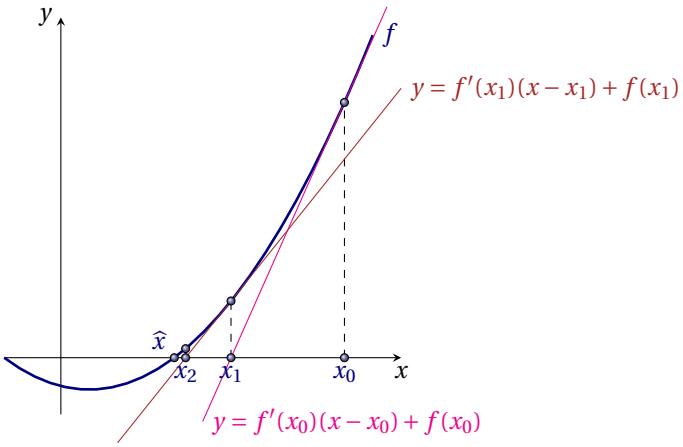
Soit $f : \mathbb{R} \to \mathbb{R}$ une fonction continûment dérivable et soit $\widehat{x}$ un zéro simple de f, c'est-à-dire $f(\widehat{x}) = 0$ et $f'(\widehat{x}) \neq 0$. Supposons que l'on connaisse une valeur x_k proche de $\widehat{x}$. Pour calculer x_{k+1} on prend l'intersection de l'axe des abscisses avec la droite tangente au graphe de f passant par le point $(x_k, f(x_k))$, *i.e.* on cherche x solution du système linéaire

$$\begin{cases} y = f'(x_k)(x - x_k) + f(x_k), \\ y = 0. \end{cases}$$

On obtient

$$x = x_k - \frac{f(x_k)}{f'(x_k)}$$

ce qui correspond à la méthode de NEWTON.



Soit $f : \mathbb{R} \to \mathbb{R}$ une fonction continue et soit $\widehat{x} \in [a, b]$ un zéro de f. Cette fois-ci, pour calculer x_{k+1} on prend l'intersection de l'axe des abscisses avec la droite passant par le point $(x_k, f(x_k))$ et parallèle à la droite passant par les points $(a, f(a))$ et $(b, f(b))$, *i.e.* on cherche x solution du système linéaire

$$\begin{cases} y = \frac{f(b)-f(a)}{b-a}(x - x_k) + f(x_k), \\ y = 0, \end{cases}$$

ce qui donne

$$x = x_k - \frac{b-a}{f(b) - f(a)} f(x_k).$$

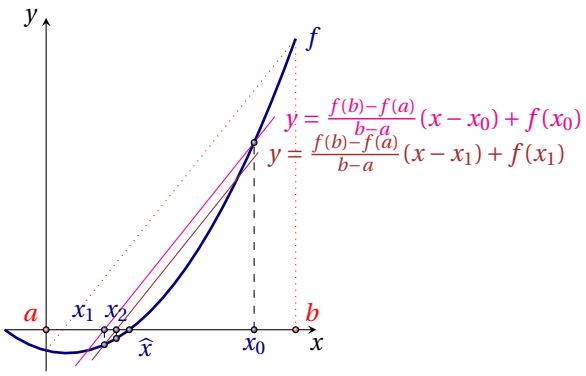
Il s'agit de la méthode de la corde 1. Cette méthode permet d'éviter qu'à chaque itération on ait à évaluer $f'(x_k)$ car on remplace $f'(x_k)$ par $\frac{f(b)-f(a)}{b-a}$. Une variante de la méthode de la corde consiste à calculer x_{k+1} comme l'intersection entre l'axe des abscisses et la droite passant par le point $(x_k, f(x_k))$ et parallèle à la droite tangente au graphe de f passant par le point $(x_0, f(x_0))$, *i.e.* on cherche x solution du système linéaire

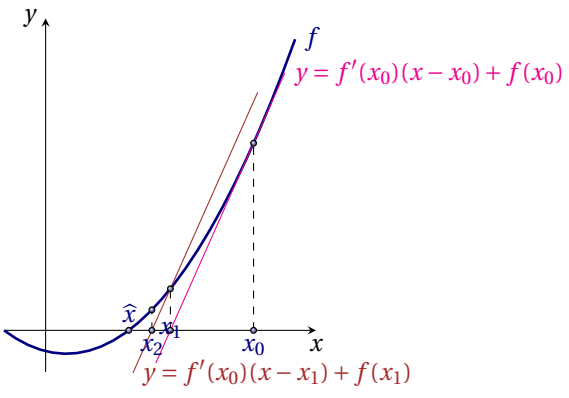
$$\begin{cases} y = f'(x_0)(x - x_k) + f(x_k), \\ y = 0, \end{cases}$$

ce qui donne

$$x = x_k - \frac{f(x_k)}{f'(x_0)}$$

Dans cette variante on remplace $f'(x_k)$ par $f'(x_0)$.





Exemple

On se trouve en possession d'une calculatrice qui ne sait effectuer que les opérations addition, soustraction et multiplication. Lorsque $a > 0$ est donné, on veut calculer sa valeur réciproque $1/a$. Le problème peut être ramené à résoudre l'équation $x = 1/a$ ce qui équivaut à chercher le zéro de la fonction

$$\begin{aligned} f : \mathbb{R}_*^+ &\to \mathbb{R} \\ x &\mapsto \frac{1}{x} - a \end{aligned}$$

Selon la formule de NEWTON on a

$$x_{k+1} = (1 + a)x_k + x_k^2,$$

une récurrence qui ne requiert pas de divisions. Pour $a = 7$ et partant de $x_0 = 0.2$ par exemple, on trouve $x_1 = 0{,}12$, $x_2 = 0{,}1392$, $x_3 = 0{,}142\,763\,520\,0$, $x_4 = 0{,}142\,857\,081\,5$, etc. Cette suite converge vers $1/7 \simeq 0{,}142\,857\,142\,857$.

Exemple *Comparaison des méthodes de* NEWTON *pour différentes formulation de la fonction initiale*

Dans $\mathbb{R}_+^*$ on veut résoudre l'équation

$$x = e^{1/x}. \tag{1.1}$$

En transformant l'équation donnée de différentes manières, on arrive à différentes formules de récurrence :

1. L'équation (1.1) équivaut à chercher le zéro de la fonction

$$\begin{aligned} f : \mathbb{R}_+^* &\to \mathbb{R} \\ x &\mapsto x - e^{1/x} \end{aligned}$$

En utilisant la méthode de NEWTON on trouve la formule itérative

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} = x_k - \frac{x_k - e^{1/x_k}}{1 + \frac{e^{1/x_k}}{x_k^2}} = x_k - x_k^2 \frac{x_k - e^{1/x_k}}{x_k^2 + e^{1/x_k}}.$$

2. Si on pose $y = 1/x$, alors on a l'équivalence

$$x = e^{1/x} \qquad \Longleftrightarrow \qquad y = e^{-y},$$

donc la solution x de l'équation (1.1) est la réciproque du zéro de la fonction

$$\begin{aligned} g : \mathbb{R}_+^* &\to \mathbb{R} \\ y &\mapsto 1 - ye^y \end{aligned}$$

En utilisant la méthode de NEWTON on trouve la formule itérative

$$y_{k+1} = y_k - \frac{g(y_k)}{g'(y_k)} = y_k - \frac{1 - y_k e^{y_k}}{-(1+y_k)e^{y_k}} = y_k + \frac{e^{-y_k} - y_k}{1 + y_k}$$

et $x_k = 1/y_k$.

3. L'équation (1.1) est encore équivalente à chercher le zéro de la fonction

$$\begin{aligned} h : \mathbb{R}_+^* &\to \mathbb{R} \\ x &\mapsto 1 - x\ln(x) \end{aligned}$$

En utilisant la méthode de NEWTON on trouve la formule itérative

$$x_{k+1} = x_k - \frac{h(x_k)}{h'(x_k)} = x_k + \frac{1 - x_k \ln(x_k)}{1 + \ln(x_k)} = \frac{1 + x_k}{1 + \ln(x_k)}.$$

La représentation graphique de f montre qu'il n'existe qu'une seule racine. Comme $f(1.7)f(1.9) < 0$, elle se trouve dans l'intervalle $[1.7; 1.9]$. En partant de $x_0 = 1.8$ on trouve les suites suivantes :

	Formule 1	Formule 2	Formule 3
$x_1 =$	1,762 878 141 2	1.7418849724	1.7634610883
$x_2 =$	1.7632228030	1.7751466845	1.7632228446
$x_3 =$	1.7632228344	1.7564077294	1.7632228344

La solution est $x \simeq 1{,}763\,222\,834\,35$.

Critères d'arrêt

Supposons que $(x_n)_{n\in\mathbb{N}}$ soit une suite qui converge vers $\widehat{x}$ zéro de la fonction f. Nous avons le choix entre deux types de critères d'arrêt pour interrompre le processus itératif d'approximation de $\widehat{x}$: ceux basés sur le résidu et ceux basés sur l'incrément. Nous désignerons par ε une tolérance fixée pour le calcul approché de $\widehat{x}$ et par $e_n = \widehat{x} - x_n$ l'erreur absolue. Nous supposerons de plus f continûment différentiable dans un voisinage de la racine.

Contrôle du résidu : les itérations s'achèvent dès que $|f(x_n)| < \varepsilon$. Il y a des situations pour lesquelles ce test s'avère trop restrictif ou, au contraire, trop optimiste.

- ⋆ si $|f'(\widehat{x})| \simeq 1$ alors $|e_n| \simeq \varepsilon$: le test donne donc une indication satisfaisante de l'erreur ;
- ⋆ si $|f'(\widehat{x})| \ll 1$, le test n'est pas bien adapté car $|e_n|$ peut être assez grand par rapport à ε (voir la figure 1.2 à droite) ;
- ⋆ si enfin $|f'(\widehat{x})| \gg 1$ alors $|e_n| \ll \varepsilon$ et le test est trop restrictif (voir la figure 1.2 à gauche).

Contrôle de l'incrément : les itérations s'achèvent dès que $|x_{n+1} - x_n| < \varepsilon$. Soit $(x_n)_{n\in\mathbb{N}}$ la suite produite par la méthode de point fixe $x_{n+1} = \varphi(x_n)$. Comme $\widehat{x} = \varphi(\widehat{x})$ et $x_{n+1} = \varphi(x_n)$, si on développe au premier ordre on sait qu'il existe $\xi_n \in I_{\widehat{x},x_n}$ tel que

$$e_{n+1} = \widehat{x} - x_{n+1} = \varphi(\widehat{x}) - \varphi(x_n) = \varphi'(\xi_n)(\widehat{x} - x_n) = \varphi'(\xi_n)e_n$$

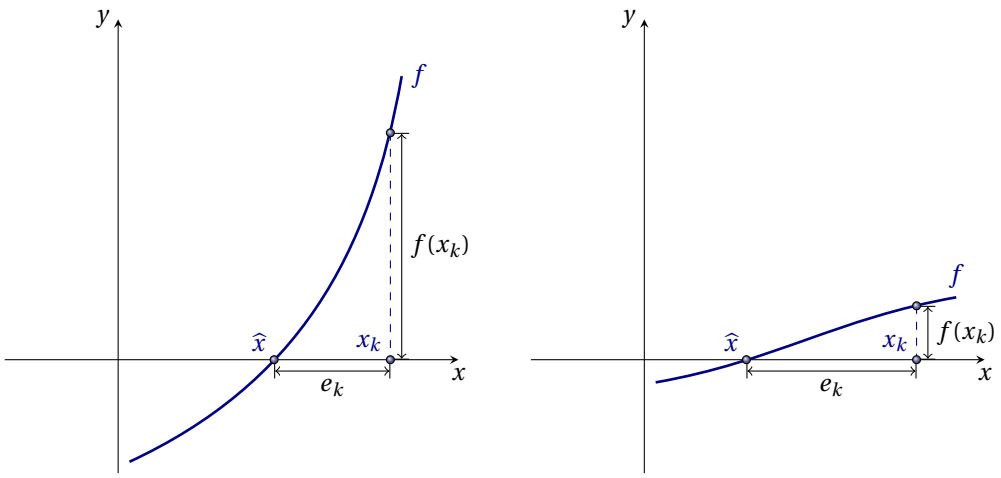


FIGURE 1.2.: Deux situations pour lesquelles le résidu $e_k = x_k - \widehat{x}$ est un mauvais estimateur d'erreur : $|f'(x)| \gg 1$ (à gauche), $|f'(x)| \ll 1$ (à droite), pour x dans un voisinage de $\widehat{x}$.

où $I_{\widehat{x},x_n}$ est l'intervalle d'extrémités $\widehat{x}$ et x_k. En utilisant l'identité

$$e_n = (\widehat{x} - x_{n+1}) + (x_{n+1} - x_n) = e_{n+1} + (x_{n+1} - x_n) = \varphi'(\xi_n)e_n + (x_{n+1} - x_n),$$

on en déduit que

$$e_n = \frac{x_{n+1} - x_n}{1 - \varphi'(\xi_n)}.$$

Par conséquent, ce critère fournit un estimateur d'erreur satisfaisant si $\varphi'(x) \simeq 0$ dans un voisinage de $\widehat{x}$. C'est le cas notamment des méthodes d'ordre 2, dont la méthode de NEWTON. Cette estimation devient d'autant moins bonne que φ' s'approche de 1.

Notons d'ailleurs que si la méthode de point fixe converge avec $K < 1$ et si on considère le critère d'arrêt $|x_{n+1} - x_n| < \varepsilon$ alors

$$|e_n| = |x_n - \widehat{x}| \leq \frac{\varepsilon}{1 - K} \leq 2\varepsilon.$$

En effet, il suffit de considérer les inégalités suivantes :

$$|e_{n+1}| = |x_{n+1} - \widehat{x}| = |\varphi(x_n) - \varphi(\widehat{x})| \leq K|x_n - \widehat{x}| = K|e_n|,$$
$$|e_{n+1}| = |x_{n+1} - \widehat{x}| = |x_{n+1} - x_n + x_n - \widehat{x}| \geq |x_n - \widehat{x}| - |x_{n+1} - x_n| = |e_n| - |x_{n+1} - x_n| > |e_n| - \varepsilon$$

✶✶✶✶✶✶✶✶✶✶✶✶✶ Codes Python ✶✶✶✶✶✶✶✶✶✶✶✶✶

`dichotomie`, `lagrange`, `newton` et `point_fix` sont quatre fonctions (informatiques) qui renvoient la valeur approchée du zéro d'une fonction (mathématique) f. En paramètre elles reçoivent `f`, la fonction dont on cherche la racine, `a` et `b` sont les extrémités de l'intervalle de recherche pour les méthodes de dichotomie et de LAGRANGE, `x_init` est la donnée initiale pour les méthodes de NEWTON et de point fixe, `maxITER` est le nombre maximal d'itérations et `tol` est la tolérance.

Méthodes numériques.

```
#!/usr/bin/python
#-*- coding: Utf-8 -*-

import math, sys

def dichotomie(f,a,b,tol,maxITER):
    fa = f(a)
    if abs(fa)<=tol:
        return a
    fb = f(b)
    if abs(fb)<=tol:
        return b
    if fa*fb > 0.0:
        print "La racine n'est pas encadree"
        sys.exit(0)
    n = int(math.ceil(math.log(abs(b-a)/tol)/math.log(2.0)))
    for k in range(min(n+1,maxITER)):
        c = (a+b)*0.5
        fc = f(c)
        if fc == 0.0:
            return c
        if fc*fb < 0.0:
            a = c
            fa = fc
        else:
            b = c
            fb = fc
    return (a+b)*0.5

def lagrange(f,a,b,tol,maxITER):
    fa = f(a)
    if abs(fa)<=tol:
        return a
    fb = f(b)
    if abs(fb)<=tol:
        return b
    if fa*fb > 0.0:
        print "La racine n'est pas encadree"
        sys.exit(0)
    k = 0
    while ( ((abs(b-a)>tol) or (abs(fc)>tol)) and (k<maxITER) ):
        k += 1
        c = a-fa*(b-a)/(fb-fa)
        fc = f(c)
        if fc == 0.0:
            return c
        if fc*fb < 0.0:
            a = c
            fa = fc
        else:
            b = c
            fb = fc
    return a-fa*(b-a)/(fb-fa)

def newton(f,x_init,tol,maxITER):
    k = 0
    x = x_init
    fx = f(x)
```

```
    h = tol
    dfx = (f(x+h)-fx)/h # calcul approche de f'(x)
    while ( (abs(fx)>tol) and (k<maxITER) ):
        x = x - fx/dfx
        fx = f(x)
        dfx = (f(x+h)-fx)/h
        k += 1
    if k==maxITER:
        print "Pas de convergence"
    else:
        return x

def point_fix(f,x_init,tol,maxITER):
    k = 0
    x = x_init
    while ( (abs(phi(x)-x)>tol) and (k<maxITER) ):
        x = phi(x)
        k += 1
    if k==maxITER:
        print "Pas de convergence"
    else:
        return x
```

Exemple d'utilisation

```
# CHOIX DU CAS TEST
exemple = 2

# DEFINITION DU CAS TEST
if exemple==1 :
    tol = 1.0e-9
    maxITER = 100
    def f(x):
        return (x+1)*(x-2)
    def phi(x):
        return x**2-2
elif exemple==2 :
    tol = 1.0e-9
    maxITER = 100
    def f(x):
        return x**2-2
    def phi(x):
        return x**2+x-2
else:
    print "Cas test non defini"
    sys.exit(0)


# CALCUL
a = -3.
b = 0.
print "A) Zero calcule par la methode de dichotomie dans l'intervalle [", a, ",", b,"] : ", dichotomie(f,a
    ➥,b,tol,maxITER)
print "B) Zero calcule par la methode de Lagrange dans l'intervalle [", a, ",", b,"] : ", lagrange(f,a,b,
    ➥tol,maxITER)
a = 0.
b = 3.
print "C) Zero calcule par la methode de dichotomie dans l'intervalle [", a, ",", b,"] : ", dichotomie(f,a
    ➥,b,tol,maxITER)
print "D) Zero calcule par la methode de Lagrange dans l'intervalle [", a, ",", b,"] : ", lagrange(f,a,b,
    ➥tol,maxITER)


x_init = 0.
```

```
116 print "E) Zero calcule par la methode de \textsc{Newton} a partir du point x_0 =",x_init," : ", newton(f,
        ➥x_init,tol,maxITER)
117 print "F) Zero calcule par la methode de point fix a partir du point x_0 =",x_init," : ", point_fix(phi,
        ➥x_init,tol,maxITER)
118
119 x_init = 1.
120 print "G) Zero calcule par la methode de \textsc{Newton} a partir du point x_0 =",x_init," : ", newton(f,
        ➥x_init,tol,maxITER)
121 print "H) Zero calcule par la methode de point fix a partir du point x_0 =",x_init," : ", point_fix(phi,
        ➥x_init,tol,maxITER)
122
123
124 # Dans python il existe un module qui implement deja ces methodes, comparons nos resultats avec ceux du
        ➥module:
125 from scipy.optimize import fsolve
126 x_init = 0.
127 print "** Zero calcule par le module scipy.optimize a partir du point x_0 =",x_init," : ", fsolve(f,x_init
        ➥)
128 x_init = 1.
129 print "** Zero calcule par le module scipy.optimize a partir du point x_0 =",x_init," : ", fsolve(f,x_init
        ➥)
```

Exercices

Exercice 1.1

Décrire les méthodes de la dichotomie et de LAGRANGE et les utiliser pour calculer le zéro de la fonction

$$f(x) = x^3 - 4x - 8.95$$

dans l'intervalle $[2;3]$ avec une précision de 10^{-2}.

CORRECTION DE L'EXERCICE 1.1. En partant de $I_0 = [a,b]$, les méthodes de la dichotomie et de LAGRANGE produisent une suite de sous-intervalles $I_k = [a_k, b_k]$ avec $I_k \subset I_{k-1}$, $k \geq 1$, et tels que $f(a_k)f(b_k) < 0$. Pour cela, soit x_k tel que $I_k = [a_k; x_k] \cup [x_k, b_k]$, alors

$$I_{k+1} = \begin{cases} [a_k; x_k] & \text{si } f(a_k)f(x_k) < 0, \\ [x_k, b_k] & \text{sinon.} \end{cases}$$

$k \leftarrow 0$
$a_k \leftarrow 2$
$b_k \leftarrow 3$
while $|b_k - a_k| > 0.01$ **do**
  $x_k \leftarrow g(a_k, b_k)$
  $k \leftarrow k+1$
  if $(a_k^3 - 4a_k - 8.95)(x_k^3 - 4x_k - 8.95) < 0$ **then**
    $a_{k+1} \leftarrow a_k$
    $b_{k+1} \leftarrow x_k$
  else
    $a_{k+1} \leftarrow x_k$
    $b_{k+1} \leftarrow b_k$
  end if
end while

L'unique différence entre les deux méthode réside dans la construction de x_k :

$$x_k = g(a_k, b_k) = \begin{cases} \frac{a_k + b_k}{2} & \text{pour la méthode de la dichotomie,} \\ \frac{a_k f(b_k) - b_k f(a_k)}{f(b_k) - f(a_k)} & \text{pour la méthode de la LAGRANGE.} \end{cases}$$

Dichotomie

k	a_k	x_k	b_k	signe de $f(a_k)$	signe de $f(x_k)$	signe de $f(b_k)$
0	2.000000	2.5000000	3.00000	−	−	+
1	2.500000	2.7500000	3.00000	−	+	+
2	2.500000	2.6250000	2.75000	−	−	+
3	2.625000	2.6875000	2.75000	−	−	+
4	2.687500	2.7187500	2.75000	−	+	+
5	2.687500	2.7031250	2.71875	−	−	+
6	2.703125	2.7109375	2.71875	−	+	+

LAGRANGE

k	a_k	x_k	b_k	signe de $f(a_k)$	signe de $f(x_k)$	signe de $f(b_k)$
0	2.000000	2.596666667	3.00000	−	−	+
1	2.596666667	2.690262642	3.00000	−	−	+
2	2.690262642	2.702092263	3.00000	−	−	+
3	2.702092263	2.703541518	3.00000	−	−	+
4	2.703541518	2.703718378	3.00000	−	−	+
5	2.703718378	2.703739951	3.00000	−	−	+
6	2.703739951	2.703742582	3.00000	−	−	+

Exercice 1.2

Soit $f : [0;1] \to \mathbb{R}$ une fonction continue strictement décroissante telle que $f(0) = 1$ et $f(1) = -1$.

1. Sachant que $f(0.3) = 0$, déterminer la suite des premiers quatre itérés de la méthode de la dichotomie dans l'intervalle $[0;1]$ pour l'approximation du zéro de f. On pourra utiliser le tableau ci-dessous :

k	a_k	x_k	b_k	signe de $f(a_k)$	signe de $f(x_k)$	signe de $f(b_k)$
0	0		1	+		−
1						
2						
3						
4						

2. Combien d'itérations faut-il effectuer pour approcher le zéro de f à 2^{-5} près ?

CORRECTION DE L'EXERCICE 1.2.

1. On a

k	a_k	x_k	b_k	signe de $f(a_k)$	signe de $f(x_k)$	signe de $f(b_k)$
0	0	0.5	1	+	−	−
1	0	0.25	0.5	+	+	−
2	0.25	0.375	0.5	+	−	−
3	0.25	0.3125	0.375	+	−	−
4	0.25	0.28125	0.3125	+	+	−

donc, après quatre itérations, le zéro de f est approché par 0.28125.

2. Il faut effectuer au moins $\log_2\left(\frac{1-0}{2^{-5}}\right) = 5$ itérations.

Exercice 1.3

Soit $f\colon [0;1] \to \mathbb{R}$ une fonction continue strictement décroissante telle que $f(0) = 1$ et $f(1) = -1$.

1. Sachant que $f(0.6) = 0$, déterminer la suite des premiers quatre itérés de la méthode de la dichotomie dans l'intervalle $[0;1]$ pour l'approximation du zéro de f. On pourra utiliser le tableau ci-dessous :

k	a_k	x_k	b_k	signe de $f(a_k)$	signe de $f(x_k)$	signe de $f(b_k)$
0	0		1	+		−
1						
2						
3						
4						

2. Combien d'itérations faut-il effectuer pour approcher le zéro de f à 2^{-10} près ?

CORRECTION DE L'EXERCICE 1.3.

1. On a

k	a_k	x_k	b_k	signe de $f(a_k)$	signe de $f(x_k)$	signe de $f(b_k)$
0	0	0.5	1	+	+	−
1	0.5	0.75	1	+	−	−
2	0.5	0.625	0.75	+	−	−
3	0.5	0.5625	0.625	+	+	−
4	0.5625	0.59375	0.625	+	+	−

donc, après quatre itérations, le zéro de f est approché par 0.59375.

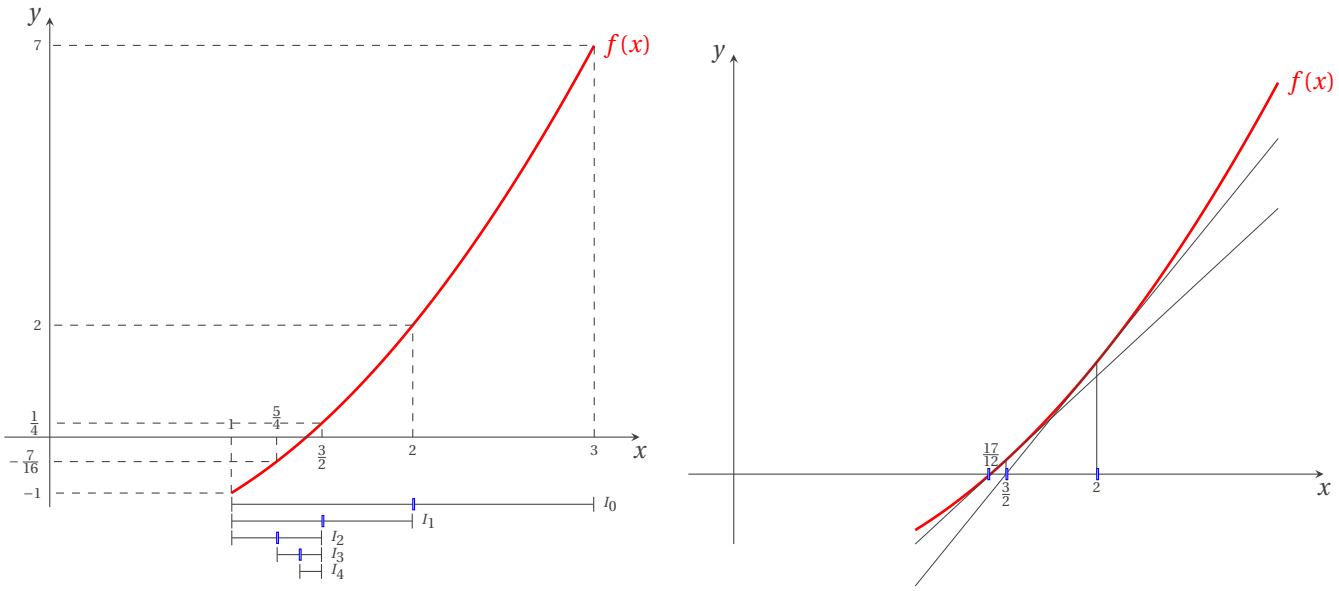
2. Il faut effectuer au moins $\log_2\left(\frac{1-0}{2^{-10}}\right) = 10$ itérations.

Exercice 1.4

Déterminer la suite des premiers 3 itérés des méthodes de dichotomie dans l'intervalle $[1,3]$ et de NEWTON avec $x_0 = 2$ pour l'approximation du zéro de la fonction $f(x) = x^2 - 2$. Combien de pas de dichotomie doit-on effectuer pour améliorer d'un ordre de grandeur la précision de l'approximation de la racine ?

CORRECTION DE L'EXERCICE 1.4. On cherche les zéros de la fonction $f(x) = x^2 - 2$:

- Méthode de la dichotomie : en partant de $I_0 = [a,b]$, la méthode de la dichotomie produit une suite de sous-intervalles $I_k = [a_k, b_k]$ avec $I_{k+1} \subset I_k$ et tels que $f(a_k)f(b_k) < 0$. Plus précisément
  - on pose $a_0 = a$, $b_0 = b$, $x_0 = \frac{a_0+b_0}{2}$,
  - pour $k \geq 0$
    - si $f(a_k)f(x_k) < 0$ on pose $a_{k+1} = a_k$, $b_{k+1} = x_k$ sinon on pose $a_{k+1} = x_k$, $b_{k+1} = b_k$



(a) Méthode de la dichotomie.

(b) Méthode de NEWTON.

FIGURE 1.3.: Approximation du zéro de la fonction $f(x) = x^2 - 2$.

⋆ et on pose $x_{k+1} = \frac{a_{k+1}+b_{k+1}}{2}$.

Voir la figure 1.3a.

⋆ Méthode de NEWTON :

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} = x_k - \frac{x_k^2 - 2}{2x_k} = \frac{1}{2}x_k + \frac{1}{x_k}.$$

Voir la figure 1.3b.

Donc on a le tableau suivant

	x_0	x_1	x_2	x_3
Dichotomie	2	$\frac{3}{2} = 1{,}5$	$\frac{5}{4} = 1{,}25$	$\frac{11}{8} = 1{,}375$
Newton	2	$\frac{3}{2} = 1{,}5$	$\frac{17}{12} = 1{,}41\bar{6}$	$\frac{17}{24} + \frac{12}{17} \simeq 1{,}4142156$

On rappelle qu'avec la méthode de la dichotomie, les itération s'achèvent à la m-ème étape quand $|x_m - \widehat{x}| \leq |I_m| < \varepsilon$, où ε est une tolérance fixée et $|I_m|$ désigne la longueur de l'intervalle I_m. Clairement $I_k = \frac{b-a}{2^k}$, donc pour avoir $|x_m - \widehat{x}| < \varepsilon$ on doit prendre

$$m \geq \log_2\left(\frac{b-a}{\varepsilon}\right).$$

Améliorer d'un ordre de grandeur la précision de l'approximation de la racine signifie avoir

$$|x_k - \widehat{x}| = \frac{|x_j - \widehat{x}|}{10}$$

donc on doit effectuer $k - j = \log_2(10) \simeq 3{,}3$ itérations de dichotomie.

Exercice 1.5

1. Donner la suite définissant la méthode de NEWTON pour la recherche d'un zéro de fonction. Justifier l'expression de la suite.
2. Écrire l'algorithme pour une convergence à 10^{-6} près.
3. Déterminer l'ordre de convergence minimale de cette suite.

CORRECTION DE L'EXERCICE 1.5.

1. Supposons $f \in \mathscr{C}^1$ et $f'(\widehat{x}) \neq 0$ (c'est-à-dire $\widehat{x}$ est une racine simple de f). La méthode de NEWTON revient à calculer le zéro de f en remplaçant localement f par sa tangente : en partant de l'équation de la tangente à la courbe $(x, f(x))$ au point x_k

$$y(x) = f(x_k) + f'(x_k)(x - x_k)$$

et en faisant comme si x_{k+1} vérifiait $y(x_{k+1}) = 0$, on obtient

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}.$$

Étant donné une valeur initiale $x^{(0)}$, cette formule permet de construire une suite x_k.

2. Algorithme pour une convergence à $\varepsilon = 10^{-6}$:

Require: x_0, $x \mapsto f(x)$
while $|x_{k+1} - x_k| > 10^{-6}$ **do**
$x_{k+1} \leftarrow x_k - \frac{f(x_k)}{f'(x_k)}$
end while

3. La relation précédent peut être mise sous la forme d'une itération de point fixe $x_{k+1} = g(x_k)$ avec

$$g(x) = x - \frac{f(x)}{f'(x)}.$$

Si $\widehat{x}$ est racine simple, c'est-à-dire si $f'(\widehat{x}) \neq 0$, on trouve $g'(\widehat{x}) = 0$ et $g''(\widehat{x}) = \frac{f''(\widehat{x})}{f'(\widehat{x})}$: la méthode de NEWTON est donc d'ordre 2. Si la racine $\widehat{x}$ est de multiplicité $m > 1$, alors $g'(\widehat{x}) = 1 - \frac{1}{m}$ et la méthode n'est que d'ordre 1. Si la valeur de m est connue à priori, on peut retrouver la convergence quadratique de la méthode de NEWTON en modifiant la méthode comme suit :

$$x_{k+1} = x_k - m\frac{f(x_k)}{f'(x_k)}.$$

Exercice 1.6

On veut calculer le zéro de la fonction

$$f(x) = x^2 - 2$$

dans l'intervalle $[0;2]$.

1. On applique la méthode de LAGRANGE : écrire l'algorithme et l'utiliser pour remplir le tableau (on s'arrêtera au plus petit k qui vérifie $|f(x_k)| < 10^{-4}$).

k	a_k	x_k	b_k	signe de $f(a_k)$	$f(x_k)$	signe de $f(b_k)$	$\lvert x_k - \sqrt{2}\rvert$
0	0.00000	1.00000	2.00000	–	-1.00000	+	0.41421
1							
⋮							

2. On applique la méthode de NEWTON : écrire l'algorithme et l'utiliser pour remplir le tableau (on s'arrêtera au plus petit k qui vérifie $|f(x_k)| < 10^{-4}$). Le point de départ x_0 est donné.

k	x_k	$f(x_k)$	$\lvert x_k - \sqrt{2}\rvert$
0	1.00000		
1			
⋮			

CORRECTION DE L'EXERCICE 1.6.

1. En partant de $I_0 = [a, b]$, la méthode de LAGRANGE produit une suite de sous-intervalles $I_k = [a_k, b_k]$, $k \geq 0$, avec $I_k \subset I_{k-1}$, $k \geq 1$, et tels que $f(a_k)f(b_k) < 0$. Dans notre cas on a

$k \leftarrow 0$
$a_k \leftarrow 0$
$b_k \leftarrow 2$
$x_k \leftarrow a_k$
while $|x_k^2 - 2| > 0.0001$ **do**
$x_k \leftarrow \frac{a_k b_k + 2}{a_k + b_k}$
if $(a_k^2 - 2)(x_k^2 - 2) < 0$ **then**
$a_{k+1} \leftarrow a_k$
$b_{k+1} \leftarrow x_k$
else
$a_{k+1} \leftarrow x_k$
$b_{k+1} \leftarrow b_k$
end if

$k \leftarrow k+1$
end while

k	a_k	x_k	b_k	signe de $f(a_k)$	$\lvert f(x_k)\rvert$	signe de $f(b_k)$	$\lvert x_k-\sqrt{2}\rvert$
0	0.00000	1.00000	2.00000	–	|-1.00000|>0.0001	+	0.41421
1	1.00000	1.33333	2.00000	–	|-0.22222|>0.0001	+	0.08088
2	1.33333	1.40000	2.00000	–	|-0.04000|>0.0001	+	0.01421
3	1.40000	1.41176	2.00000	–	|-0.00692|>0.0001	+	0.00245
4	1.41176	1.41379	2.00000	–	|-0.00119|>0.0001	+	0.00042
5	1.41379	1.41414	2.00000	–	|-0.00020|>0.0001	+	0.00007
6	1.41414	1.41420	2.00000	–	|-0.00004|<0.0001	+	0.00001

2. La méthode de NEWTON est une méthode de point fixe avec fonction d'itération $\phi(x) = x - \frac{f(x)}{f'(x)}$ ce qui donne l'algorithme suivant :

$k \leftarrow 0$
$x_k \leftarrow 1.00000$
while $|x_k^2 - 2| > 10^{-4}$ **do**
$x_{k+1} \leftarrow \frac{x_k}{2} + \frac{1}{x_k}$
$k \leftarrow k+1$
end while

k	x_k	$\lvert f(x_k)\rvert$	$\lvert x_k-\sqrt{2}\rvert$
0	1.00000	|-1.00000|>0.0001	0.41421
1	1.50000	|0.25000|>0.0001	0.08579
2	1.41667	|0.00695|>0.0001	0.00246
3	1.41422	|0.00002|<0.0001	0.00001

Exercice 1.7 *Évolution d'un capital*

On investit un capital $C_0 > 0$. Le placement a un taux de 5% par an et des frais de gestion fixes de 50 euros qui sont prélevés chaque année.

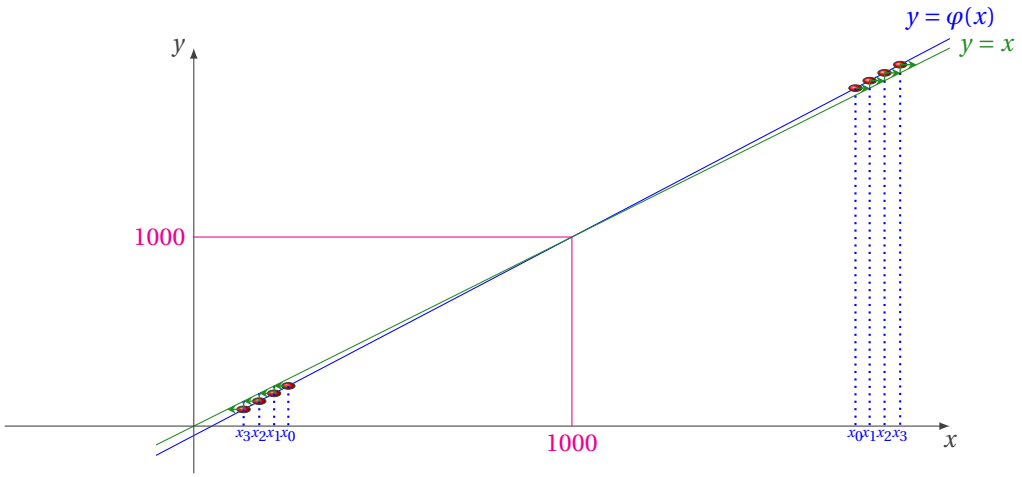
1. Décrire la suite récurrente qui décrit l'évolution du placement.
2. Donner les points fixes du système et indiquer s'ils sont attractifs ou répulsifs.
3. Étudier l'évolution du capital au fil des ans selon la valeur de C_0.

CORRECTION DE L'EXERCICE 1.7.

1. Notons u_n le capital au début de la n-ième année, alors on a

$$\begin{cases} u_0 = C_0, \\ u_{n+1} = (1+5\%)\,u_n - 50 = 1.05\,u_n - 50, \quad \forall n \in \mathbb{N}. \end{cases}$$

2. Il s'agit d'une suite récurrente définie par $u_{n+1} = \varphi(u_n)$ avec $\varphi(x) = 1.05x - 50$. On a $\varphi(x) = x$ ssi $x = 1000$: l'unique point fixe du système est $x = 1000$. Comme $\varphi'(1000) = 1.05 > 1$, il s'agit donc d'un point fixe répulsif.
3. Évolution du capital au fil des ans selon la valeur de C_0 :
   - ⋆ si $C_0 > 1000$ alors $u_n \to +\infty$,
   - ⋆ si $C_0 = 1000$ alors $u_n = 1000$ pour tout $n \in \mathbb{N}$,
   - ⋆ si $C_0 < 1000$ alors $u_n \to -\infty$.



Exercice 1.8 *Détermination des points fixes attractifs et répulsifs*

On considère des systèmes dynamiques donnés par la loi d'évolution $x \mapsto \varphi(x)$. Dans chaque cas déterminer les points fixes et leur nature (sont-ils attractifs ou répulsifs ?). Tracer le graphe de φ et quelques points de la suite.

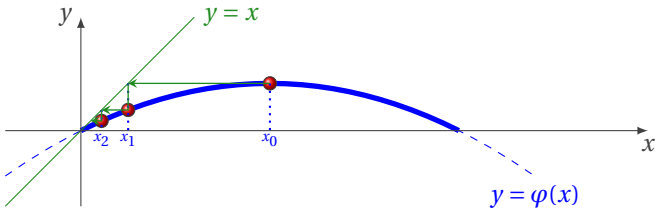
① $\varphi\colon [0;1] \to \mathbb{R}$, $x \mapsto \frac{1}{2}x(1-x)$

② $\varphi\colon [0;1] \to \mathbb{R}$, $x \mapsto \frac{1}{2}x(1+x)$

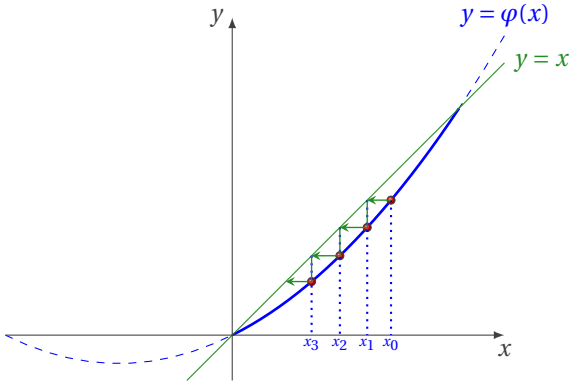
③ $\varphi\colon \mathbb{R} \to \mathbb{R}$, $x \mapsto \frac{1}{2}x(1+x^2)$

CORRECTION DE L'EXERCICE 1.8.

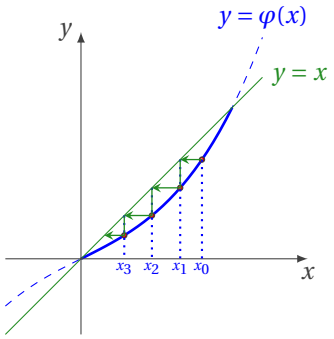
① $\varphi(x) = \frac{1}{2}x(1-x) = \frac{-x^2+x}{2}$ et $\varphi'(x) = -x + \frac{1}{2}$
- ⋆ Point fixe (dans $[0;1]$) : $\ell = 0$.
- ⋆ Nature : $\varphi'(\ell) = \frac{1}{2} \in]-1;1[$ donc ℓ est attractif.



② $\varphi(x) = \frac{1}{2}x(1+x) = \frac{x^2+x}{2}$ et $\varphi'(x) = x + \frac{1}{2}$
- ⋆ Points fixes (dans $[0;1]$) : $\ell_1 = 0$ et $\ell_2 = 1$.
- ⋆ Nature : $\varphi'(\ell_1) = \frac{1}{2} \in]-1;1[$ donc ℓ_1 est attractif, $\varphi'(\ell_2) = \frac{3}{2} > 1$ donc ℓ_2 est répulsif.



③ $\varphi(x) = \frac{1}{2}x(1+x^2) = \frac{x^3+x}{2}$ et $\varphi'(x) = \frac{3x^2+1}{2}$
- ⋆ Points fixes (dans $[0;1]$) : $\ell_1 = 0$ et $\ell_2 = 1$.
- ⋆ Nature : $\varphi'(\ell_1) = \frac{1}{2} \in]-1;1[$ donc ℓ_1 est attractif, $\varphi'(\ell_2) = 2 > 1$ donc ℓ_2 est répulsif.



Exercice 1.9 *Points fixes où la dérivée vaut* 1

On considère les systèmes dynamiques sur $[0;1]$ donnés par les lois d'évolution suivantes :

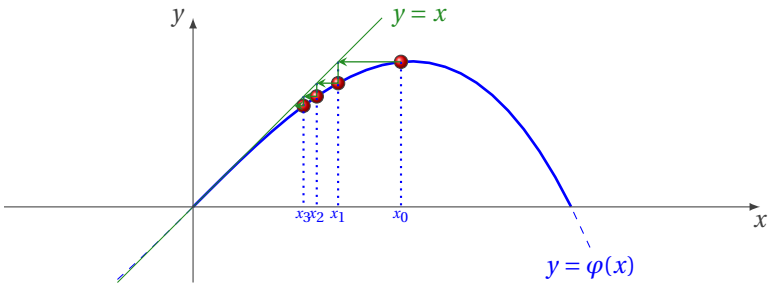
1. $\varphi(x) = x - x^3$
2. $\varphi(x) = x + x^3$

3. $\varphi(x) = x + x^2$

Dans chacun des cas, montrer que 0 est un point fixe du système et que la dérivée de la loi d'évolution en 0 est égale à 1. Dans chacun des cas, tracer le graphe de la loi d'évolution et quelques orbites (*i.e.* quelques points de la suite). Dans quel cas le point fixe 0 est-il attractif ? Répulsif ?

CORRECTION DE L'EXERCICE 1.9.

1. $\varphi(x) = x - x^3 = x(1-x)(1+x)$ et $\varphi'(x) = 1 - 3x^2 = (1-\sqrt{3}x)(1+\sqrt{3}x)$
   - ⋆ Points fixes (dans $[0;1]$) : $\ell = 0$.
   - ⋆ Nature : $\varphi'(\ell) = 1$ donc on ne peut pas établir directement la nature du point fixe.



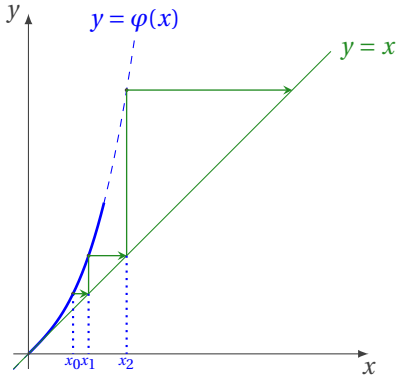
On voit qu'il s'agit d'un point fixe attractif. Pour le démontrer il suffit d'étudier directement la suite définie par récurrence

$$\begin{cases} u_0 \in [0;1] \\ u_{n+1} = \varphi(u_n), \quad \forall n \in \mathbb{N}. \end{cases}$$

   - ⋆ Si la suite converge, elle converge vers ℓ.
   - ⋆ On vérifie facilement que $\varphi([0;1]) \subset [0;1]$ ainsi $u_n \in [0;1]$ pour tout $n \in \mathbb{N}$.
   - ⋆ $u_{n+1} - u_n = -u_n^3 < 0$ pour tout $n \in \mathbb{N}$ donc la suite est monotone décroissante.

   Conclusion : $u_n \to 0$.

2. $\varphi(x) = x + x^3 = x(1+x^2)$ et $\varphi'(x) = 1 + 3x^2$
   - ⋆ Points fixes (dans $[0;1]$) : $\ell = 0$.
   - ⋆ Nature : $\varphi'(\ell) = 1$ donc on ne peut pas établir directement la nature du point fixe.



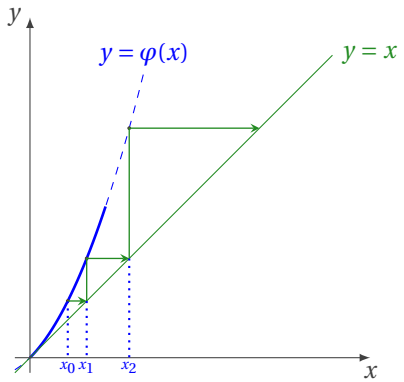
On voit qu'il s'agit d'un point fixe répulsif. Pour le démontrer il suffit d'étudier directement la suite définie par récurrence suivante (on considère φ définie sur $\mathbb{R}_+$)

$$\begin{cases} u_0 \in [0;1] \\ u_{n+1} = \varphi(u_n), \quad \forall n \in \mathbb{N}. \end{cases}$$

   - ⋆ Si la suite converge, elle converge vers ℓ.
   - ⋆ On vérifie facilement que $\varphi(x) \geq 0$ pour tout $x \in \mathbb{R}_+$ ainsi $u_n \geq 0$ pour tout $n \in \mathbb{N}$.
   - ⋆ $u_{n+1} - u_n = u_n^3 > 0$ pour tout $n \in \mathbb{N}$ donc la suite est monotone croissante.

   Conclusion : $u_n \to +\infty$.

3. $\varphi(x) = x + x^2 = x(1+x)$ et $\varphi'(x) = 1 + 2x$
   - ⋆ Points fixes (dans $[0;1]$) : $\ell = 0$.
   - ⋆ Nature : $\varphi'(\ell) = 1$ donc on ne peut pas établir directement la nature du point fixe.



On voit qu'il s'agit d'un point fixe répulsif. Pour le démontrer il suffit d'étudier directement la suite définie par récurrence suivante (on considère φ définie sur $\mathbb{R}_+$)

$$\begin{cases} u_0 \in [0;1] \\ u_{n+1} = \varphi(u_n), \quad \forall n \in \mathbb{N}. \end{cases}$$

- ⋆ Si la suite converge, elle converge vers ℓ.
- ⋆ On vérifie facilement que $\varphi(x) \geq 0$ pour tout $x \in \mathbb{R}_+$ ainsi $u_n \geq 0$ pour tout $n \in \mathbb{N}$.
- ⋆ $u_{n+1} - u_n = u_n > 0$ pour tout $n \in \mathbb{N}$ donc la suite est monotone croissante.

Conclusion : $u_n \to +\infty$.

Exercice 1.10

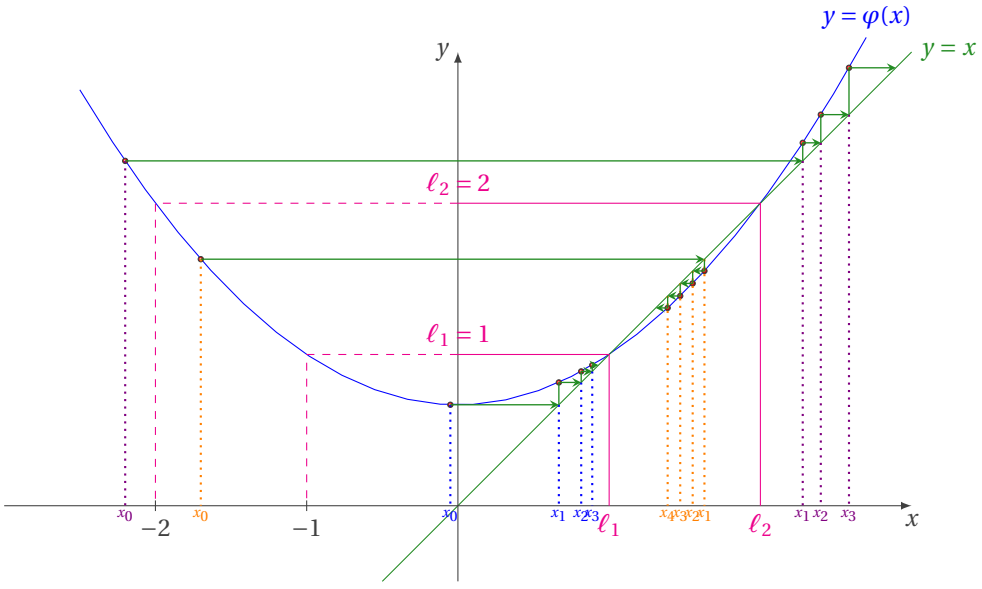
Pour approcher les racines réelles de la fonction $f : \mathbb{R} \to \mathbb{R}$ définie par $f(x) = (x^2 - 3x + 2)e^x$ on veut utiliser la méthode de point fixe suivante :

$$\begin{cases} x_0 \text{ donné,} \\ x_{n+1} = \varphi(x_n) \quad \text{pour tout } n \in \mathbb{N} \end{cases} \qquad \text{où} \qquad \varphi(x) = \frac{x^2+2}{3}.$$

1. Montrer qu'il existe deux racines réelles $\ell_1 < \ell_2$ de f et les calculer.
2. Faire l'*étude graphique* de la convergence de la méthode de point fixe et montrer que
   - ⋆ si $x_0 \in]-2;2[$ alors la suite converge vers ℓ_1,
   - ⋆ si $x_0 = \pm 2$ alors $x_n = \ell_2$ pour tout $n \in \mathbb{N}^*$,
   - ⋆ si $x_0 < -2$ ou $x_0 > 2$ alors la suite diverge vers $+\infty$.
3. Notons $[a;b]$ l'intervalle maximale contenant ℓ_1 pour lequel le théorème de point fixe s'applique. Calculer a et b et expliquer pourquoi la suite converge vers ℓ_1 même si $x_0 \in]-2;2[\setminus [a;b]$.

CORRECTION DE L'EXERCICE 1.10.

1. $f(x) = (x-1)(x-2)e^x$ donc f admet deux uniques racines réelles $\ell_1 = 1$ et $\ell_2 = 2$.
2. Comparons φ, qui est une parabole convexe de sommet $(0, 2/3)$, à l'identité :



L'étude graphique montre que la suite converge quel que soit $x_0 \in [-2;2]$. Plus précisément, on voit que

- ⋆ si $x_0 = -2$ alors $x_2 = 2$ pour tout $n \in \mathbb{N}^*$,
- ⋆ si $x_0 \in]-2;-1[$ alors $x_n \searrow 1$ pour tout $n \in \mathbb{N}^*$,
- ⋆ si $x_0 = -1$ alors $x_n = 1$ pour tout $n \in \mathbb{N}^*$,
- ⋆ si $x_0 \in]-1;1[$ alors $x_n \nearrow 1$ pour tout $n \in \mathbb{N}^*$,
- ⋆ si $x_0 = 1$ alors $x_n = 1$ pour tout $n \in \mathbb{N}$,
- ⋆ si $x_0 \in]1;2[$ alors $x_n \searrow 1$ pour tout $n \in \mathbb{N}$,
- ⋆ si $x_0 = 2$ alors $x_2 = 2$ pour tout $n \in \mathbb{N}$,
- ⋆ si $x_0 > 2$ alors $x_n \nearrow +\infty$.

3. L'intervalle maximale pour appliquer le théorème de point fixe est défini comme le plus grand intervalle pour lequel $|\varphi'(x)| < 1$, *i.e.* l'intervalle $\left[-\frac{3}{2};\frac{3}{2}\right]$. On remarque que $\left[-\frac{3}{2};\frac{3}{2}\right] \subset [-2;2]$. Or, si $x_0 \in]-2;-3/2]$ ou $x_0 \in [3/2;2[$, alors $x_1 = \varphi(x_0) < |x_0| \in]1;2[$ (car $\varphi(x) < x$ lorsque $x \in]1;2[$) et on montre que la suite $(x_n)_{n\in\mathbb{N}}$ est monotone décroissante et minorée par ℓ_1 donc convergente. Comme l'unique limite possible est ℓ_1 on conclut que $x_n \searrow 1$.

Exercice 1.11

Considérons l'équation $x(1+e^x) = e^x$.

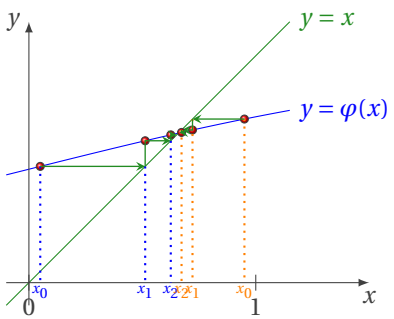
1. Montrer que cette équation admet une unique solution réelle ℓ dans $[0;1]$.
2. Écrire la méthode de NEWTON pour approcher la solution ℓ.
3. Proposer une autre itération de point fixe pour approcher ℓ. Montrer analytiquement que cette itération converge vers ℓ pour tout $x_0 \in [0;1]$ et faire l'étude graphique de la convergence.

CORRECTION DE L'EXERCICE 1.11.

1. Soit $f : \mathbb{R} \to \mathbb{R}$ définie par $f(x) = x(1+e^x) - e^x$. $f(x) = 0$ si et seulement si x est solution de l'équation donnée. f est de classe $\mathscr{C}^\infty(\mathbb{R})$, $f(0) = -1 < 0$ et $f(1) = 1 > 0$ donc, d'après le théorème des valeurs intermédiaire, la fonction f admet au moins une racine sur $[0;1]$. De plus, f est monotone sur $[0;1]$ (car $f'(x) = 1 + xe^x > 0$ pour tout $x \in [0;1]$), donc cette racine est unique.
2. Méthode de NEWTON :
$$\begin{cases} x_0 \in [0;1] \\ x_{n+1} = x_n - \frac{x_n(1+e^{x_n}) - e^{x_n}}{1+x_n e^{x_n}} = \frac{x_n^2 - x_n + 1}{1+x_n e^{x_n}} e^{x_n} \end{cases}$$
3. On considère l'itération
$$\begin{cases} x_0 \in [0;1] \\ x_{n+1} = \varphi(x_n) \end{cases} \qquad \text{avec} \qquad \varphi(x) = \frac{e^x}{1+e^x}.$$
   - ⋆ φ est une fonction croissante sur $\mathbb{R}$ car $\varphi'(x) = \frac{e^x}{(1+e^x)^2} > 0$;
   - ⋆ Montrons que $\varphi([0;1]) \subset [0;1]$:
$$\begin{cases} x \in [0;1] \\ \varphi \text{ continue et croissante} \end{cases} \implies \varphi(x) \in [\varphi(0);\varphi(1)] = \left[\frac{1}{2};\frac{e}{1+e}\right] \subset [0;1];$$
   - ⋆ φ est contractante stricte car $|\varphi'(x)| < 1$ pour tout $x \in \mathbb{R}$.

   D'après le théorème de convergence globale des itérations de point fixe, l'itération proposée converge pour tout $x \in [0;1]$. De plus, $0 < \varphi'(x) < 1$ pour tout $x \in \mathbb{R}$, donc la convergence est monotone et du premier ordre.

   Comparons φ à l'identité :



Exercice 1.12

Pour calculer les racines de la fonction $f(x) = x^3 - x^2 + 8x - 8$ on utilise 4 méthodes de point fixe différentes décrites par les fonctions d'itération suivantes :

$$\varphi_1(x) = -x^3 + x^2 - 7x + 8, \qquad \varphi_2(x) = \frac{8 - x^3}{8 - x}, \qquad \varphi_3(x) = -\frac{1}{10}x^3 + \frac{1}{10}x^2 + \frac{1}{5}x + \frac{4}{5}, \qquad \varphi_4(x) = \frac{2x^3 - x^2 + 8}{3x^2 - 2x + 8}.$$

① Montrer que $\ell = 1$ est l'unique racine réelle de f.

② Étudier la convergence locale des méthodes de point fixe $x_{k+1} = \varphi_i(x_k)$ pour $i = 1, \dots, 4$.

Correction de l'exercice 1.12. Les fonctions φ_i sont de classe $\mathscr{C}^\infty$ au voisinage de ℓ. De plus, on remarque que $f(x) = (x-1)(x^2+8)$, donc l'unique racine réelle de f est $\ell = 1$.

1. $\varphi_1'(x) = -3x^2 + 2x - 7$ et $\varphi_1'(1) = -8$: la suite diverge en oscillant ;
2. $\varphi_2'(x) = \frac{-3x^3(8-x)+(8-x^3)}{(8-x)^2}$ et $\varphi_2'(1) = -\frac{14}{49}$: la suite converge de façon oscillante ;
3. $\varphi_3'(x) = -\frac{3}{10}x^2 + \frac{1}{5}x + \frac{1}{5}$ et $\varphi_3'(1) = \frac{4}{10}$: la suite converge de façon monotone ;
4. $\varphi_4'(x) = \frac{(6x^2-2x)(3x^2-2x+8)-(2x^3-x^2+8)(6x-2)}{(3x^2-2x+8)^2}$ et $\varphi_4'(1) = 0$: la suite converge à l'ordre au moins 2.

Dans le tableau suivant sont reportées les suites des itérées obtenues par ces quatre méthodes.

	Méthode φ_3	Méthode φ_4	Méthode φ_1	Méthode φ_2
x_0	0.5000000000000000	0.500000000000000	0.5000000000000000	0.5000000000000000
x_1	0.9125000000000001	1.032258064516129	4.625000000000000	1.050000000000000
x_2	0.9897857421875000	1.000235245684712	−101.9160156250000	0.9845143884892086
x_3	0.9989578145726552	1.000000012299503	$1.069697123778202 \times 10^6$	1.004312677086027
x_4	0.9998955643403695	1.000000000000000	$-1.224001861234915 \times 10^{18}$	0.9987590594698483
x_5	0.9999895542527895	1.000000000000000	$1.833775789385161 \times 10^{54}$	1.000353832012369
x_6	0.9999989554034564	1.000000000000000	$-6.166499545700052 \times 10^{162}$	0.9998988463640411

Exercice 1.13

Pour approcher les racines réelles de la fonction $f : \mathbb{R} \to \mathbb{R}$ définie par $f(x) = x - e^{-(1+x)}$ on utilise quatre méthodes de point fixe :

$$\begin{cases} x_0 \text{ donné,} \\ x_{n+1} = \varphi_i(x_n) \quad \text{pour tout } n \in \mathbb{N} \end{cases}$$

où

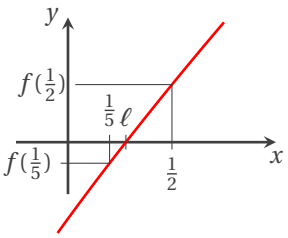
$$\varphi_1(x) = e^{-(1+x)}, \qquad \varphi_2(x) = x^2 e^{(1+x)}, \qquad \varphi_3(x) = -1 - \ln(x), \qquad \varphi_4(x) = \frac{1+x}{1+e^{(1+x)}}.$$

1. Montrer qu'il existe une unique racine réelle ℓ de f. Montrer que $\ell \in \left]\frac{1}{5}; \frac{1}{2}\right[$.
2. Montrer que les quatre méthodes de point fixe sont consistantes avec la recherche du zéro de f, *i.e.* montrer que pour $x \in \left]\frac{1}{5}; \frac{1}{2}\right[$ on a
$$\varphi_i(x) = x \qquad \Longleftrightarrow \qquad f(x) = 0 \qquad i = 1, 2, 3, 4.$$
3. Étudier la convergence locale des trois méthodes de point fixe (*i.e.* vérifier si on peut appliquer le théorème d'Ostrowski) et, si elles convergent, donner l'ordre de convergence.
   Attention : on ne demande pas la convergence globale, autrement dit on ne demande pas de vérifier si on peut appliquer le théorème de point fixe sur l'intervalle $\left]\frac{1}{5}; \frac{1}{2}\right[$ mais de vérifier s'il existe un voisinage de ℓ tel que pour tout x_0 assez proche de ℓ la méthode converge.
4. Pour la première méthode, faire l'étude graphique de la convergence pour tout $x_0 \in \mathbb{R}$ et établir analytiquement pour quelles valeurs de x_0 la suite converge (*i.e.* trouver des intervalles pour lesquels le théorème de point fixe s'applique).

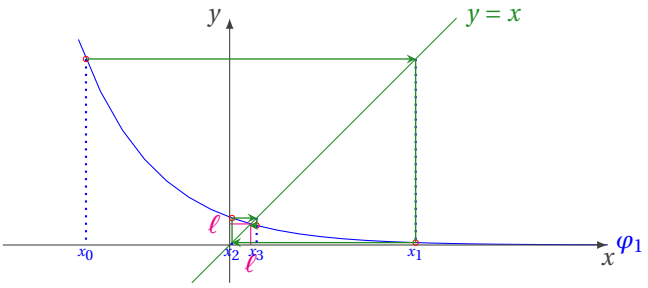
Correction de l'exercice 1.13.

1. On étudie brièvement f :

- $\mathscr{D}_f = \mathbb{R}$ et f de classe $\mathscr{C}^\infty(\mathbb{R})$,
- $\lim_{x\to\pm\infty} f(x) = \pm\infty$,
- $f'(x) = 1 + e^{-(1+x)}$, f croissante pour tout $x \in \mathbb{R}$,
- $f(\frac{1}{5}) = \frac{1}{5} - \frac{1}{e^{6/5}} < 0$, $f(\frac{1}{2}) = \frac{1}{2} - \frac{1}{e^{3/2}} > 0$,
  $f'(x) > 0$ pour tout $x \in \mathbb{R}$:
  il existe une unique racine $\ell \in [1/5; 1/2]$ de f.



2. 2.1. $\varphi_1(x) = x$ ssi $x = e^{-(1+x)}$ ssi $f(x) = 0$.
   2.2. $\varphi_2(x) = x$ ssi $x = x^2 e^{(1+x)}$ ssi $x = 0$ ou $1 = xe^{(1+x)}$ ssi $x = 0$ ou $\varphi_1(x) = x$ ssi $x = 0$ ou $f(x) = 0$.
   2.3. φ_3 n'est définie que pour $x > 0$ et $\varphi_3(x) = x$ ssi $x = -1 - \ln(x)$ ssi $-(1+x) = \ln(x)$ ssi $\varphi_1(x) = x$ ssi $f(x) = 0$.
   2.4. $\varphi_4(x) = x$ ssi $x = \frac{1+x}{1+e^{(1+x)}}$ ssi $xe^{(1+x)} = 1$ ssi $\varphi_1(x) = x$ ssi $f(x) = 0$.
3. 3.1. $|\varphi_1'(\ell)| = e^{-(1+\ell)} = \ell \in [1/5; 1/2]$ donc la suite converge à l'ordre 1 pour x_0 suffisamment proche de ℓ.
   3.2. Sachant que $\ell > 1/5$ on a $|\varphi_2'(\ell)| = (2+\ell)\ell e^{(1+\ell)} > \frac{11}{25} e^{6/5} > 1$ donc la suite ne converge pas.
   3.3. Sachant que $\ell < 1/2$ on a $|\varphi_3'(\ell)| = 1/\ell > 2$ donc la suite ne converge pas.
   3.4. $\varphi_4'(\ell) = \frac{1-\ell e^{(1+\ell)}}{(1+e^{(1+\ell)})^2} = \varphi_4(\ell)\frac{1}{1+e^{(1+\ell)}} - \ell = \frac{\ell}{1+e^{(1+\ell)}} - \ell = \varphi_4(\ell) - \ell = \ell - \ell = 0$ donc la suite converge au moins à l'ordre 2 pour x_0 suffisamment proche de ℓ.
4. Le graphe de φ_1 s'obtient à partir de celui de e^x en faisant une symétrie par rapport à l'axe des ordonnées (ce qui donne le graphe de e^{-x}) suivie de la translation vers la gauche d'une unité. Si on n'a pas observé ce comportement, on peut étudier brièvement φ_1 pour pouvoir tracer son graphe et le comparer à l'identité :
   - $\mathscr{D}_{\varphi_1} = \mathbb{R}$ et φ_1 de classe $\mathscr{C}^\infty(\mathbb{R})$,
   - $\varphi_1(x) > 0$ pour tout $x \in \mathbb{R}$,
   - $\lim_{x\to-\infty} \varphi_1(x) = +\infty$,
   - $\lim_{x\to+\infty} \varphi_1(x) = 0^+$,
   - $\varphi_1'(x) = -\varphi_1(x) < 0$ pour tout $x \in \mathbb{R}$: φ_1 décroissante pour tout $x \in \mathbb{R}$,
   - $\varphi_1(x) = x$ ssi $f(x) = 0$ donc il existe un unique $\ell \in [0.2; 0.5]$ tel que $\varphi_1(\ell) = \ell$,
   - $\varphi_1''(x) = \varphi_1(x)$: φ_1 convexe pour tout $x \in \mathbb{R}$.



L'étude graphique suggère que la suite converge quel que soit $x_0 \in \mathbb{R}$. Mieux encore, on voit que $x_n > 0$ pour tout $n \in \mathbb{N}^*$.

Pour prouver cela on vérifie si

$$|\varphi_1'(x)| < 1, \qquad \forall x \in \mathbb{R}.$$

Comme $\varphi_1'(x) = -\varphi_1(x)$, on a $|\varphi_1'(x)| < 1$ ssi $x > -1$ donc la condition de contraction stricte n'est pas satisfaite. Voyons si on peut appliquer le théorème au moins pour $x > -1$. On a $\varphi_1(]-1;+\infty[) =]0;1[\subset]-1;+\infty[$ donc φ_1 est stable sur $]-1;+\infty[$ et contractante stricte. Le théorème de point fixe permet alors de conclure que la méthode de point fixe converge pou tout $x_0 \in]-1;+\infty[$.

Que peut-on dire si $x_0 \leq -1$? Dans ce cas le théorème de point fixe ne s'applique pas. Cependant on a $x_1 = \varphi_1(x_0) \in]0;+\infty[\subset]-1;+\infty[$ et le théorème de point fixe s'applique à partir de x_1.

On conclut que la méthode 1 converge vers l'unique point fixe de φ_1 pour tout $x_0 \in \mathbb{R}$.

Exercice 1.14

Pour approcher les racines réelles de la fonction

$$f : \mathbb{R} \to \mathbb{R}$$
$$x \mapsto x^3 - x^2 - 1$$

on utilise trois méthodes de point fixe :

$$\begin{cases} x_0 \text{ donné,} \\ x_{n+1} = \varphi_i(x_n) \quad \text{pour tout } n \in \mathbb{N} \end{cases}$$

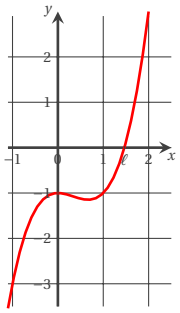
où

$$\varphi_1(x) = x^3 - x^2 + x - 1, \qquad \varphi_2(x) = \sqrt[3]{x^2+1}, \qquad \varphi_3(x) = \frac{2x^3 - x^2 + 1}{x(3x-2)}.$$

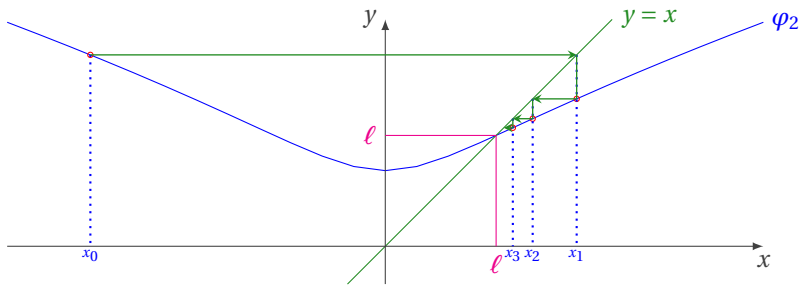
1. Montrer qu'il existe une unique racine réelle ℓ de f. Montrer que $\ell \in [1;2]$.
2. Étudier la convergence locale des trois méthodes de point fixe et, si elles convergent, donner l'ordre de convergence.
3. Pour la deuxième méthode, faire l'étude graphique de la convergence globale et établir analytiquement pour quelles valeurs de x_0 la suite converge.

CORRECTION DE L'EXERCICE 1.14.

1. On étudie brièvement f :
   - $\mathscr{D}_f = \mathbb{R}$ et f de classe $\mathscr{C}^\infty(\mathbb{R})$,
   - $\lim_{x\to\pm\infty} f(x) = \pm\infty$,
   - $f'(x) = x(3x-2)$,
   - f croissante pour $x<0$ et $x>3/2$, décroissante pour $0<x<2/3$,
   - maximum local en $x=0$ et $f(0) = -1 < 0$, minimum local en $x = 2/3$ et $f(2/3) = -31/27 < 0$,
   - $f(1) = -1$, $f(2) = 3$ et $f'(x) = x(3x-2) > 0$ pour tout $x \in [1;2]$: il existe une unique racine $\ell \in [1;2]$ de f.



2. Vérifions si on peut appliquer le théorème d'OSTROWSKI (attention : il ne s'agit pas de vérifier si on peut appliquer le théorème de point fixe sur l'intervalle $[1;2]$ mais de vérifier s'il existe un voisinage de ℓ tel que pour tout x_0 assez proche de ℓ la méthode converge).
   2.1. $\varphi_1(x) = x \iff f(x) = 0$: la méthode est consistante. Comme $\varphi_1'(\ell) = 3\ell^2 - 2\ell + 1 = \ell(3\ell-2) + 1 > \ell + 1 > 1$, la suite ne converge pas.
   2.2. $\varphi_2(x) = x \iff f(x) = 0$: la méthode est consistante. Sachant que $\ell \in [1;2]$ on a $|\varphi_2'(\ell)| = \frac{2\ell}{3\sqrt[3]{(\ell^2+1)^2}} < \frac{4}{3\sqrt[3]{2^2}} = \frac{2^{4/3}}{3} < 1$ donc la suite converge à l'ordre 1 pour x_0 suffisamment proche de ℓ.
   2.3. $\varphi_3(x) = x \iff f(x) = 0$: la méthode est consistante. $\varphi_3'(\ell) = \frac{(6\ell^2-2\ell)(3\ell^2-2\ell)-(2\ell^3-\ell^2+1)(6\ell-2)}{\ell^2(3\ell-2)^2} = \frac{2\ell(\ell-2)}{\ell(3\ell-2)} - \varphi_3(\ell)\frac{6\ell-2}{\ell(3\ell-2)} = \frac{2\ell(\ell-2)}{\ell(3\ell-2)} - \ell\frac{6\ell-2}{\ell(3\ell-2)} = 0$ donc la suite converge au moins à l'ordre 2 pour x_0 suffisamment proche de ℓ. (Il s'agit en effet de la méthode de NEWTON).
3. Pour faire l'étude globale de la convergence on essaye d'appliquer le théorème de point fixe sur l'intervalle $[1;2]$ mais ici on peut facilement étendre l'étude à $\mathbb{R}$.
   On étudie donc brièvement φ_2 pour pouvoir tracer son graphe et le comparer à l'identité :
   - $\mathscr{D}_{\varphi_2} = \mathbb{R}$ et φ_2 de classe $\mathscr{C}^\infty(\mathbb{R})$,
   - $\lim_{x\to\pm\infty} \varphi_2(x) = +\infty$,
   - $\varphi_2'(x) = \frac{2x}{3\sqrt[3]{(x^2+1)^2}}$,
   - φ_2 croissante pour $x>0$, décroissante pour $x<0$, minimum locale en $x=0$ et $\varphi_2(0) = 1$,
   - $\varphi_2(x) = x$ ssi $f(x) = 0$ donc il existe un unique $\ell \in [1;2]$ tel que $\varphi_2(\ell) = \ell$,
   - $\varphi_2''(x) = \frac{-2(x^2-3)}{9\sqrt[3]{(x^2+1)^5}}$, φ_2 convexe pour $-\sqrt{3} < x < \sqrt{3}$, φ_2 concave pour $x < -\sqrt{3}$ et $x > \sqrt{3}$.



L'étude graphique suggère que la suite converge quel que soit $x_0 \in \mathbb{R}$. Mieux encore, on voit que

- si $x_0 > \ell$ alors $x_n \searrow \ell$ pour tout $n \in \mathbb{N}$,
- si $0 < x_0 < \ell$ alors $x_n \nearrow \ell$ pour tout $n \in \mathbb{N}$,
- si $-\ell < x_0 < 0$ alors $0 < x_1 < \ell$ et $x_n \nearrow \ell$ pour tout $n \in \mathbb{N}^*$,
- si $x_0 < -\ell$ alors $x_1 > \ell$ et $x_n \searrow \ell$ pour tout $n \in \mathbb{N}^*$.

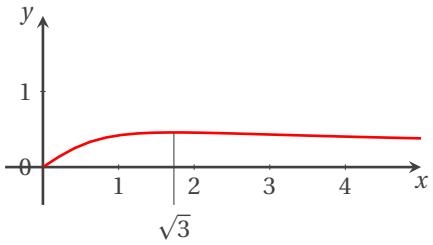
Pour prouver analytiquement la convergence pour tout $x_0 \in \mathbb{R}$ on va montrer que

$$|\varphi_2'(x)| = \frac{2}{3}\frac{|x|}{\sqrt[3]{(x^2+1)^2}} < 1 \qquad \forall x \in \mathbb{R}.$$

Comme φ_2' est une fonction impaire, il suffit de l'étudier sur $\mathbb{R}_+$.
Soit $g : \mathbb{R}_+ \to \mathbb{R}$ la fonction définie par $g(x) = \varphi_2'(x)$.

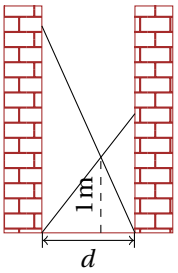
- $g(x) \geq 0$ pour tout $x \in \mathbb{R}_+$,
- g est croissante pour $0 < x < \sqrt{3}$,
- décroissante pour $x > \sqrt{3}$,
- $g(\sqrt{3}) = 1/(\sqrt[3]{2}\sqrt{3}) < 1$

donc $0 \leq g(x) < 1$ pour tout $x \in \mathbb{R}_+$ et alors $|\varphi_2'(x)| < 1$ pour tout $x \in \mathbb{R}$: la méthode converge pour tout choix du point initial x_0.



Exercice 1.15

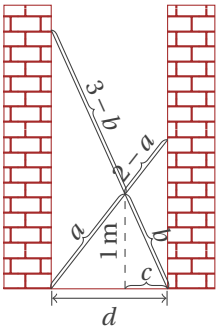
Entre deux murs (verticaux) parallèles, on place deux échelles en les croisant. La première fait 3 m de long, la seconde 2 m. On constate qu'elles se croisent à une hauteur de 1 m. Écrire la méthode de NEWTON pour le calcul approché de la distance entre les deux murs.



CORRECTION DE L'EXERCICE 1.15.

En utilisant la similarité des triangles rectangles qui ont hypoténuses respectivement 3 et 2 on a les deux équations :

$$\begin{cases} \frac{\sqrt{3^2-d^2}}{1} = \frac{3}{b} = \frac{3}{\sqrt{1^2+c^2}}, \\ \frac{\sqrt{2^2-d^2}}{1} = \frac{2}{a} = \frac{2}{\sqrt{1^2+(1-c)^2}}. \end{cases}$$



On a alors

$$\begin{cases} \sqrt{9-d^2} = \frac{3}{\sqrt{1^2+c^2}}, \\ \sqrt{4-d^2} = \frac{2}{\sqrt{1^2+(1-c)^2}}, \end{cases} \Longrightarrow \begin{cases} c = \frac{d}{\sqrt{9-d^2}}, \\ c = \frac{d}{\sqrt{4-d^2}}. \end{cases}$$

Il reste à résoudre $\dfrac{1}{\sqrt{4-d^2}} + \dfrac{1}{\sqrt{9-d^2}} = 1$.

Posons $f(d) = \frac{1}{\sqrt{4-d^2}} + \frac{1}{\sqrt{9-d^2}} - 1$. À partir de d_0 donné dans l'intervalle $]0;2[$, la méthode de NEWTON construit une suite $(d_k)_{k\in\mathbb{N}}$ par la récurrence suivante

$$d_{k+1} = d_k - \frac{f(d_k)}{f'(d_k)} = d_k - \frac{\frac{1}{\sqrt{4-d^2}} + \frac{1}{\sqrt{9-d^2}} - 1}{\frac{d}{\sqrt{(4-d^2)^3}} + \frac{d}{\sqrt{(9-d^2)^3}}}.$$

Pour que cette suite converge il faut choisir d_0 dans un intervalle $[a;b] \subset]0;2[$ tel que $\left|\left(d_k - \frac{f(d_k)}{f'(d_k)}\right)'\right| < 1$ pour tout $x \in [a;b]$.

Exercice 1.16

Soit $f, g : [a;b] \to \mathbb{R}$ deux fonctions monotones de classe $\mathscr{C}^1([a;b])$. On suppose qu'il existe un et un seul $\ell \in [a;b]$ tel que $f(\ell) = g(\ell)$. À partir de $x_0 \in [a;b]$, on construit une suite $(x_n)_{n\in\mathbb{N}}$ par la relation $f(x_{n+1}) = g(x_n)$ pour $n \in \mathbb{N}$.

1. Montrer que si $\left|\frac{g'(\ell)}{f'(\ell)}\right| < 1$ alors il existe un intervalle $[\alpha;\beta] \subset [a;b]$ tel que $x_n \to \ell$ pour tout $x_0 \in [\alpha;\beta]$.
2. Dans le cas où $\left|\frac{g'(\ell)}{f'(\ell)}\right| > 1$, proposer une méthode itérative convergente pour calculer ℓ.

Correction de l'exercice 1.16.

1. La fonction f est inversible donc la méthode donnée correspond à une méthode de point fixe $x_{n+1} = \varphi(x_n)$ où $\varphi = f^{-1} \circ g : [a;b] \to [a;b]$ est une fonction de classe $\mathscr{C}^1([a;b])$ et $\varphi'(x) = \frac{g'(x)}{f'(f^{-1}(g(x)))}$. Donc $\varphi'(\ell) = \frac{g'(\ell)}{f'(f^{-1}(g(\ell)))} = \frac{g'(\ell)}{f'(\ell)}$. Si $|g'(\ell)/f'(\ell)| < 1$ alors, pour le théorème d'Ostrowski, il existe un intervalle $[\alpha;\beta] \subset [a;b]$ tel que $x_n \to \ell$ pour tout $x_0 \in [\alpha;\beta]$.
2. Si $|g'(\ell)/f'(\ell)| > 1$, il suffit de construire la suite $(x_n)_{n\in\mathbb{N}}$ par la relation $f(x_n) = g(x_{n+1})$ pour $n \in \mathbb{N}$ et appliquer le raisonnement du point précédant.

Exercice 1.17

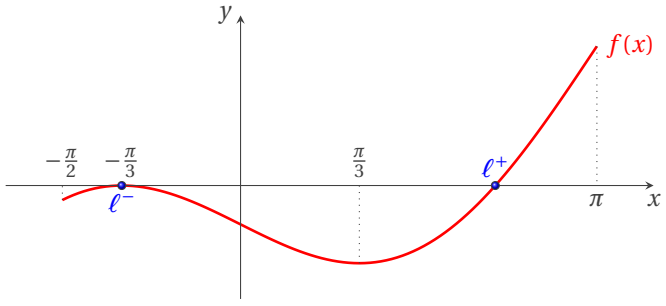
L'objectif de cet exercice est de déterminer les zéros de la fonction $f : [-\frac{\pi}{2};\pi] \to \mathbb{R}$ définie par

$$f(x) = \frac{x}{2} - \sin(x) + \frac{\pi}{6} - \frac{\sqrt{3}}{2}.$$

1. Montrer qu'il existe deux solutions $\ell^- < 0$ et $\ell^+ > 0$ de l'équation $f(x) = 0$ pour $x \in [-\frac{\pi}{2};\pi]$.
2. Peut-on appliquer la méthode de la bissection pour calculer les deux racines ? Pourquoi ? Dans le cas où c'est possible, estimer le nombre minimal d'itérations nécessaires pour calculer le(s) zéro(s) avec une tolérance $\varepsilon = 10^{-10}$ après avoir choisi un intervalle convenable.
3. Écrire la méthode de Newton pour la fonction f. À l'aide du graphe de la fonction f, déduire l'ordre de convergence de la méthode pour les deux zéros.

Correction de l'exercice 1.17.

1. Étude de la fonction f :
   - f est classe $\mathscr{C}^\infty([-\frac{\pi}{2};\pi])$;
   - $f(-\frac{\pi}{2}) = 1 - \frac{\sqrt{3}}{2} - \frac{\pi}{12} \approx -0.12785 < 0$, $f(0) = \frac{\pi}{6} - \frac{\sqrt{3}}{2} \approx -0.34244 < 0$, $f(\pi) = \frac{2\pi}{3} - \frac{\sqrt{3}}{2} \approx 1.2284 > 0$;
   - $f'(x) = \frac{1}{2} - \cos(x)$;
   - f est croissante sur $[-\frac{\pi}{2};-\frac{\pi}{3}] \cup [\frac{\pi}{3};\pi]$, décroissante sur $[-\frac{\pi}{3};\frac{\pi}{3}]$;
   - $x = -\frac{\pi}{3}$ est un maximum local et $f(-\frac{\pi}{3}) = 0$; $x = \frac{\pi}{3}$ est un minimum local et $f(\frac{\pi}{3}) < 0$;
   - $f''(x) = \sin(x)$;
   - f est concave sur $[-\frac{\pi}{2};0]$, convexe sur $[0;\pi]$.



   Par conséquence $\ell^- = -\frac{\pi}{3}$ est l'unique solution de l'équation $f(x) = 0$ pour $x \in [-\frac{\pi}{2};0]$ et il existe un et un seul ℓ^+ solution de l'équation $f(x) = 0$ pour $x \in [0;\pi]$. On peut même améliorer l'encadrement et conclure que $\ell^+ \in [\frac{\pi}{3};\pi]$.
2. La méthode de dichotomie ne peut pas être utilisée pour approcher ℓ^- car il est impossible de trouver un intervalle $]a,b[\subset \mathbb{R}^-$ sur lequel $f(a)f(b) < 0$. En ce qui concerne l'approximation de ℓ^+, en partant de $[a,b] = [\frac{\pi}{3};\pi]$, la méthode de dichotomie converge en $\log_2\left(\frac{b-a}{\varepsilon}\right) \approx 35$ itérations vers la valeur 2.246005589.

3. La méthode de NEWTON est une méthode de point fixe

$$\begin{cases} x_{k+1} = \phi(x_k), \\ x_0 \text{ donné}, \end{cases}$$

avec ϕ l'application définie par $\phi(x) = x - \frac{f(x)}{f'(x)}$. Ici donc elle s'écrit

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} = x_k - \frac{x_k - 2\sin(x_k) + \frac{\pi}{3} - \sqrt{3}}{1 - 2\cos(x_k)}.$$

À l'aide du graphe de la fonction f, on voit que la méthode converge vers ℓ^+ quel que soit $x_0 \in [-\pi/2; \pi/3]$ avec un ordre de convergence quadratique et converge vers ℓ^- quel que soit $x_0 \in [\pi/3; \pi]$ avec un ordre de convergence linéaire (car $f(\ell^-) = f'(\ell^-) = 0$).

Exercice 1.18 *(Python)*

Soit la fonction $f_\gamma(x) = \cosh(x) + \cos(x) - \gamma$. Pour $\gamma = 1, 2, 3$ trouver (graphiquement) un intervalle qui contient le zéro de f_γ. Calculer ce dernier par la méthode de dichotomie avec une tolérance de 10^{-10}. Utiliser ensuite la méthode de NEWTON. Pourquoi cette méthode n'est-elle pas précise quand $\gamma = 2$?

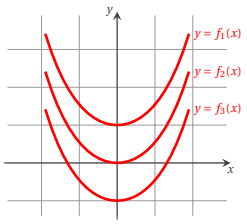
CORRECTION DE L'EXERCICE 1.18.

Étude de f_γ. On se rappelle que $\cosh(x) = \frac{e^x + e^{-x}}{2}$ et $\sinh(x) = \frac{e^x - e^{-x}}{2}$ donc

- $\lim_{x \to \pm\infty} f_\gamma(x) = +\infty$
- $f'_\gamma(x) = \sinh(x) - \sin(x)$ et $f'_\gamma(x) = 0$ si et seulement si $x = 0$ (comparer les graphes de sinh et sin et se rappeler que pour $x > 0$ on a $\sinh(x) > x > \sin(x)$ et pour $x < 0$ on a $\sinh(x) < x < \sin(x)$)
- $f''_\gamma(x) = \cosh(x) - \cos(x) > 0$ pour tout $x \neq 0$.

Par conséquence,

- pour $\gamma = 1$, la fonction n'a pas de zéro réel,
- pour $\gamma = 2$ il n'y a que le zéro $\widehat{x} = 0$ et il est de multiplicité quatre (c'est-à-dire $f_2(\widehat{x}) = f'_2 a(\widehat{x}) = f''_2(\widehat{x}) = f'''_2(\widehat{x}) = 0$ et $f_2^{(IV)}(\widehat{x}) \neq 0$),
- pour $\gamma = 3$, f_3 admet deux zéros distincts, un dans l'intervalle $]-3, -1[$ et l'autre dans $]1, 3[$.



Méthode de la dichotomie. Dans le cas $\gamma = 2$, la méthode de dichotomie ne peut pas être utilisée car il est impossible de trouver un intervalle $]a, b[$ sur lequel $f_2(a) f_2(b) < 0$. Pour $\gamma = 3$, en partant de $[a, b] = [-3, -1]$, la méthode de dichotomie converge en 34 itérations vers la valeur $\widehat{x} = -1.85792082914850$ avec $f_3(\widehat{x}) \simeq -3.6 \times 10^{-12}$. De même, en prenant $[a, b] = [1, 3]$, la méthode de dichotomie converge en 34 itérations vers la valeur $\widehat{x} = 1.85792082914850$ avec $f_3(\widehat{x}) \simeq -3.6877 \times 10^{-12}$.

Méthode de Newton. Considérons le cas où $\gamma = 2$. En partant de la donnée initiale $x_0 = 1$, la méthode de NEWTON converge vers la valeur $\widehat{x} = 1.4961 \times 10^{-4}$ en 31 itérations avec $\varepsilon = 10^{-10}$ tandis que la racine exacte de f_2 est 0. Cet écart est dû au fait que f_2 est quasiment constante au voisinage de sa racine, donc le problème de recherche du zéro est mal conditionné. La méthode converge vers la même solution et avec le même nombre d'itérations même si on prend ε égal au zéro machine. Considérons le cas $\gamma = 3$. La méthode de NEWTON avec ε égal au zéro machine converge vers 1.85792082915020 après 9 itérations en partant de $x_0 = 1$, alors que si $x_0 = -1$, elle converge après 9 itérations vers -1.85792082915020.

Voici les instructions :

```
130 def f(x):
131 ⟶return math.cosh(x)+math.cos(x)-gamma
132
133 maxITER = 100
134
135 gamma = 3
```

```
tol = 1.0e-15
a = -3.
b = -1.
x_init = -1.
print "Zero calcule par la methode de dichotomie dans l'intervalle [", a, ",", b,"] : ", dichotomie(f,a,b,
    ➥tol,maxITER)
print "Zero calcule par la methode de \textsc{Newton} a partir du point x_0 =",x_init," : ", newton(f,
    ➥x_init,tol,maxITER)
a = 1.
b = 3.
x_init = 1.
print "Zero calcule par la methode de dichotomie dans l'intervalle [", a, ",", b,"] : ", dichotomie(f,a,b,
    ➥tol,maxITER)
print "Zero calcule par la methode de \textsc{Newton} a partir du point x_0 =",x_init," : ", newton(f,
    ➥x_init,tol,maxITER)

gamma = 2
tol = 1.0e-10
x_init = -1.
print "Zero calcule par la methode de \textsc{Newton} a partir du point x_0 =",x_init," : ", newton(f,
    ➥x_init,tol,maxITER)
x_init = 1.
print "Zero calcule par la methode de \textsc{Newton} a partir du point x_0 =",x_init," : ", newton(f,
    ➥x_init,tol,maxITER)
```

Exercice 1.19 *Équation d'état d'un gaz (Python)*

Nous voulons déterminer le volume V occupé par un gaz dont la température est T et dont la pression est p. L'équation d'état (*i.e.* l'équation liant p, V et T) selon le modèle de VAN DER WAALS est donnée par

$$\left(p + a\left(\frac{N}{V}\right)^2\right)(V - Nb) = kNT,$$

où a et b sont deux coefficients qui dépendent du gaz considéré, N est le nombre de molécules contenues dans le volume V et $k = 1.3806503 \times 10^{-23}\,\mathrm{J\,K^{-1}}$ est la constante de Boltzmann. Nous devons donc résoudre une équation non linéaire dont la racine est V.
Pour le dioxyde de carbone CO_2, les coefficients a et b prennent les valeurs $a = 0.401\,\mathrm{Pa\,m^3}$ et $b = 42.7 \times 10^{-6}\,\mathrm{m^3}$. Trouver le volume occupé par 1000 molécules de CO_2 à la température $T = 300\,\mathrm{K}$ et la pression $p = 3.5 \times 10^7$ Pa par la méthode de dichotomie, avec une tolérance de 10^{-12}.

CORRECTION DE L'EXERCICE 1.19. On doit calculer les zéros de la fonction $f(V) = pV + aN^2/V - abN^3/V^2 - pNb - kNT$, où N est le nombre de molécules. On a

- $\lim_{V\to 0^+} f(V) = -\infty$ et $\lim_{V\to+\infty} f(V) = +\infty$
- $f'(V) = p - aN^2/V^2 + 2abN^3/V^3 = p + aN^2(2bN/V - 1)/V^2$
- $f'(V) = 0$ si et seulement si $\frac{p}{aN^2}V^3 - V = -2bN$ donc pour aucun $V > 0$.

En traçant le graphe de f, on voit que cette fonction n'a qu'un zéro simple dans l'intervalle $]0.01, 0.06[$ avec $f(0.01) < 0$ et $f(0.06) > 0$. On peut calculer ce zéro en utilisant la méthode de dichotomie comme suit :

```
def f(V):
    a = 0.401
    b = 42.7e-6
    N = 1000.
    T = 300.
    p = 3.5e7
    k = 1.3806503e-23
    return p*V+a*N**2/V-a*b*N**3/V**2-p*N*b-k*N*T

tol = 1.0e-12
left = 0.01
right = 0.06

print "Zero calcule par la methode de dichotomie dans l'intervalle [", left, ",", right,"] : ", dichotomie
    ➥(f,left,right,tol,maxITER)
```

ce qui donne $\widehat{V} = 0.042699999999\,\mathrm{m}^3$.

Exercice 1.20

Soit A est un nombre positif donné et considérons l'algorithme suivant : étant donné une valeur x_0, on calcule

$$x_{k+1} = x_k + \frac{A - x_k^2}{2}, \qquad k = 0, 1, 2, \ldots$$

1. Montrer que si la suite x_k converge, alors sa limite est soit $\sqrt{A}$ soit $-\sqrt{A}$.
2. On considère le cas où $A \in]0, 4[$. Montrer qu'il existe $\varepsilon > 0$ tel que, si $|x_0 - \sqrt{A}| \leq \varepsilon$ alors la suite x_k converge vers $\sqrt{A}$.
3. Vérifier graphiquement que si x_0 est proche de $-\sqrt{A}$ mais différent de $-\sqrt{A}$, alors la suite x_k ne converge pas vers $-\sqrt{A}$.
4. Vérifier que si $x_0 = 1$, alors l'algorithme coïncide avec la méthode de la corde 2 pour résoudre $x^2 - A = 0$.
5. Proposer un algorithme plus efficace pour calculer la racine carrée d'un nombre positif A.

Correction de l'exercice 1.20.

1. Supposons que x_k converge vers ℓ. En passant à la limite dans la formule de récurrence on obtient

$$\ell = \ell + \frac{A - \ell^2}{2},$$

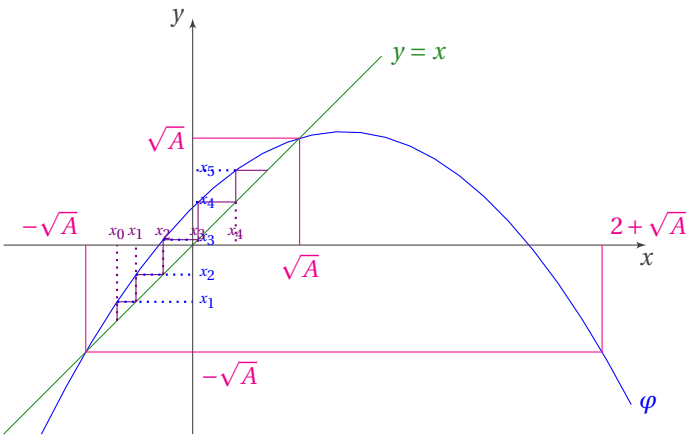
c'est-à-dire $\ell^2 = A$ et donc $\ell = \pm\sqrt{A}$.

2. La méthode peut s'écrire sous la forme d'une méthode de point fixe où la fonction φ est définie par

$$\varphi(x) = x + \frac{A - x^2}{2}.$$

Si $A \in]0, 4[$ et $\ell = \sqrt{A}$, puisque $\varphi'(x) = 1 - x$, alors $|\varphi'(\ell)| = |1 - \sqrt{A}| < 1$: on peut appliquer le théorème d'Ostrowski donc il existe $\varepsilon > 0$ tel que, si $|x_0 - \sqrt{A}| \leq \varepsilon$ alors la suite x_k converge vers $\sqrt{A}$.

3. On a représenté dans la figure ci-dessous le graphe de la fonction φ lorsque $A = 1/2$. Si on choisit $x_0 < -\sqrt{A}$ alors la suite diverge vers $-\infty$; si $-\sqrt{A} < x_0 < \sqrt{A}$ alors la suite converge (de manière monotone croissante) vers $\sqrt{A}$; si $\sqrt{A} < x_0 < 2 + \sqrt{A}$ alors la suite converge (de manière monotone croissante après la première itération) vers $\sqrt{A}$; si $x_0 > 2 + \sqrt{A}$ alors la suite diverge vers $-\infty$.



4. Soit f la fonction définie par $f(x) = x^2 - A$. La méthode de la corde 2 pour résoudre $f(x) = 0$ s'écrit dans ce cas

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_0)} = x_k - \frac{x_k^2 - A}{2x_0}, \qquad k = 0, 1, 2, \ldots.$$

Si on choisit $x_0 = 1$, cette méthode s'écrit donc

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_0)} = x_k - \frac{x_k^2 - A}{2}, \qquad k = 0, 1, 2, \ldots.$$

Ainsi on conclut que la méthode donnée coïncide avec la méthode de la corde 2 pour résoudre $x^2 - A = 0$ lorsque $x_0 = 1$ comme point de départ.

5. Si on choisit la méthode de NEWTON pour résoudre $f(x)=0$ avec $f(x)=x^2-A$, on a

$$x_{k+1}=x_k-\frac{f(x_k)}{f'(x_k)}=x_k-\frac{x_k^2-A}{2x_k},\qquad k=0,1,2,\ldots.$$

Cette méthode est plus efficace que la précédente car elle converge à l'ordre 2 pour tout $x_0>0$.

Exercice 1.21

Soit $f:\mathbb{R}\to\mathbb{R}$ la fonction définie par $f(x)=x^3-2$. On veut approcher le zéro α de f par la méthode de point fixe suivante :

$$\begin{cases} x_0 \text{ donné},\\ x_{k+1}=g_\omega(x_k) \text{ pour tout } k\geq 0, \end{cases} \tag{1.2}$$

avec $g_\omega:\mathbb{R}\to\mathbb{R}$ la fonction définie par

$$g_\omega(x)=(1-\omega)x^3+\left(1-\frac{\omega}{3}\right)x+2(\omega-1)+\frac{2\omega}{3x^2},\qquad \omega\in\mathbb{R}.$$

1. Pour quelles valeurs du paramètre ω la méthode de point fixe (1.2) est-elle consistante (*i.e.* α est un point fixe de g_ω) ?
2. Pour quelles valeurs du paramètre ω la méthode de point fixe (1.2) est d'ordre 2 ?
3. Existe-t-il des valeurs du paramètre ω pour lesquelles la méthode de point fixe (1.2) est-elle d'ordre 3 ?

CORRECTION DE L'EXERCICE 1.21. Comme α est le zéro de f, on a $\alpha^3=2$.

1. La méthode de point fixe (1.2) est consistante pour tout $\omega\in\mathbb{R}$ car

$$g_\omega(\alpha)=(1-\omega)\alpha^3+\left(1-\frac{\omega}{3}\right)\alpha+2(\omega-1)+\frac{2\omega}{3\alpha^2}=(1-\omega)(\alpha^3-2)+\left(1-\frac{\omega}{3}\right)\alpha+\frac{2\omega}{3\alpha^2}=\alpha-\frac{\omega\alpha}{3}+\frac{2\omega}{3\alpha^2}=\alpha-\frac{\omega(\alpha^3-2)}{3\alpha^2}=\alpha.$$

2. La méthode de point fixe (1.2) est au moins d'ordre 2 si $g'(\alpha)=0$. On a

$$g'_\omega(\alpha)=3(1-\omega)\alpha^2+1-\frac{\omega}{3}-\frac{4\omega}{3\alpha^3}=3(1-\omega)\alpha^2+1-\omega=(1-\omega)(3\alpha^2+1)$$

donc la méthode de point fixe (1.2) est au moins d'ordre 2 si $\omega=1$.

3. Pour que la méthode de point fixe (1.2) soit d'ordre 3 il faudrait $g'(\alpha)=g''(\alpha)=0$. Puisque $g'(\alpha)=0$ si et seulement si $\omega=1$ et $g_1''(\alpha)=\frac{4\omega}{\alpha^4}\neq 0$, il n'est pas possible d'avoir une convergence d'ordre supérieur à 2.

Exercice 1.22

On considère le problème du calcul de $\ell\in[0,\pi]$ tel que $\ell=1-\frac{1}{4}\cos(\ell)$.

1. Montrer qu'on peut utiliser la méthode de la dichotomie pour approcher ℓ. Que vaut l'approximation de ℓ après 3 itérations ? Quel est l'erreur maximale qu'on obtient après 3 itérations ?

k	0	1	2	3
$[a_k,b_k]$	$[0,\pi]$			
ℓ_k	$\frac{\pi}{2}$			

2. On considère la méthode de point fixe suivante :

$$\begin{cases} x_0\in[0,\pi],\\ x_{k+1}=g(x_k) \text{ pour tout } k\geq 0, \end{cases} \tag{1.3}$$

avec $g:[0,\pi]\to\mathbb{R}$ la fonction définie par $g(x)=1-\frac{1}{4}\cos(x)$.

2.1. Étudier graphiquement la convergence de cette méthode.

2.2. Montrer rigoureusement que la méthode converge pour tout $x_0\in[0,\pi]$.

2.3. Montrer que l'erreur satisfait l'inégalité $|x_k-\ell|\leq C^k|x_0-\ell|$. Donner une estimation de la constante C et l'utiliser pour minorer le nombre d'itérations nécessaires pour approcher ℓ à 10^{-3} près.

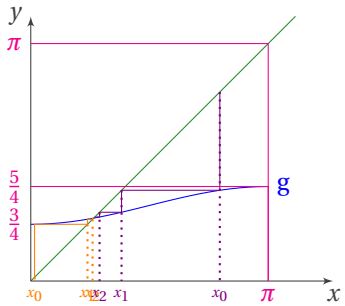
2.4. Montrer que si on utilise le critère d'arrêt $|x_{k+1}-x_k|\leq\varepsilon$ alors $|x_{k+1}-\ell|\leq\frac{\varepsilon}{1-C}$. Quelle valeur de ε faut-il choisir pour approcher ℓ à 10^{-3} près ?

CORRECTION DE L'EXERCICE 1.22.

1. Soit $f\colon [0,\pi]\to\mathbb{R}$ la fonction définie par $f(x)=1-\frac{1}{4}\cos(x)-x$. Elle est de classe $\mathscr{C}^\infty$, $f(0)=3/4>0$ et $f(\pi)=5/4-\pi<0$, le théorème des valeurs intermédiaires permet de conclure qu'il existe au moins un $\ell\in[0,\pi]$ tel que $f(\ell)=0$. De plus, comme $f'(x)=\frac{1}{4}\cos(x)-1<0$, ce zéro est unique. On peut alors utiliser la méthode de la dichotomie pour approcher ℓ et l'on a

k	0	1	2	3
$[a_k,b_k]$	$[0,\pi]$	$\left[0,\frac{\pi}{2}\right]$	$\left[\frac{\pi}{4},\frac{\pi}{2}\right]$	$\left[\frac{\pi}{4},\frac{3\pi}{8}\right]$
ℓ_k	$\frac{\pi}{2}$	$\frac{\pi}{4}$	$\frac{3\pi}{8}$	$\frac{5\pi}{16}$

   L'erreur qu'on obtient après 3 itérations est au plus égale à la largeur de l'intervalle $[a_3;b_3]$, c'est-à-dire inférieure à $\frac{b-a}{2^3}=\frac{\pi}{8}$.
2. On considère la méthode de point fixe de fonction d'itération g.
   2.1. Étude graphique de la convergence :
   - ⋆ g est de classe $\mathscr{C}^\infty$, $g(0)=3/4$, $g(\pi)=5/4$, $g'(x)=\frac{1}{4}\sin(x)\in[0,1/4]$, g est croissante sur $[0,\pi]$.
   - ⋆ La suite x_n est monotone croissante si $x_0<\ell$ et monotone décroissante si $x_0>\ell$.



   2.2. $g([0,\pi])=[3/4,5/4]\subset[0,\pi]$ et $|g'(x)|\le 1/4<1$: la méthode de point fixe converge vers ℓ pour tout $x_0\in[0,\pi]$.
   2.3. Pour tout $k\in\mathbb{N}$ il existe ξ_k compris entre ℓ et x_k tel que $|x_k-\ell|=|g(x_{k-1})-g(\ell)|=|g'(\xi_k)||x_{k-1}-\ell|\le\frac{1}{4^k}|x_0-\ell|\le\frac{\pi}{4^k}$. Donc, pour approcher ℓ à 10^{-3} près, il faut prendre le plus petit $k\in\mathbb{N}$ qui vérifie $k\ge\log_4(10^3\pi)\approx 5.9$, *i.e.* $k=6$.
   2.4. Pour tout $k\in\mathbb{N}$ on a $|x_k-\ell|-|x_{k+1}-x_k|\le|x_{k+1}-x_k+x_k-\ell|=|x_{k+1}-\ell|\le C|x_k-\ell|$ avec $C=1/4$ d'où

$$|x_{k+1}-\ell|\le\frac{1}{1-C}|x_{k+1}-x_k|\le\frac{\varepsilon}{1-C}.$$

   Pour que l'erreur soit inférieur à 10^{-3} il faut alors choisir $\varepsilon\le(1-C)10^{-3}$.

Exercice 1.23

On considère le problème du calcul de $\ell\in[0,\pi]$ tel que $\ell=1+\frac{1}{2}\sin(\ell)$.

1. Montrer qu'on peut utiliser la méthode de la dichotomie pour approcher ℓ. Que vaut l'approximation de ℓ après 3 itérations ?
2. On considère la méthode de point fixe suivante :

$$\begin{cases} x_0\in[0,\pi], \\ x_{k+1}=g(x_k) \text{ pour tout } k\ge 0, \end{cases} \tag{1.4}$$

   avec $g\colon[0,\pi]\to\mathbb{R}$ la fonction définie par $g(x)=1+\frac{1}{2}\sin(x)$.
   2.1. Étudier graphiquement la convergence de cette méthode.
   2.2. Montrer rigoureusement que la méthode converge pour tout $x_0\in[0,\pi]$.
   2.3. Montrer que l'erreur satisfait l'inégalité $|x_k-\ell|\le C^k|x_0-\ell|$. Donner une estimation de la constante C et l'utiliser pour minorer le nombre d'itérations nécessaires pour approcher ℓ à 10^{-3} près.
   2.4. Montrer que si on utilise le critère d'arrêt $|x_{k+1}-x_k|\le\varepsilon$ alors $|x_{k+1}-\ell|\le\frac{\varepsilon}{1-C}$. Quelle valeur de ε faut-il choisir pour approcher ℓ à 10^{-3} près ? *(Rappel : $|a-c|-|c-b|\le|a-b|\le|a-c|+|c-b|$ pour tout $a,b,c\in\mathbb{R}$)*

CORRECTION DE L'EXERCICE 1.23.

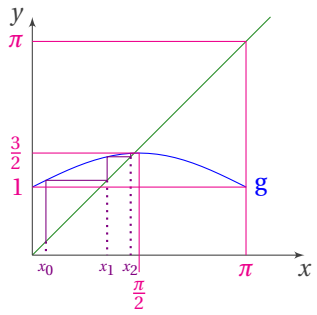
1. Soit $f\colon[0,\pi]\to\mathbb{R}$ la fonction définie par $f(x)=1+\frac{1}{2}\sin(x)-x$. Elle est de classe $\mathscr{C}^\infty$, $f(0)=1>0$ et $f(\pi)=1-\pi<0$, le théorème des valeurs intermédiaires permet de conclure qu'il existe au moins un $\ell\in[0,\pi]$ tel que $f(\ell)=0$. De plus, comme $f'(x)=\frac{1}{2}\cos(x)-1<0$, ce zéro est unique. On peut alors utiliser la méthode de la dichotomie pour approcher ℓ et l'on a

k	0	1	2	3
$[a_k, b_k]$	$[0,\pi]$	$\left[0,\frac{\pi}{2}\right]$	$\left[\frac{\pi}{4},\frac{\pi}{2}\right]$	$\left[\frac{3\pi}{8},\frac{\pi}{2}\right]$
ℓ_k	$\frac{\pi}{2}$	$\frac{\pi}{4}$	$\frac{3\pi}{8}$	$\frac{7\pi}{16}$

2. On considère la méthode de point fixe de fonction d'itération g.

2.1. Étude graphique de la convergence :

g est de classe $\mathscr{C}^\infty$, $g(0) = g(\pi) = 1$, $g'(x) = \frac{1}{2}\cos(x) \in [-1/2, 1/2]$, g est croissante sur $[0, \frac{\pi}{2}]$, décroissante sur $[\frac{\pi}{2}, \pi]$ et $g(\pi/2) = 3/2 < \pi$



2.2. $g([0,\pi]) = [1, 3/2] \subset [0,\pi]$ et $|g'(x)| \le 1/2 < 1$: la méthode de point fixe converge pour tout $x_0 \in [0,\pi]$.

2.3. Pour tout $k \in \mathbb{N}$ il existe ξ_k compris entre ℓ et x_k tel que $|x_k - \ell| = |g(x_{k-1}) - g(\ell)| = |g'(\xi_k)||x_{k-1} - \ell| \le \frac{1}{2^k}|x_0 - \ell| \le \frac{\pi}{2^k}$. Donc, pour approcher ℓ à 10^{-3} près, il faut prendre le plus petit $k \in \mathbb{N}$ qui vérifie $k \ge \log_2(10^3\pi) \approx 11.7$, *i.e.* $k = 12$.

2.4. Pour tout $k \in \mathbb{N}$ on a $[|x_k - \ell| - |x_{k+1} - x_k| \le |x_{k+1} - x_k + x_k - \ell| = |x_{k+1} - \ell| \le C|x_k - \ell|$ d'où

$$|x_{k+1} - \ell| \le \frac{1}{1-C}|x_{k+1} - x_k| \le \frac{\varepsilon}{1-C}.$$

Pour que l'erreur soit inférieur à 10^{-3} il faut alors choisir $\varepsilon \le (1-C) \times 10^{-3}$.

Exercice 1.24

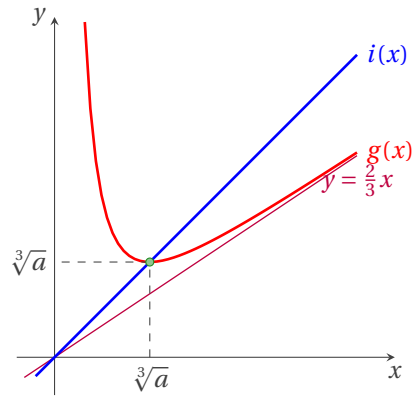
Le but de cet exercice est de calculer la racine cubique d'un nombre positif a. Soit g la fonction définie sur $\mathbb{R}_+^*$ par

$$g(x) = \frac{2}{3}x + \frac{1}{3}\frac{a}{x^2} \qquad (a > 0 \text{ fixé}).$$

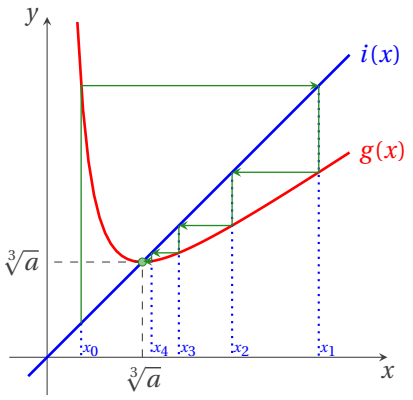
1. Faire l'étude complète de la fonction g.
2. Comparer g à l'identité.
3. Soit la suite $(x_n)_{n\in\mathbb{N}}$ définie par
$$x_{n+1} = g(x_n), \qquad x_0 > 0.$$
À l'aide des graphe de g et de l'identité sur $\mathbb{R}_+^*$, dessiner la suite $(x_n)_{n\in\mathbb{N}}$ sur l'axe des abscisses. Observer graphiquement la convergence.
4. Justifier mathématiquement la convergence observée graphiquement. En particulier, montrer que cette suite est décroissante à partir du rang 1.
5. Calculer l'ordre de convergence de la suite.
6. Écrire l'algorithme défini par la suite $(x_n)_{n\in\mathbb{N}}$ qui permet de déterminer $\sqrt[3]{a}$ à une précision de 10^{-6}.
7. Expliciter la méthode de Newton pour la recherche du zéro de la fonction f définie par $f(x) = x^3 - a$. Que remarque-t-on ?

Correction de l'exercice 1.24.

1. Étude de la fonction $g : \mathbb{R}_+^* \to \mathbb{R}$ définie par $g(x) = \frac{2}{3}x + \frac{1}{3}\frac{a}{x^2}$:
   - ⋆ $g(x) > 0$ pour tout $x \in \mathbb{R}_+^*$;
   - ⋆ $\lim\limits_{x\to 0^+} g(x) = \lim\limits_{x\to+\infty} g(x) = +\infty$;
     $\lim\limits_{x\to+\infty} \frac{g(x)}{x} = \frac{2}{3}$ et $\lim\limits_{x\to+\infty} g(x) - \frac{2}{3}x = 0$ donc $y = \frac{2}{3}x$ est un asymptote et l'on a $g(x) > \frac{2}{3}x$ pour tout $x > 0$;
   - ⋆ $g'(x) = \frac{2}{3}\left(1 - \frac{a}{x^3}\right)$;
   - ⋆ g est croissante sur $[\sqrt[3]{a}, +\infty[$, décroissante sur $[0, \sqrt[3]{a}]$;
   - ⋆ $x = \sqrt[3]{a}$ est un minimum absolu et $g(\sqrt[3]{a}) = \sqrt[3]{a}$,
   - ⋆ $g''(x) = \frac{2a}{x^4} > 0$: g est convexe sur $\mathbb{R}_+^*$.



(a) Graphe de g comparé au graphe de $i(x) = x$.



(b) Étude graphique de la convergence de la méthode de point fixe.

FIGURE 1.4.: Exercice 1.24

x	0	$\sqrt[3]{a}$	$+\infty$
$g'(x)$	$-$		$+$
$g(x)$	$+\infty$	$\sqrt[3]{a}$	$+\infty$

2. Graphe de g comparé au graphe de $i(x) = x$: voir la figure 1.4a. On vérifie analytiquement qu'il existe une et une seule intersection entre la courbe d'équation $y = g(x)$ et la droite d'équation $y = x$:

$$g(x) = x \iff \frac{2}{3}x + \frac{1}{3}\frac{a}{x^2} = x \iff x^3 = a.$$

3. Étude graphique de la convergence de la méthode de point fixe : voir la figure 1.4a.
4. On en déduit que pour tout $x > 0$ on a $g(x) \geq \sqrt[3]{a}$. Donc, pour tout $k > 0$, $x_k = g(x_{k-1}) \geq \sqrt[3]{a}$. Vérifions les hypothèses du théorème de point fixe qui fournit une condition suffisante de convergence de la suite :
   4.1. pour tout x dans $[\sqrt[3]{a}, +\infty[$ on a $g(x) > \sqrt[3]{a}$ donc $g([\sqrt[3]{a}, +\infty[) \subset [\sqrt[3]{a}, +\infty[$ (*i.e.* l'intervalle $\sqrt[3]{a}, +\infty[$ est stable) ;
   4.2. $g \in \mathscr{C}^1([\sqrt[3]{a}, +\infty[)$;
   4.3. pour tout x dans $[\sqrt[3]{a}, +\infty[$ on a
   $$|g'(x)| = \left|\frac{2}{3}\left(1 - \frac{a}{x^3}\right)\right| < 1$$
   donc g est contractante.

   Alors la méthode converge vers $\widehat{x}$ point fixe de g. De plus, pour tout $x_0 \in [\sqrt[3]{a}, +\infty[$ on a $\widehat{x} = g(\widehat{x}) \iff \widehat{x} = \sqrt[3]{a}$: la méthode permet donc de calculer de façon itérative la racine cubique de a.
5. Étant donné que
   $$g'(\widehat{x}) = 0, \qquad g''(\widehat{x}) = \frac{2a}{\widehat{x}^4} \neq 0$$
   la méthode de point fixe converge à l'ordre 2.
6. Algorithme de point fixe :

   Require: $x_0 > 0$
   while $|x_{k+1} - x_k| > 10^{-6}$ **do**
   $x_{k+1} \leftarrow g(x_k)$
   end while

   Quelques remarques à propos du critère d'arrêt basé sur le contrôle de l'incrément. Les itérations s'achèvent dès que $|x_{k+1} - x_k| < \varepsilon$; on se demande si cela garantît-t-il que l'erreur absolue e_{k+1} est elle aussi inférieur à ε. L'erreur absolue à l'itération $(k+1)$ peut être évaluée par un développement de Taylor au premier ordre
   $$e_{k+1} = |g(\widehat{x}) - g(x_k)| = |g'(z_k)e_k|$$

avec z_k compris entre $\widehat{x}$ et x_k. Donc

$$|x_{k+1}-x_k| = |e_{k+1}-e_k| = |g'(z_k)-1|e_k \simeq |g'(\widehat{x})-1|e_k.$$

Puisque $g'(\widehat{x})=0$, on a bien $|x_{k+1}-x_k| \simeq e_k$.

7. La méthode de NEWTON est une méthode de point fixe avec $g(x) = x - \frac{f(x)}{f'(x)}$. Ici elle s'écrit

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} = x_k - \frac{x_k^3-a}{3x_k^2} = x_k - \frac{1}{3}x_k + \frac{a}{3x_k^2} = \frac{2}{3}x_k + \frac{a}{3x_k^2}$$

autrement dit la méthode de point fixe assignée est la méthode de NEWTON (qu'on sait être d'ordre de convergence égale à 2 lorsque la racine est simple).

Exercice 1.25

On veut résoudre l'équation $e^{-\alpha x} = x$ avec $0 < \alpha < 1$.

1. Vérifier que cette équation admet une unique solution, notée ℓ_α, dans $\mathbb{R}$.
2. Soit $g\colon \mathbb{R} \to \mathbb{R}$ la fonction définie par $g(x) = e^{-\alpha x}$. On définit la suite récurrente

$$\begin{cases} u_0 \in \mathbb{R} \\ u_{n+1} = g(u_n). \end{cases} \tag{1.5}$$

On veut montrer que u_n converge vers ℓ_α. Pour cela, comparer d'abord le graphe de g à l'identité et observer graphiquement la convergence, ensuite justifier mathématiquement la convergence observée graphiquement.

3. Écrire la méthode de NEWTON pour résoudre l'équation $e^{-\alpha x} = x$ avec $0 < \alpha < 1$. Parmi la méthode de NEWTON et la méthode de point fixe (1.5), laquelle faut-il préférer vis-à-vis de la vitesse de convergence ?

CORRECTION DE L'EXERCICE 1.25.

1. Deux méthodes (équivalentes) possibles :

   Méthode 1 : La fonction $g\colon x \mapsto e^{-\alpha x}$ est continue monotone décroissante, $\lim_{x\to-\infty} e^{-\alpha x} = +\infty$ et $\lim_{x\to+\infty} e^{-\alpha x} = 0$; par conséquente elle intersecte la droite d'équation $y = x$ une et une seule fois. Notons ce point ℓ_α. Comme la fonction $x \mapsto e^{-\alpha x}$ est positive pour tout $x \in \mathbb{R}$ tandis que la fonction $x \mapsto x$ est positive si et seulement si $x > 0$, on en déduit que $\ell_\alpha > 0$. De plus, comme $g(1) = e^{-\alpha} < 1$, on peut conclure que $\ell_\alpha \in]0;1[$.

   Méthode 2 : La fonction $f\colon x \mapsto e^{-\alpha x} - x$ est continue monotone décroissante, $\lim_{x\to-\infty} e^{-\alpha x} - x = +\infty$ et $\lim_{x\to+\infty} e^{-\alpha x} - x = -\infty$; par le théorème des valeurs intermédiaires on conclut qu'il existe un et un seul $\ell_\alpha \in \mathbb{R}$ tel que $f(\ell_\alpha) = 0$. Comme $f(0) > 0$, on peut appliquer à nouveau le théorème des valeurs intermédiaires à l'intervalle $[0;\infty[$ et en déduire que $\ell_\alpha > 0$. De plus, comme $f(1) < e^{-1} - 1 < 0$, on peut conclure que $\ell_\alpha \in]0;1[$.

2. Le graphe de la fonction g est celui en figure 1.25. On en déduit que
   - ⋆ la suite $(u_n)_n$ converge pour tout $u_0 \in \mathbb{R}$;
   - ⋆ $g(\mathbb{R}) =]0;+\infty[$ et $g(]0;+\infty[) =]0;1[$ ainsi $u_1 \in]0;+\infty[$ et $u_n \in]0;1[$ pour tout $n > 1$;
   - ⋆ la convergence n'est pas monotone : la sous-suite des termes d'indice pair est monotone croissante tandis que la sous-suite des termes d'indice impair est monotone décroissante (ce qui veut dire d'une part qu'on ne pourra pas utiliser les théorèmes du type «monotone+bornée=convergente» pour prouver la convergence, d'autre part on voit aussi que ni l'intervalle $[\ell_\alpha;+\infty[$ ni l'intervalle $[0;\ell_\alpha]$ sont stables) ;
   - ⋆ $|g'(x)|$ n'est pas bornée pour tout $x \in \mathbb{R}$ (croissance exponentielle à $-\infty$). Plus particulièrement, $|g'(x)| < 1$ ssi $e^{\alpha x} > \alpha$ ssi $x > \ln(\alpha)/\alpha$. Comme $0 < \alpha < 1$, on conclut que $|g'(x)| < 1$ pour tout $x \geq 0$.

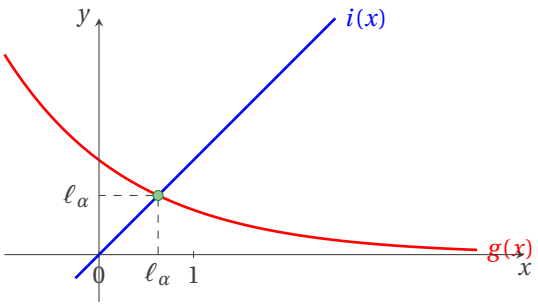
   Cette étude préliminaire suggère d'utiliser le théorème de point fixe dans l'intervalle $]0;+\infty[$. On a
   - ⋆ $g \in \mathscr{C}^\infty(]0;+\infty[)$,
   - ⋆ $g(]0;+\infty[) \subset]0;+\infty[$,
   - ⋆ $|g'(x)| < 1$ pour tout $x \in]0;+\infty[$,

   on peut alors utiliser le théorème de point fixe pour conclure que la suite $(u_n)_{n\in\mathbb{N}}$ converge vers ℓ_α pour tout $u_0 \in]0;+\infty[$. Comme $g(x) \in]0;+\infty[$ pour tout $x \in \mathbb{R}$, alors $u_n \in]0;+\infty[$ pour tout $n \in \mathbb{N}^*$, on peut donc conclure que la suite $(u_n)_{n\in\mathbb{N}}$ converge vers ℓ_α pour tout $u_0 \in \mathbb{R}$.

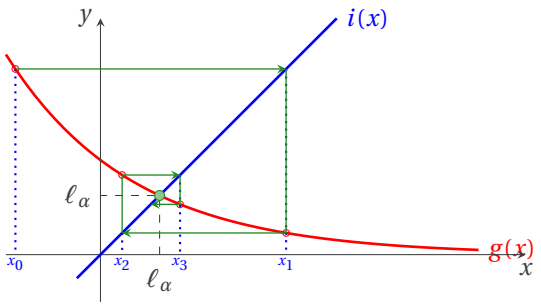
3. Soit $f(x) = e^{-\alpha x} - x$. La méthode de NEWTON (qui s'applique à f et non à g) définit la suite récurrente

$$\begin{cases} u_0 \in \mathbb{R} \\ u_{n+1} = u_n - \frac{e^{-\alpha u_n} - u_n}{-\alpha e^{-\alpha u_n} - 1}. \end{cases} \tag{1.6}$$

La méthode de point fixe (1.5) n'est que d'ordre 1 car $g'(\ell_\alpha) \neq 0$ tandis que la méthode de NEWTON, qui est encore une méthode de point fixe, est d'ordre 2 (car α est un zéro simple).



(a) Graphe de g comparé au graphe de $i(x) = x$.



(b) Étude graphique de la convergence de la méthode de point fixe.

FIGURE 1.5.: Exercice 1.25

Exercice 1.26

Soit f une application de $\mathbb{R}$ dans $\mathbb{R}$ définie par $f(x) = \exp(x^2) - 4x^2$. On se propose de trouver les racines réelles de f.

1. Situer les 4 racines de f (*i.e.* indiquer 4 intervalles disjoints qui contiennent chacun une et une seule racine).
2. Montrer qu'il y a une racine $\widehat{x}$ comprise entre 0 et 1.
3. Soit la méthode de point fixe
$$\begin{cases} x_{k+1} = \phi(x_k), \\ x_0 \in]0,1[, \end{cases} \tag{1.7}$$
avec ϕ l'application de $\mathbb{R}$ dans $\mathbb{R}$ définie par $\phi(x) = \frac{\sqrt{\exp(x^2)}}{2}$. Examiner la convergence de cette méthode et en préciser l'ordre de convergence.
4. Écrire la méthode de NEWTON pour la recherche des zéros de la fonction f.
5. Entre la méthode de NEWTON et la méthode de point fixe (1.7), quelle est la plus efficace ? Justifier la réponse.

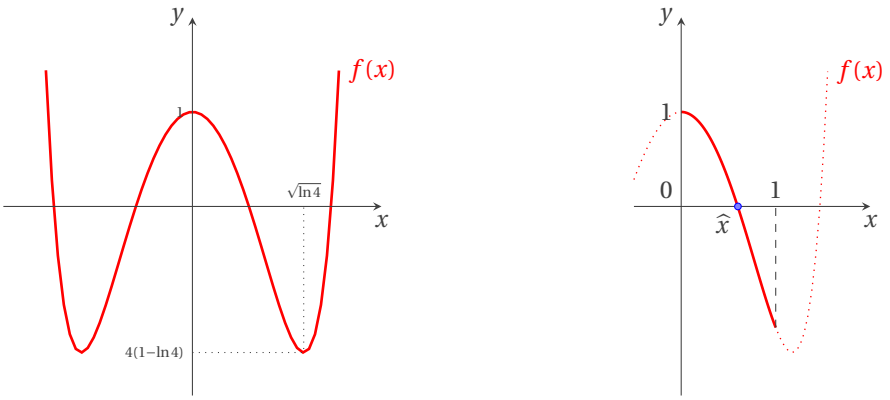
CORRECTION DE L'EXERCICE 1.26. On cherche les zéros de la fonction $f(x) = \exp(x^2) - 4x^2$.

1. On remarque que $f(-x) = f(x)$: la fonction est paire. On fait donc une brève étude sur $[0, +\infty[$:
   - ⋆ $f(0) = 1$ et $\lim_{x\to+\infty} f(x) = +\infty$,
   - ⋆ $f'(x) = 0$ pour $x = 0$ et $x = \sqrt{\ln 4}$ et on a $f(0) = 1$ et $f(\sqrt{\ln 4}) = 4(1 - \ln 4) < 0$; f est croissante pour $x > \sqrt{\ln 4}$ et décroissante pour $0 < x < \sqrt{\ln 4}$.

   On a
   - ⋆ une racine dans l'intervalle $]-\infty, -\sqrt{\ln 4}[$,
   - ⋆ une racine dans l'intervalle $]-\sqrt{\ln 4}, 0[$,
   - ⋆ une racine dans l'intervalle $]0, \sqrt{\ln 4}[$,
   - ⋆ une racine dans l'intervalle $]\sqrt{\ln 4}, \infty[$.

   Voir la figure 1.6a pour le graphe de f sur $\mathbb{R}$.
2. Puisque $f(0) = 1 > 0$ et $f(1) = e - 4 < 0$, pour le théorème des valeurs intermédiaires il existe au moins un $\widehat{x} \in]0,1[$ tel que $f(\widehat{x}) = 0$. Puisque $f'(x) = 2x\exp(x^2) - 8x = 2x(\exp(x^2) - 2^2) < 2x(e-4) < 0$ pour tout $x \in]0,1[$, ce $\widehat{x}$ est unique. Voir la figure 1.6b.
3. Étude de la convergence de la méthode (1.7) :
   3.1. pour tout x dans $]0,1[$ on a
   $$0 < \sqrt{\frac{\exp(x^2)}{4}} < \sqrt{\frac{e}{4}} < 1$$
   donc $\phi :]0,1[\to]0,1[$;
   3.2. $\phi \in \mathscr{C}^1(]0,1[)$;
   3.3. pour tout x dans $]0,1[$ on a
   $$|\phi'(x)| = \left|\frac{x\sqrt{\exp(x^2)}}{2}\right| = \left|x\phi(x)\right| < |x| < 1$$
   donc ϕ est contractante.

   Alors la méthode (1.7) converge vers $\widehat{x}$ point fixe de ϕ. De plus, pour tout $\widehat{x} \in]0,1[$,
$$\widehat{x} = \phi(\widehat{x}) \iff 2\widehat{x} = \sqrt{\exp(\widehat{x}^2)} \iff 4\widehat{x}^2 = \exp(\widehat{x}^2) \iff f(\widehat{x}) = 0;$$



(a) Graphe de $f(x) = \exp(x^2) - 4x^2$. (b) Zoom.

FIGURE 1.6.: Exercice 1.26

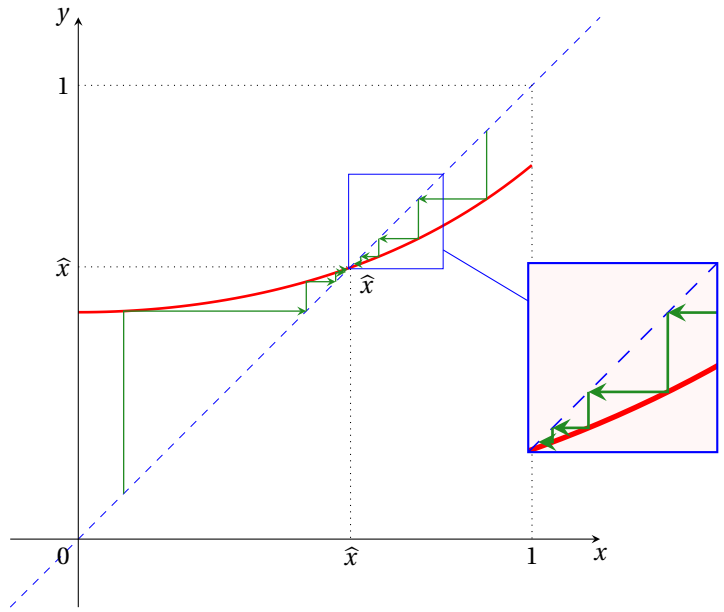


FIGURE 1.7.: Exercice 1.26 : convergence de la méthode de point fixe.

donc $\hat{x}$, point fixe de ϕ, est un zéro de f.
Étant donné que

$$\phi'(\hat{x}) = \hat{x}\phi(\hat{x}) = \hat{x}^2 \neq 0,$$

la méthode de point fixe (1.7) converge seulement à l'ordre 1.

4. La méthode de NEWTON est une méthode de point fixe avec $\phi(x) = x - \frac{f(x)}{f'(x)}$. Ici donc elle s'écrit

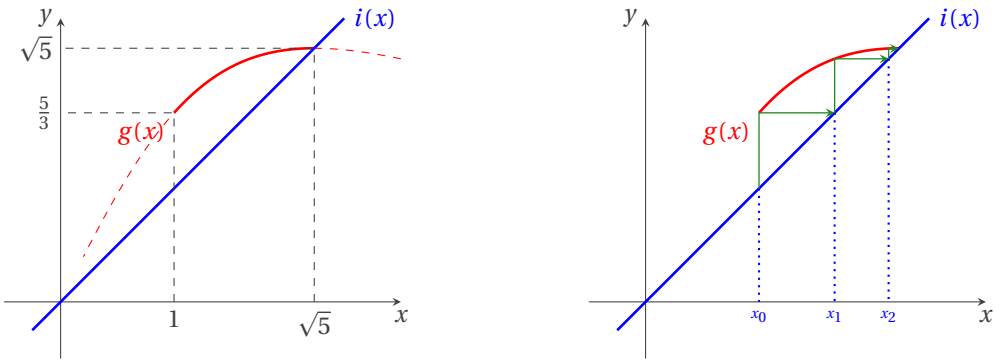
$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} = x_k - \frac{\exp(x_k^2) - 4x_k^2}{2x_k\exp(x_k^2) - 8x_k} = x_k - \frac{\exp(x_k^2) - 4x_k^2}{2x_k(\exp(x_k^2) - 4)}.$$

5. Puisque $\hat{x}$ est une racine simple de f, la méthode de NEWTON converge à l'ordre 2 tandis que la méthode de point fixe (1.7) converge seulement à l'ordre 1 : la méthode de NEWTON est donc plus efficace.

Exercice 1.27

On cherche à évaluer $\sqrt{5}$ à l'aide d'un algorithme n'autorisant que les opérations élémentaires. Soit $(x_n)_{n\in\mathbb{N}}$ la suite définie par récurrence

$$\begin{cases} x_0 = 1, \\ x_{n+1} = \dfrac{10x_n}{x_n^2 + 5} \quad \forall n \in \mathbb{N}. \end{cases}$$



(a) Graphe de g comparé au graphe de $i(x) = x$.

(b) Étude graphique de la convergence de la méthode de point fixe $x_{k+1} = g(x_k)$.

FIGURE 1.8.: Exercice 1.27

1. Montrer que si la suite converge, alors elle converge vers 0 ou $\sqrt{5}$.
2. Soit la fonction g définie sur $[1;\sqrt{5}]$ par $g(x) = \frac{10x}{x^2+5}$. Étudier g et la comparer à l'identité.
3. Montrer que la suite $(x_n)_{n\in\mathbb{N}}$ est croissante et majorée par $\sqrt{5}$. Conclure.
4. Déterminer l'ordre de convergence de cette suite.

CORRECTION DE L'EXERCICE 1.27.

1. Supposons qu'il existe $\ell \in \mathbb{R}$ tel que $x_n \xrightarrow[n\to+\infty]{} \ell$.
   - ⋆ Par définition de convergence on a $\ell = \frac{10\ell}{\ell^2+5}$ et par conséquent $\ell \in \{-\sqrt{5}, 0, \sqrt{5}\}$.
   - ⋆ On prouve par récurrence que
     - ⋆ si $x_0 = 0$ alors $x_n = 0$ pour tout $n \in \mathbb{N}$ donc $\ell = 0$,
     - ⋆ si $x_0 > 0$ alors $x_n > 0$ pour tout $n \in \mathbb{N}$ donc $\ell \geq 0$,
     - ⋆ si $x_0 < 0$ alors $x_n < 0$ pour tout $n \in \mathbb{N}$ donc $\ell \leq 0$.

   Comme $x_0 = 1 > 0$, alors $x_n > 0$ pour tout $n \in \mathbb{N}$ et $\ell \in \{0, \sqrt{5}\}$.
2. Soit la fonction g définie sur $[1;\sqrt{5}]$ par $g(x) = \frac{10x}{x^2+5}$. On étudie la fonction g :
   - ⋆ $g(x) > 0$ pour tout $x \in [1;\sqrt{5}]$;
   - ⋆ $g(1) = \frac{5}{3}$, $g(\sqrt{5}) = \sqrt{5}$;
   - ⋆ $g'(x) = -10\frac{x^2-5}{(x^2+5)^2}$;
   - ⋆ g est croissante sur $[1;\sqrt{5}[$ et $g'(\sqrt{5}) = 0$.

   Graphe de g comparé au graphe de $i(x) = x$: voir la figure 1.8a. On vérifie analytiquement qu'il existe une et une seule intersection entre la courbe d'équation $y = g(x)$ et la droite d'équation $y = x$ dans $[1;\sqrt{5}]$:

$$g(x) = x \iff \frac{10x}{x^2+5} = x \iff x^2 = 5.$$

3. On a $g(x) \in [5/3;\sqrt{5}]$ pour tout $x \in [1;\sqrt{5}]$ et on a vu au point précédent que g est croissante et $g(\sqrt{5}) = \sqrt{5}$.
   De plus, $g(x) \geq x$ car

$$g(x) = \frac{10x}{x^2+5} \geq \frac{10x}{(\sqrt{5})^2+5} = x,$$

   par conséquent la suite $x_{k+1} = g(x_k) \geq x_k$ est croissante.
   Comme $g(x) \leq (\sqrt{5}) = \sqrt{5}$ alors la suite $x_{k+1} = g(x_k) \leq \sqrt{5}$ est bornée. On a ainsi une suite croissante et borné, ce qui implique qu'elle converge. Comme au premier point on a montré que si elle converge vers ℓ alors $\ell \in \{0, \sqrt{5}\}$, on conclut que $x_n \xrightarrow[n\to+\infty]{} \sqrt{5}$. Pour l'étude graphique de la convergence de la méthode de point fixe voir la figure 1.8b.
   Dans ce cas, on ne peut pas utiliser le théorème de point fixe pour prouver la convergence de la suite sur l'intervalle $[1;\sqrt{5}]$. En effet
   - ⋆ g est au moins de classe $\mathscr{C}^1([1;\sqrt{5}])$
   - ⋆ $g([1;\sqrt{5}]) = [5/3;\sqrt{5}] \subset [1;\sqrt{5}]$
   - ⋆ mais $0 \leq g'(x) < 1$ ssi $x \in [\sqrt{-10+5\sqrt{5}};\sqrt{5}]$ (et on a $\sqrt{-10+5\sqrt{5}} > 1$).

En revanche, on peut utiliser le théorème de point fixe pour prouver la convergence de la suite sur l'intervalle $[5/3;\sqrt{5}]$ car

- ⋆ g est au moins de classe $\mathscr{C}^1([5/3;\sqrt{5}])$
- ⋆ $g([5/3;\sqrt{5}]) \subset [5/3;\sqrt{5}]$
- ⋆ $0 \leq g'(x) < 1$ pour tout $x \in [5/3;\sqrt{5}]$.

4. Comme $g'(\sqrt{5}) = 0$ et $g''(\sqrt{5}) \neq 0$, la méthode de point fixe associée à la fonction d'itération g est d'ordre 2.

Exercice 1.28

L'objectif de cet exercice est de déterminer le zéro d'une fonction $\mathscr{C}^2(\mathbb{R},\mathbb{R})$ vérifiant $-2 < f'(x) < -1$ sur $\mathbb{R}$. On définit la suite $\{x_n\}_{n\in\mathbb{N}}$ de $\mathbb{R}$ par la récurrence suivante

$$x_{n+1} = g(x_n) = x_n + \alpha f(x_n),$$

où $\alpha > 0$ et $x_0 \in \mathbb{R}$ sont donnés.

1. Montrer que $\lim_{x\to-\infty} f(x) = +\infty$ et $\lim_{x\to+\infty} f(x) = -\infty$.
2. En déduire qu'il existe un unique ℓ élément de $\mathbb{R}$ tel que $f(\ell) = 0$.
3. Montrer que si $0 < \alpha < 1$, la fonction g définie par $g(x) = x + \alpha f(x)$ vérifie
$$-1 < 1 - 2\alpha < g'(x) < 1 - \alpha < 1 \qquad \text{sur } \mathbb{R}.$$
4. En déduire la convergence de la suite $\{x_n\}_{n\in\mathbb{N}}$ si $0 < \alpha < 1$ pour tout $x_0 \in \mathbb{R}$.
5. La suite converge-t-elle pour $\alpha = -\frac{1}{f'(\ell)}$?
6. Donner l'ordre de convergence de la suite $\{x_n\}_{n\in\mathbb{N}}$ pour $0 < \alpha < 1$ en distinguant le cas $\alpha = \frac{1}{f'(\ell)}$.
7. Peut-on choisir $\alpha = -\frac{1}{f'(\ell)}$ d'un point de vue pratique ?
8. On choisit alors d'approcher $\alpha = -\frac{1}{f'(\ell)}$ par $\alpha_n = -\frac{1}{f'(x_n)}$ et la suite $\{x_n\}_{n\in\mathbb{N}}$ est définie par
$$x_{n+1} = g(x_n) = x_n + \alpha_n f(x_n).$$
Quel est le nom de cette méthode itérative ? Montrer que la suite $\{x_n\}_{n\in\mathbb{N}}$ converge quel que soit $x_0 \in \mathbb{R}$.

CORRECTION DE L'EXERCICE 1.28.

1. Puisque f est de classe $\mathscr{C}^2(\mathbb{R},\mathbb{R})$ et $f'(x) < 0$ sur $\mathbb{R}$ alors f est monotone décroissante.
De plus, puisque $-2 < f'(x) < -1$ sur $\mathbb{R}$, on obtient :
⋆ si $x > 0$ alors $f(x) = \int_0^x f'(x)\,\mathrm{d}x + f(0) < \int_0^x -1\,\mathrm{d}x + f(0) = -x + f(0) \xrightarrow[x\to+\infty]{} -\infty$,
⋆ si $x < 0$ alors $f(x) = \int_0^x f'(x)\,\mathrm{d}x + f(0) = -\int_x^0 f'(x)\,\mathrm{d}x + f(0) > -\int_x^0 -2\,\mathrm{d}x + f(0) = -2x + f(0) \xrightarrow[x\to-\infty]{} +\infty$.
donc
$$\lim_{x\to-\infty} f(x) = +\infty \qquad \lim_{x\to+\infty} f(x) = -\infty.$$
NB : seul la condition $f'(x) < -1$ permet de conclure car une fonction peut être monotone décroissante mais avoir une limite finie !
2. Puisque $\lim_{x\to-\infty} f(x) = +\infty > 0$ et $\lim_{x\to+\infty} f(x) = -\infty < 0$, pour le théorème des valeurs intermédiaires il existe au moins un $\ell \in \mathbb{R}$ tel que $f(\ell) = 0$. Puisque $f'(x) < 0$ pour tout $x \in \mathbb{R}$, ce ℓ est unique.
3. Considérons la fonction g définie par $g(x) = x + \alpha f(x)$ alors g est de classe $\mathscr{C}^2(\mathbb{R},\mathbb{R})$ et
$$g'(x) = 1 + \alpha f'(x) \qquad \text{sur } \mathbb{R}.$$
Puisque $f'(x) < -1$ et $0 < \alpha < 1$ on a
$$g'(x) < 1 - \alpha < 1 \qquad \text{sur } \mathbb{R}$$
et puisque $f'(x) > -2$ et $0 < \alpha < 1$ alors
$$g'(x) > 1 - 2\alpha > -1 \qquad \text{sur } \mathbb{R}.$$
Autrement dit
$$|g'(x)| < 1 \qquad \text{sur } \mathbb{R}.$$
4. Soit $0 < \alpha < 1$. On étudie la suite
$$x_{n+1} = g(x_n)$$
et on va vérifier qu'il s'agit d'une méthode de point fixe pour le calcul du zéro ℓ de f.

4.1. On vérifie d'abord que, si la suite converge vers un point fixe de g, ce point est bien un zéro de f (ici le réciproque est vrai aussi) : soit $\ell \in \mathbb{R}$, alors

$$\ell = g(\ell) \iff \ell = \ell + \alpha f(\ell) \iff 0 = \alpha f(\ell) \iff f(\ell) = 0;$$

4.2. vérifions maintenant que la suite converge vers un point fixe de g (et donc, grâce à ce qu'on a vu au point précédant, elle converge vers l'unique zéro de f) :
4.2.1. on a évidemment que $g : \mathbb{R} \to \mathbb{R}$;
4.2.2. on a déjà remarqué que $g \in \mathscr{C}^1(\mathbb{R}, \mathbb{R})$;
4.2.3. pour tout x dans $\mathbb{R}$ on a prouvé que $|g'(x)| < 1$, *i.e.* que g est contractante.

Alors la suite $x_{n+1} = g(x_n)$ converge vers ℓ point fixe de g et zéro de f.

5. Si $\alpha = -\frac{1}{f'(\ell)}$ alors
$$x_{n+1} = g(x_n) = x_n - \frac{f(x_n)}{f'(\ell)},$$
qui converge car $-2 < f'(\ell) < -1$ ssi $\frac{1}{2} < \alpha < 1$ et donc on rentre dans le cas de $0 < \alpha < 1$.
6. Étant donné que
$$g'(\ell) = 1 + \alpha f'(\ell)$$
   - ⋆ la méthode de point fixe converge à l'ordre 2 si $\alpha f'(\ell) = -1$,
   - ⋆ la méthode de point fixe converge à l'ordre 1 si $-2 < \alpha f'(\ell) < 0$ mais $\alpha f'(\ell) \neq -1$,
   - ⋆ la méthode de point fixe ne converge pas si $\alpha f'(\ell) < -2$ ou $\alpha f'(\ell) > 0$.

   Étant donné que $-2 < f'(\ell) < -1$ et que $0 < \alpha < 1$ on peut conclure que
   - ⋆ la méthode de point fixe converge à l'ordre 2 si $\alpha = -\frac{1}{f'(\ell)}$,
   - ⋆ la méthode de point fixe converge à l'ordre 1 si $\alpha \neq -\frac{1}{f'(\ell)}$.
7. D'un point de vue pratique on ne peut pas choisir $\alpha = -\frac{1}{f'(\ell)}$ car on ne connaît pas ℓ.
8. Si on choisit d'approcher $\alpha = -\frac{1}{f'(\ell)}$ par $\alpha_n = -\frac{1}{f'(x_n)}$ et on considère la suite $\{x_n\}_{n\in\mathbb{N}}$ définie par
$$x_{n+1} = g(x_n) = x_n + \alpha_n f(x_n),$$
on obtient la méthode de NEWTON (qui est d'ordre 2).
De plus, comme $-2 < f'(x) < -1$ on rentre dans le cas $0 < \alpha < 1$ donc la suite $\{x_n\}_{n\in\mathbb{N}}$ converge quel que soit $x_0 \in R$.

Exercice 1.29

L'objectif de cet exercice est de déterminer le zéro d'une fonction $f \in \mathscr{C}^2(\mathbb{R}, \mathbb{R})$ vérifiant $1 < f'(x) < 2$ sur $\mathbb{R}$. On définit la suite $\{x_n\}_{n\in\mathbb{N}}$ de $\mathbb{R}$ par la récurrence suivante
$$x_{n+1} = g(x_n),$$
où $\alpha > 0$ et $x_0 \in \mathbb{R}$ sont donnés et la fonction $g : \mathbb{R} \to \mathbb{R}$ est définie par $g(x) = x - \alpha f(x)$.

1. Montrer que $\lim_{x\to-\infty} f(x) = -\infty$, $\lim_{x\to+\infty} f(x) = +\infty$ et en déduire qu'il existe un unique $\ell \in \mathbb{R}$ tel que $f(\ell) = 0$.
2. Montrer que si $0 < \alpha < 1$, la fonction g vérifie $|g'(x)| < 1$ sur $\mathbb{R}$. En déduire la convergence de la suite $\{x_n\}_{n\in\mathbb{N}}$ pour tout $\alpha \in]0;1[$ quel que soit $x_0 \in \mathbb{R}$.
3. Donner l'ordre de convergence de la suite $\{x_n\}_{n\in\mathbb{N}}$ en fonction de $\alpha \in]0;1[$.
4. Comme d'un point de vue pratique on ne peut pas choisir $\alpha = \frac{1}{f'(\ell)}$, on va l'approcher par $\alpha_n = \frac{1}{f'(x_n)}$ et on obtient la suite $\{x_n\}_{n\in\mathbb{N}}$ définie par
$$x_{n+1} = x_n - \alpha_n f(x_n).$$
Quel est le nom de cette méthode itérative ? Montrer que la suite $\{x_n\}_{n\in\mathbb{N}}$ converge quel que soit $x_0 \in \mathbb{R}$.

CORRECTION DE L'EXERCICE 1.29.

1. Puisque f est de classe $\mathscr{C}^2(\mathbb{R}, \mathbb{R})$ et $f'(x) > 0$ sur $\mathbb{R}$ alors f est monotone croissante.
De plus, puisque $1 < f'(x) < 2$, on obtient :
⋆ si $x > 0$ alors $f(x) = \int_0^x f'(x)\,\mathrm{d}x > \int_0^x 1\,\mathrm{d}x = x \xrightarrow[x\to+\infty]{} +\infty$,
⋆ si $x < 0$ alors $f(x) = \int_0^x f'(x)\,\mathrm{d}x = -\int_x^0 f'(x)\,\mathrm{d}x < -\int_x^0 2\,\mathrm{d}x = 2x \xrightarrow[x\to-\infty]{} -\infty$.
donc
$$\lim_{x\to-\infty} f(x) = -\infty \qquad \lim_{x\to+\infty} f(x) = +\infty.$$

NB : seul la condition $1 < f'(x) < 2$ permet de conclure car une fonction peut être monotone croissante mais avoir une limite finie !

Puisque $\lim_{x\to-\infty} f(x) = -\infty < 0$ et $\lim_{x\to+\infty} f(x) = +\infty > 0$, pour le théorème des valeurs intermédiaires il existe au moins un $\ell \in \mathbb{R}$ tel que $f(\ell) = 0$. Puisque $f'(x) > 0$ pour tout $x \in \mathbb{R}$, ce ℓ est unique.

2. g est de classe $\mathscr{C}^2(\mathbb{R},\mathbb{R})$. Puisque $1 < f'(x) < 2$ et $0 < \alpha < 1$ on a

$$-1 < 1 - 2\alpha < g'(x) = 1 - \alpha f'(x) < 1 - \alpha < 1$$

Autrement dit

$$|g'(x)| < 1 \qquad \text{sur } \mathbb{R}.$$

On étudie alors la suite

$$x_{n+1} = g(x_n)$$

et on va vérifier qu'il s'agit d'une méthode de point fixe pour le calcul du zéro ℓ de f.

2.1. On vérifie d'abord que, si la suite converge vers un point fixe de g, ce point est bien un zéro de f (ici le réciproque est vrai aussi) : soit $\ell \in \mathbb{R}$, alors

$$\ell = g(\ell) \iff \ell = \ell - \alpha f(\ell) \iff 0 = \alpha f(\ell) \overset{\alpha\neq 0}{\iff} f(\ell) = 0;$$

2.2. vérifions maintenant que la suite converge vers un point fixe de g (et donc, grâce à ce qu'on a vu au point précédant, elle converge vers l'unique zéro de f) : $g \in \mathscr{C}^1(\mathbb{R},\mathbb{R})$ et pour tout x dans $\mathbb{R}$ on a prouvé que $|g'(x)| < 1$, *i.e.* g est contractante, alors la suite $x_{n+1} = g(x_n)$ converge vers ℓ point fixe de g et zéro de f.

3. Étant donné que

$$g'(\ell) = 1 - \alpha f'(\ell)$$

avec $0 < f'(\ell) < 2$ et $0 < \alpha < 1$, on peut conclure que

- ⋆ la méthode de point fixe converge à l'ordre 2 si $\alpha = \frac{1}{f'(\ell)}$,
- ⋆ la méthode de point fixe converge à l'ordre 1 si $\alpha \neq \frac{1}{f'(\ell)}$.

4. D'un point de vue pratique on ne peut pas choisir $\alpha = \frac{1}{f'(\ell)}$ car on ne connaît pas ℓ. Si on choisit d'approcher $\alpha = \frac{1}{f'(\ell)}$ par $\alpha_n = \frac{1}{f'(x_n)}$ et on considère la suite $\{x_n\}_{n\in\mathbb{N}}$ définie par

$$x_{n+1} = x_n - \alpha_n f(x_n),$$

on obtient la méthode de NEWTON (qui est d'ordre 2).

De plus, comme $1 < f'(x) < 2$ alors $0 < \alpha_n < 1$ donc la suite $\{x_n\}_{n\in\mathbb{N}}$ converge quel que soit $x_0 \in \mathbb{R}$.

Exercice 1.30

Soit g la fonction définie sur $\mathbb{R}_+^*$ par

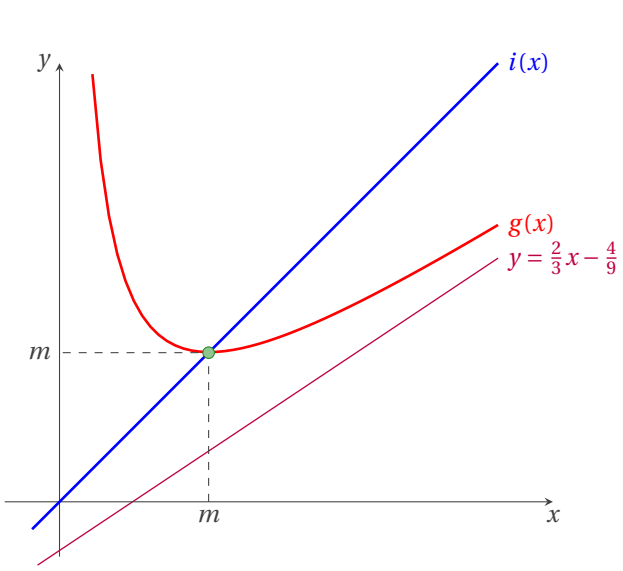
$$g(x) = \frac{2x^3 + 4x^2 + 10}{3x^2 + 8x}.$$

1. Faire l'étude complète de la fonction g. (On admettra que $x^3 + 4x^2 - 10 = 0$ admet comme unique solution $m \approx 1{,}36$ et que $g(m) = m$.)
2. Comparer g à l'identité.
3. Soit la suite $(x_n)_{n\in\mathbb{N}}$ définie par

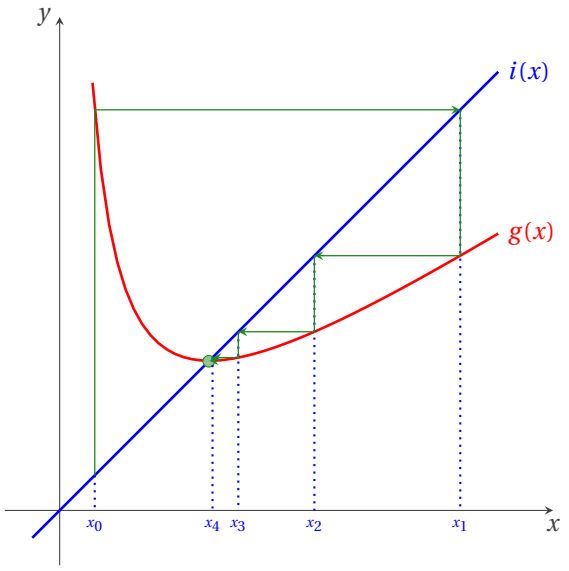
$$x_{n+1} = g(x_n), \qquad x_0 > 0.$$

À l'aide des graphe de g et de l'identité sur $\mathbb{R}_+^*$, dessiner la suite $(x_n)_{n\in\mathbb{N}}$ sur l'axe des abscisses. Observer graphiquement la convergence. En particulier, montrer que cette suite est décroissante à partir du rang 1.

4. Expliciter (sans la vérifier) la condition nécessaire pour la convergence observée graphiquement.
5. Écrire l'algorithme défini par la suite $(x_n)_{n\in\mathbb{N}}$ qui permet de déterminer le point fixe à une précision de ε.
6. Expliciter la méthode de NEWTON pour la recherche du zéro de la fonction f définie par $f(x) = x^3 + 4x^2 - 10$. Que remarque-t-on ?
7. Donner l'ordre de convergence de la suite.



(a) Graphe de g comparé au graphe de i.



(b) Étude graphique de la convergence de la méthode de point fixe.

FIGURE 1.9.

CORRECTION DE L'EXERCICE 1.30.

1. Étude de la fonction $g : \mathbb{R}_+^* \to \mathbb{R}$ définie par $g(x) = \frac{2x^3+4x^2+10}{3x^2+8x}$:
   - ⋆ $g(x) > 0$ pour tout $x \in \mathbb{R}_+^*$;
   - ⋆ $\lim\limits_{x\to 0^+} g(x) = \lim\limits_{x\to+\infty} g(x) = +\infty$;
     $\lim\limits_{x\to+\infty} \frac{g(x)}{x} = \frac{2}{3}$ et $\lim\limits_{x\to+\infty} g(x) - \frac{2}{3}x = -\frac{4}{9}$ donc $y = \frac{2}{3}x - \frac{4}{9}$ est un asymptote ;
   - ⋆ $g'(x) = \frac{2(3x+4)(x^3+4x^2-10)}{x^2(3x+8)^2}$;
   - ⋆ g est croissante sur $[m, +\infty[$, décroissante sur $[0, m]$ où $m \approx 1{,}36$;
   - ⋆ $x = m$ est un minimum absolu et $g(m) = m$.

x	0		m		$+\infty$
$g'(x)$		$-$		$+$	
$g(x)$	$+\infty$	↘	m	↗	$+\infty$

2. Graphe de g comparé au graphe de $i(x) = x$: voir la figure 1.9a. On vérifie analytiquement qu'il existe une et une seule intersection entre la courbe d'équation $y = g(x)$ et la droite d'équation $y = x$:

$$g(x) = x \iff \frac{2x^3+4x^2+10}{3x^2+8x} = x \iff x^3+4x^2-10 = 0 \iff x = m \iff f(x) = 0.$$

3. Pour l'étude graphique de la convergence de la méthode de point fixe voir la figure 1.9b.
4. On en déduit que pour tout $x > 0$ on a $g(x) \geq m$. Donc, pour tout $k > 0$, $x_k = g(x_{k-1}) \geq m$. Pour étudier la convergence de la méthode vérifions si on peut appliquer le théorème de point fixe :
   - 4.1. pour tout x dans $[m, +\infty[$ on a $g(x) > m$ donc $g([m, +\infty[) \subset [m, +\infty[$;
   - 4.2. $g \in \mathscr{C}^1([m, +\infty[)$;
   - 4.3. pour tout x dans $[m, +\infty[$, on a $|g'(x)| = \left|\frac{(6x^2+8x)-g(x)(6x+8)}{3x^2+8x}\right| < 1$ alors g est contractante.

   Si les conditions précédentes sont vérifiées alors la méthode converge vers m point fixe de g. De plus, pour tout $\alpha \in [m, +\infty[: \alpha = g(\alpha) \iff \alpha = m$ donc le point fixe de g est racine de f.
5. Algorithme de point fixe :

   Require: $x_0 > 0$, $g : x \mapsto g(x)$

while $|x_{k+1} - x_k| > \varepsilon$ **do**
  $x_{k+1} \leftarrow g(x_k)$
  $k \leftarrow k+1$
end while

6. La méthode de NEWTON est une méthode de point fixe avec $g(x) = x - \frac{f(x)}{f'(x)}$. Ici donc elle s'écrit

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} = x_k - \frac{x_k^3 + 4x_k^2 - 10}{3x_k^2 + 8x_k} = g(x_k)$$

autrement dit la méthode de point fixe assignée est la méthode de NEWTON.

7. Étant donné que la méthode de point fixe donnée est la méthode de NEWTON et que la racine m de f est simple, elle converge à l'ordre 2.

Quelques remarques à propos du critère d'arrêt basé sur le contrôle de l'incrément. Les itérations s'achèvent dès que $|x_{k+1} - x_k| < \varepsilon$; on se demande si cela garantît-t-il que l'erreur absolue e_{k+1} est elle aussi inférieur à ε. L'erreur absolue à l'itération $(k+1)$ peut être évaluée par un développement de TAYLOR au premier ordre

$$e_{k+1} = |g(\widehat{x}) - g(x_k)| = |g'(z_k)e_k|$$

avec z_k compris entre m et x_k. Donc

$$|x_{k+1} - x_k| = |e_{k+1} - e_k| = |g'(z_k) - 1|e_k \simeq |g'(m) - 1|e_k.$$

Puisque $g'(x) = 2\frac{3x+4}{x^2(3x+8)^2} f(x)$, alors $g'(m) = 0$ donc on a bien $|x_{k+1} - x_k| \simeq e_k$.

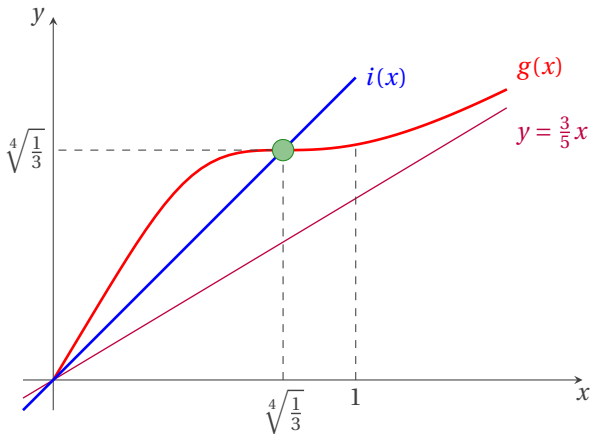
Exercice 1.31

On se propose de calculer $\sqrt[4]{\frac{1}{3}}$ en trouvant les racines réelles de l'application f de $\mathbb{R}$ dans $\mathbb{R}$ définie par $f(x) = x^4 - \frac{1}{3}$.

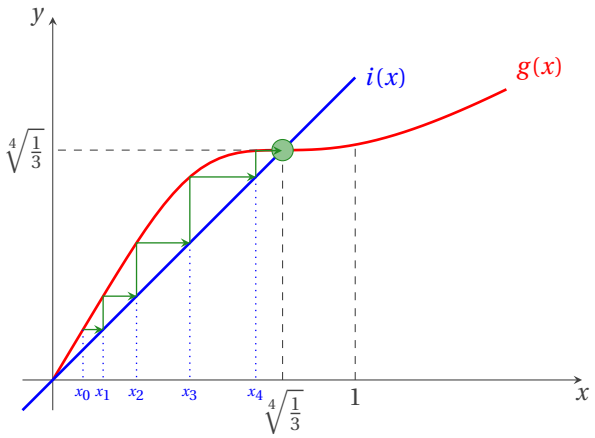
1. Situer les 2 racines de f (i.e. indiquer 2 intervalles disjoints qui contiennent chacun une et une seule racine). En particulier, montrer qu'il y a une racine $\widehat{x}$ comprise entre 0 et 1.
2. Soit g la fonction définie sur $[0;1]$ par
$$g(x) = \frac{x(9x^4+5)}{3(5x^4+1)}.$$
   2.1. Faire l'étude complète de la fonction g et la comparer à l'identité.
   2.2. Soit la suite $(x_n)_{n\in\mathbb{N}}$ définie par
$$x_{n+1} = g(x_n), \qquad x_0 \in]0;1[.$$
   À l'aide des graphe de g et de l'identité sur $[0;1]$, dessiner la suite $(x_n)_{n\in\mathbb{N}}$ sur l'axe des abscisses. Observer graphiquement la convergence.
   2.3. Justifier mathématiquement la convergence observée graphiquement.
   2.4. Calculer l'ordre de convergence de la suite.
   2.5. Écrire l'algorithme défini par la suite $(x_n)_{n\in\mathbb{N}}$ qui permet de déterminer $\sqrt[4]{\frac{1}{3}}$ à une précision de ε.
3. Expliciter la méthode de NEWTON pour la recherche du zéro de la fonction f.
4. Entre la méthode de NEWTON et la méthode de point fixe $x_{k+1} = g(x_k)$, quelle est la plus efficace ? Justifier la réponse.

CORRECTION DE L'EXERCICE 1.31.

1. f est paire ; comme $f'(x) = 4x^3$, f est croissante pour $x > 0$ et décroissante pour $x < 0$; puisque $f(0) < 0$ et $f(-1) = f(1) > 0$, on conclut que il n'y a que deux racines réelles distinctes : $\widehat{x} \in]0;1[$ et $-\widehat{x} \in]-1;0[$.
2. On étudie la fonction $g(x) = \frac{x(9x^4+5)}{3(5x^4+1)}$ pour $x \geq 0$.
   2.1. ⋆ $g(x) \geq 0$ pour tout $x \geq 0$ et $g(x) = 0$ ssi $x = 0$;
   ⋆ $g'(x) = \frac{5(9x^8-6x^4+1)}{3(5x^4+1)^2} = \frac{5}{3}\left(\frac{3x^4-1}{5x^4+1}\right)^2$ donc $g'(x) \geq 0$ pour tout $x \in]0;1[$ et $g'(x) = 0$ ssi $x = \sqrt[4]{\frac{1}{3}}$. De plus, $g\left(\sqrt[4]{\frac{1}{3}}\right) = \sqrt[4]{\frac{1}{3}}$.
   ⋆ Enfin, $g''(x) = \frac{10}{3}\frac{3x^4-1}{5x^4+1}\frac{32x^3}{(5x^4+1)^2} = \sqrt{\frac{20}{3}g'(x)}\frac{32x^3}{(5x^4+1)^2} = \frac{320x^3(3x^4-1)}{(5x^4+1)^3}$ donc $g''(x) = 0$ ssi $x = 0$ ou $x = \sqrt[4]{\frac{1}{3}}$, g est concave pour $x \in \left]0; \sqrt[4]{\frac{1}{3}}\right[$, convexe pour $x > \sqrt[4]{\frac{1}{3}}$.



(a) Graphe de g comparé au graphe de i.



(b) Étude graphique de la convergence de la méthode de point fixe.

FIGURE 1.10.

- ⋆ Pour le graphe de g comparé au graphe de $i(x) = x$ pour $x \in [0;1]$ voir la figure 1.10a.
- ⋆ On vérifie analytiquement qu'il existe une et une seule intersection entre la courbe d'équation $y = g(x)$ et la droite d'équation $y = x$:

$$g(x) = x \iff \frac{x(9x^4+5)}{3(5x^4+1)} = x \iff 9x^4+5 = 3(5x^4+1) \iff x^4 = \frac{1}{3} \iff f(x) = 0.$$

2.2. Pour l'étude graphique de la convergence de la méthode de point fixe voir la figure 1.10b.

2.3. Étudions la convergence de la méthode. On remarque que

$$\frac{x_{k+1}}{x_k} = \frac{9x_k^4+5}{3(5x_k^4+1)} > 1 \iff x_k < \sqrt[4]{\frac{1}{3}}$$

donc la suite récurrente

$$\begin{cases} x_0 \in \left]0; \sqrt[4]{\frac{1}{3}}\right[\\ x_{k+1} = g(x_k) \end{cases}$$

est monotone croissante et majorée par $\sqrt[4]{\frac{1}{3}}$: elle est donc convergente vers $\ell \leq \sqrt[4]{\frac{1}{3}}$. Comme $\ell = g(\ell)$ ssi $\ell = \sqrt[4]{\frac{1}{3}}$, on conclut qu'elle converge vers $\sqrt[4]{\frac{1}{3}}$. De même, la suite récurrente

$$\begin{cases} x_0 \in \left]\sqrt[4]{\frac{1}{3}}; 0\right[\\ x_{k+1} = g(x_k) \end{cases}$$

est monotone décroissante et minoré par $\sqrt[4]{\frac{1}{3}}$: elle est donc convergente vers $\ell \leq \sqrt[4]{\frac{1}{3}}$. Comme $\ell = g(\ell)$ ssi $\ell = \sqrt[4]{\frac{1}{3}}$, on conclut qu'elle converge vers $\sqrt[4]{\frac{1}{3}}$.

Par conséquent, quelque soit le point initiale, la méthode de point fixe donnée converge vers $\sqrt[4]{\frac{1}{3}}$ point fixe de g (et racine de f).
Soulignons qu'on ne peut pas utiliser le théorème de point fixe pour prouver la convergence de la méthode car g n'est pas contractante sur $[0;1]$. En effet, dans $[0;1]$ on a

$$|g'(x)| < 1 \iff g'(x) < 1 \iff 5(3x^4-1)^2 < 3(5x^4+1)^2 \iff 15x^8+30x^4-1 > 0 \iff x^4 > -1+\sqrt{\frac{16}{15}} \in]0;1[.$$

2.4. Si on pose $\widehat{x} = \sqrt[4]{\frac{1}{3}}$ alors $g(\widehat{x}) = \widehat{x}$, $g'(\widehat{x}) = 0$, $g''(\widehat{x}) = 0$ et $g'''(\widehat{x}) = -320\widehat{x}^2 \frac{25\widehat{x}^8-22\widehat{x}^4+1}{(5\widehat{x}^4+1)^4} = \frac{15\sqrt{3}}{2}$: on conclut que la suite converge à l'ordre 3.

2.5. Algorithme de point fixe :
Require: $x_0 > 0$, $g\colon x \mapsto g(x)$

while $|x_{k+1} - x_k| > \varepsilon$ **do**
  $x_{k+1} \leftarrow g(x_k)$
  $k \leftarrow k+1$
end while

3. Entre la méthode de NEWTON et la méthode de point fixe $x_{k+1} = g(x_k)$, la plus efficace est la méthode de point fixe $x_{k+1} = g(x_k)$ car elle est d'ordre 3 tandis que celle de NEWTON n'est que d'ordre 2.

Exercice 1.32 *(Python)*

Comparer les méthodes de la dichotomie, de LAGRANGE et de NEWTON pour approcher la racine $\widehat{x} \simeq 0.5149332646611294$ de la fonction $f(x) = \cos^2(2x) - x^2$ sur l'intervalle $]0, 1.5[$ avec une précision de 10^{-16}. Pour la méthode de NEWTON on prendra $x_0 = 0.75$.

CORRECTION DE L'EXERCICE 1.32. On modifie les fonctions données à la page 24 pour que les méthodes s'arrêtent lorsque le nombre d'itérations est égal à `maxITER` :

```
import math, sys

def dichotomie(f,a,b,tol,maxITER):
⟶fa = f(a)
⟶if abs(fa)<=tol:
⟶⟶return a
⟶fb = f(b)
⟶if abs(fb)<=tol:
⟶⟶return b
⟶if fa*fb > 0.0:
⟶⟶print "La racine n'est pas encadree"
⟶⟶sys.exit(0)
⟶n = int(math.ceil(math.log(abs(b-a)/tol)/math.log(2.0)))
⟶for k in range(min(n+1,maxITER)):
⟶⟶c = (a+b)*0.5
⟶⟶fc = f(c)
⟶⟶if fc == 0.0:
⟶⟶⟶return c
⟶⟶if fc*fb < 0.0:
⟶⟶⟶a = c
⟶⟶⟶fa = fc
⟶⟶else:
⟶⟶⟶b = c
⟶⟶⟶fb = fc
⟶return (a+b)*0.5

def lagrange(f,a,b,tol,maxITER):
    fa = f(a)
    if abs(fa)<=tol:
            return a
    fb = f(b)
    if abs(fb)<=tol:
            return b
    if fa*fb > 0.0:
            print "La racine n'est pas encadree"
            sys.exit(0)
    k = 0
    fc = 2.*tol
    while ( (abs(b-a)>tol) and (abs(fc)>tol) and (k<maxITER) ):
        k += 1
        c = a-fa*(b-a)/(fb-fa)
        fc = f(c)
        if fc == 0.0:
            return c
        if fc*fb < 0.0:
            a = c
            fa = fc
        else:
```

```
            b = c
            fb = fc
    return a-fa*(b-a)/(fb-fa)

def newton(f,x_init,tol,maxITER):
⟶k = 0
⟶x = x_init
⟶fx = f(x)
⟶h = tol
⟶dfx = df(x)
⟶while ( (abs(fx)>tol) and (k<maxITER) ):
⟶⟶x = x - fx/dfx
⟶⟶fx = f(x)
⟶⟶dfx = df(x)
⟶⟶k += 1
⟶return x
```

Ensuite on construit une matrice dont la première colonne contient le nombre d'itérations, la deuxième colonne l'erreur absolue obtenue par la méthode de la dichotomie avec le nombre d'itérations indiqué dans la première colonne, la troisième colonne l'erreur absolue obtenue par la méthode de LAGRANGE et la dernière par la méthode de NEWTON.

```
def f(x):
    return (math.cos(2.*x))**2-x**2
def df(x):
    return -4.*math.cos(2.*x)*math.sin(2.*x)-2.*x

exact = 0.5149332646611294

nITER = 10
tol = sys.float_info.epsilon
a = 0.
b = 1.5
x_init = 0.75


XXX = []
Dic = []
Lag = []
New = []

for i in range(nITER):
    maxITER = i
    XXX.append(maxITER)
    Dic.append(abs(exact-dichotomie(f,a,b,tol,maxITER)))
    Lag.append(abs(exact-lagrange(f,a,b,tol,maxITER)))
    New.append(abs(exact-newton(f,x_init,tol,maxITER)))
    print "%2.g %15.17f %15.17f %15.17f" % (XXX[i], Dic[i], Lag[i], New[i])
```

On obtient ainsi le tableau

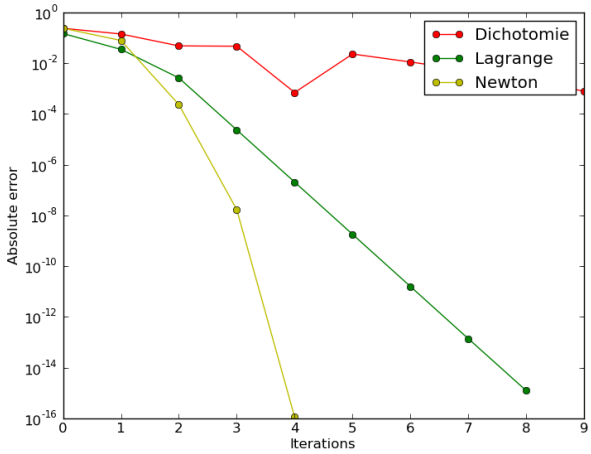
maxITER	Dichotomie	LAGRANGE	NEWTON
0	0.23506673533887057	0.14588447288050665	0.23506673533887057
1	0.13993326466112943	0.03464350570111258	0.07773975719741250
2	0.04756673533887057	0.00260088757041255	0.00023079676801208
3	0.04618326466112943	0.00002318596617201	0.00000001670020067
4	0.00069173533887057	0.00000020304484416	0.00000000000000011
5	0.02274576466112943	0.00000000177782544	0.00000000000000000
6	0.01102701466112943	0.00000000001556633	0.00000000000000000
7	0.00516763966112943	0.00000000000013634	0.00000000000000000
8	0.00223795216112943	0.00000000000000122	0.00000000000000000
9	0.00077310841112943	0.00000000000000000	0.00000000000000000

On affiche enfin les erreurs absolues $|\widehat{x} - x_{\texttt{maxITER}}|$ en fonction du nombre d'itérations pour chaque méthode avec une échelle logarithmique pour l'axe des ordonnées.

```
from matplotlib.pylab import *
```

```
92 xlabel('Iterations')
93 ylabel('Absolute error')
94 axis([0.,nITER,0.,0.1])
95 semilogy(XXX,Dic,"r-o",XXX,Lag,"g-o",XXX,New,"y-o")
96 legend(['Dichotomie','Lagrange','Newton'])
97 show()
```

Le résultat est le suivant



On remarque tout d'abord que la décroissance de l'erreur avec la méthode de la dichotomie n'est pas monotone. De plus, on voit que la méthode de NEWTON est d'ordre 2 tandis que la méthode de LAGRANGE est d'ordre 1.

2. Interpolation

Étant donné $n+1$ points $\{(x_i, y_i)\}_{i=0}^{n}$, trouver une fonction $f : x \mapsto f(x)$ telle que $f(x_i) = y_i$

Approcher une fonction f consiste à la remplacer par une autre fonction φ dont la forme est plus simple et dont on peut se servir à la place de f. On verra dans le prochain chapitre qu'on utilise fréquemment cette stratégie en intégration numérique quand, au lieu de calculer $\int_a^b f(x)\,\mathrm{d}x$ on calcule de manière exacte $\int_a^b \varphi(x)\,\mathrm{d}x$, où φ est une fonction simple à intégrer (par exemple polynomiale). Dans d'autres contextes, la fonction f peut n'être connue que par les valeurs qu'elle prend en quelques points particuliers. Dans ce cas, on cherche à construire une fonction continue φ représentant une loi empirique qui se cacherait derrière les données.

2.1. Interpolation polynomiale

Étant donné $n+1$ couples $\{(x_i, y_i)\}_{i=0}^{n}$, le problème consiste à trouver une fonction $\varphi = \varphi(x)$ telle que $\varphi(x_i) = y_i$; on dit alors que φ interpole l'ensemble de valeurs $\{y_i\}_{i=0}^{n}$ aux nœuds $\{x_i\}_{i=0}^{n}$. Les quantités y_i représentent les valeurs aux nœuds x_i d'une fonction f connue analytiquement ou des données expérimentales. Dans le premier cas, l'approximation a pour but de remplacer f par une fonction plus simple en vue d'un calcul numérique d'intégrale ou de dérivée. Dans l'autre cas, le but est d'avoir une représentation synthétique de données expérimentales (dont le nombre peut être très élevé). On parle d'*interpolation polynomiale* quand φ est un polynôme et d'interpolation polynomiale par morceaux (ou d'interpolation par fonctions *splines*) si φ est polynomiale par morceaux.

Notons $\mathbb{R}_m[x]$ l'espace vectoriel formé par tous les polynômes de degré inférieur ou égale à m. Il est bien connu que $\mathbb{R}_m[x]$ a dimension $m+1$ et que sa base canonique est donnée par $\{1, x, x^2, \ldots, x^m\}$.

Supposons que l'on veuille chercher un polynôme P_m de degré $m \geq 0$ qui, pour des valeurs $x_0, x_1, x_2, \ldots, x_m$ distinctes données (appelés nœuds d'interpolation), prenne les valeurs $y_0, y_1, y_2, \ldots, y_m$ respectivement, c'est-à-dire

$$P_m(x_i) = y_i \qquad \text{pour } 0 \leq i \leq m. \tag{2.1}$$

Si un tel polynôme existe, il est appelé *polynôme d'interpolation* ou *polynôme interpolant.*

Théorème *Interpolation polynomiale*

Étant donné $m+1$ points distincts $x_0, \ldots, x_m$ et $m+1$ valeurs correspondantes $y_0, \ldots, y_m$, il existe un unique polynôme $P_m \in \mathbb{R}_m[x]$ tel que $P_m(x_i) = y_i$, pour $i = 0, \ldots m$.

2.1.1. Méthode directe (ou "naïve")

Une manière apparemment simple de résoudre ce problème est d'écrire le polynôme dans la base canonique de $\mathbb{R}_m[x]$:

$$P_m(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_m x^m,$$

où $a_0, a_1, a_2, \ldots, a_m$ sont des coefficients qui devront être déterminés. Les $(m+1)$ relations (2.1) s'écrivent alors

$$\begin{cases} a_0 + a_1 x_0 + \ldots a_n x_0^m = y_0 \\ a_0 + a_1 x_1 + \ldots a_n x_1^m = y_1 \\ \ldots \\ a_n + a_1 x_m + \ldots a_m x_m^m = y_m \end{cases}$$

Puisque les valeurs x_i et y_i sont connues, ces relations forment un système linéaire de $(m+1)$ équations en les $(m+1)$ inconnues $a_0, a_1, a_2, \dots, a_m$ qu'on peut mettre sous la forme matricielle [1]

$$\begin{pmatrix} 1 & x_0 & \dots & x_0^m \\ 1 & x_1 & \dots & x_1^m \\ \vdots & \vdots & & \vdots \\ 1 & x_m & \dots & x_m^m \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_m \end{pmatrix}. \tag{2.2}$$

Ainsi, le problème consistant à chercher le polynôme P_m satisfaisant (2.1) peut se réduire à résoudre le système linéaire (2.2). Cependant, résoudre une système linéaire de $(m+1)$ équations à $(m+1)$ inconnues n'est pas une tache triviale. Cette méthode pour trouver le polynôme P_m n'est donc pas une bonne méthode en pratique. Dans la suite on va étudier une méthode plus astucieuse pour construire le polynôme P_m.

2.1.2. Méthode de Lagrange

Quand on écrit le polynôme P_m dans la base canonique de $\mathbb{R}_m[x]$, le problème est de déterminer les $(m+1)$ coefficients $a_0, a_1, a_2, \dots, a_m$ tels que

$$P_m(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_m x^m.$$

On se demande s'il existe une autre base $\{L_0, L_1, L_2, \dots, L_m\}$ de $\mathbb{R}_m[x]$ telle que le polynôme P_m s'écrit

$$P_m(x) = y_0 L_0(x) + y_1 L_1(x) + y_2 L_2(x) + \cdots + y_m L_m(x),$$

autrement dit s'il existe une base telle que les coordonnées du polynôme dans cette base ne sont rien d'autre que les valeurs connues $y_0, y_1, \dots, y_m$.

Pour trouver une telle base, commençons par imposer le passage du polynômes par les $m+1$ points donnés : les $(m+1)$ relations (2.1) imposent la condition :

$$L_i(x_j) = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{sinon} \end{cases} \qquad \text{pour } 0 \le i, j \le m,$$

ce qui donne

$$L_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^{m} \frac{x - x_j}{x_i - x_j} = \frac{(x-x_0)(x-x_1)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_m)}{(x_i-x_0)(x_i-x_1)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_m)}.$$

Clairement, le numérateur de $L_i(x)$ est un produit de m termes $(x - x_j)$ avec $i \neq j$ et est donc un polynôme de degré m. Le dénominateur est une constante et il est facile de vérifier que

- $L_i(x) \in \mathbb{R}_m[x]$,
- $L_i(x_j) = 0$ si $i \neq j$, $0 \le i \le m$,
- $L_i(x_i) = 1$.

De plus, les polynômes $L_0, L_1, L_2, \dots, L_m$ sont linéairement indépendants car si l'équation $\sum_{i=0}^m \alpha_i L_i(x) = 0$ doit être satisfaite pour tout $x \in \mathbb{R}$ alors $\sum_{i=0}^m \alpha_i L_i(x_j) = 0$ doit être vraie pour tout $j = 0, 1, \dots, m$ et puisque $\sum_{i=0}^m \alpha_i L_i(x_j) = \alpha_j$, on conclut que tous les α_j sont nuls. Par conséquent, la famille $\{L_0, L_1, L_2, \dots, L_m\}$ forme une base de $\mathbb{R}_m[x]$.

Il est important de remarquer que nous avons construit explicitement une solution du problème (2.1) et ceci pour n'importe quelles valeurs $y_0, y_1, y_2, \dots, y_m$ données. Ceci montre que le système linéaire (2.2) a toujours une unique solution.

Théorème *Interpolation de* LAGRANGE

Étant donné $m+1$ points distincts $x_0, \dots, x_m$ et $m+1$ valeurs correspondantes $y_0, \dots, y_m$, il existe un unique polynôme $P_m \in \mathbb{R}_m[x]$ tel que $P_m(x_i) = y_i$, pour $i = 0, \dots m$ qu'on peut écrire sous la forme

$$P_m(x) = \sum_{i=0}^{m} y_i L_i(x) \ \in \mathbb{R}_m[x] \qquad \text{où} \quad L_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^{m} \frac{x - x_j}{x_i - x_j}.$$

Cette relation est appelée formule d'interpolation de LAGRANGE et les polynômes L_i sont les polynômes caractéristiques (de LAGRANGE).

1. La matrice $\begin{pmatrix} 1 & x_0 & \dots & x_0^m \\ 1 & x_1 & \dots & x_1^m \\ \vdots & \vdots & & \vdots \\ 1 & x_m & \dots & x_m^m \end{pmatrix}$ s'appelle matrice de VANDERMONDE.

Exemple

Pour $m = 2$ le polynôme de LAGRANGE s'écrit

$$P(x) = y_0 \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + y_1 \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + y_2 \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}$$

Exemple

On cherche le polynôme d'interpolation de LAGRANGE qui en -1 vaut 8, en 0 vaut 3 et en 1 vaut 6. On a

$$\begin{aligned} P(x) &= y_0 \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + y_1 \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + y_2 \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \\ &= 8\frac{x(x-1)}{2} + 3\frac{(x+1)(x-1)}{-1} + 6\frac{(x+1)x}{2} = 4x^2 - x + 3. \end{aligned}$$

Remarque

Si m est petit il est souvent plus simple de calculer directement les coefficients a_0, a_1, …, a_m avec la méthode "naïve" en résolvant le système linéaire (2.2).

Soit $f : \mathbb{R} \to \mathbb{R}$ une fonction continue donnée et soit $x_0, x_1, x_2, \ldots, x_m$, $(m+1)$ points distincts donnés. Interpoler la fonction f aux points x_i, $0 \le i \le m$ signifie chercher un polynôme P_m de degré m tel que

$$P_m(x_i) = f(x_i) \qquad \text{pour } 0 \le i \le m. \tag{2.3}$$

La solution de ce problème est donc donnée par

$$P_m(x) = \sum_{i=0}^{m} f(x_i) L_i(x) \ \in \mathbb{R}_m[x] \qquad \text{où} \quad L_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^{m} \frac{x - x_j}{x_i - x_j}$$

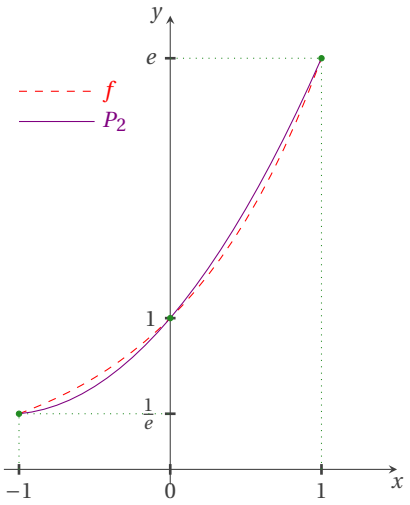
et le polynôme P_m est appelée *interpolant de f de degré m aux points $x_0, x_1, x_2, \ldots, x_m$*.

Exemple

Soit $f : \mathbb{R} \to \mathbb{R}$ la fonction définie par $f(x) = e^x$. On cherche l'interpolant de f aux points -1, 0, 1. On a

$$\begin{aligned} P(x) &= f(x_0) \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + f(x_1) \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + f(x_2) \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \\ &= \frac{1}{e}\frac{x(x-1)}{2} + \frac{(x+1)(x-1)}{-1} + e\frac{(x+1)x}{2} = \left(\frac{1}{2e} - 1 - \frac{e}{2}\right)x^2 + \left(\frac{e}{2} - \frac{1}{2e}\right)x + 1. \end{aligned}$$

La figure ci-dessous montre le graphe de la fonction f et de son interpolant aux points -1, 0, 1.



Proposition *Erreur*

Si $y_i = f(x_i)$ pour $i = 0, 1, \ldots, n$, $f : I \to \mathbb{R}$ étant une fonction donnée de classe $\mathscr{C}^{n+1}(I)$ où I est le plus petit intervalle

contenant les nœuds distincts $\{x_i\}_{i=0}^n$, alors il existe $\xi \in I$ tel que l'erreur d'interpolation au point $x \in I$ est donnée par

$$E_n(x) \equiv f(x) - P_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x)$$

où $\omega_{n+1}(x) \equiv \prod_{i=0}^{m}(x - x_j)$.

Démonstration. Le résultat est évidemment vrai si x coïncide avec l'un des nœuds d'interpolation car $E_n(x_i) = 0$ pour $i = 0, 1, \ldots, n$. Autrement, soit $x \in I$ fixé, $x \neq x_i$ pour $i = 0, \ldots, n$ et définissons la fonction

$$\begin{aligned} G\colon I &\to \mathbb{R} \\ t &\mapsto E_n(t) - E_n(x)\frac{\omega_{n+1}(t)}{\omega_{n+1}(x)} \end{aligned}$$

Puisque $f \in \mathscr{C}^{(n+1)}(I)$ et puisque ω_{n+1} est un polynôme, $G \in \mathscr{C}^{(n+1)}(I)$ et possède au moins $n+2$ zéros distincts dans I. En effet, les zéros de G sont les $n+1$ nœuds x_i et le point x car

$$\begin{aligned} G(x_i) &= E_n(x_i) - E_n(x)\frac{\omega_{n+1}(x_i)}{\omega_{n+1}(x)} = 0, \qquad i = 0, \ldots, n \\ G(x) &= E_n(x) - E_n(x)\frac{\omega_{n+1}(x)}{\omega_{n+1}(x)} = 0. \end{aligned}$$

Ainsi, d'après le théorème des valeurs intermédiaires, G' admet au moins $n+1$ zéros distincts et par récurrence $G^{(j)}$ a au moins $n+2-j$ zéros distincts. Par conséquent, $G^{(n+1)}$ a au moins un zéro, qu'on note ξ. D'autre part, puisque $E_n^{(n+1)}(t) = f^{(n+1)}(t)$ et $\omega_{n+1}^{(n+1)}(x) = (n+1)!$ on a

$$G^{(n+1)}(t) = f^{(n+1)}(t) - E_n(x)\frac{(n+1)!}{\omega_{n+1}(x)}$$

ce qui donne, avec $t = \xi$, l'expression voulue pour $E_n(x)$. □

Dans le cas d'une distribution uniforme de nœuds, *i.e.* quand $x_i = x_{i-1} + h$ avec $i = 1, 2, \ldots, n$ et $h > 0$ et x_0 donnés, on a

$$|\omega_{n+1}(x)| \leq n!\frac{h^{n+1}}{4}$$

et donc

$$\max_{x\in I}|E_n(x)| \leq \frac{\max_{x\in I}|f^{(n+1)}(x)|}{4(n+1)}h^{n+1}.$$

Attention *Les défauts de l'interpolation polynomiale avec nœuds équirépartis*

Malheureusement, on ne peut pas déduire de cette relation que l'erreur tend vers 0 quand n tend vers l'infini, bien que $h^{n+1}/[4(n+1)]$ tend effectivement vers 0. En fait, il existe des fonctions f pour lesquelles $\max_{x\in I}|E_n(x)| \xrightarrow[n\to+\infty]{} +\infty$. Ce résultat frappant indique qu'en augmentant le degré n du polynôme d'interpolation, on n'obtient pas nécessairement une meilleure reconstruction de f.

Exemple *Le contre-exemple de* RUNGE

Ce phénomène est bien illustré par la fonction de RUNGE : soit la fonction $f\colon [-5,5] \to \mathbb{R}$ définie par $f(x) = \frac{1}{1+x^2}$. La fonction f est infiniment dérivable sur $[-5,5]$ et $|f^{(n)}(\pm 5)|$ devient très rapidement grand lorsque n tend vers l'infini. Si on considère une distribution uniforme des nœuds on voit que l'erreur tend vers l'infini quand n tend vers l'infini. Ceci est lié au fait que la quantité $\max_{x\in[-5,5]}|f^{(n+1)}(x)|$ tend plus vite vers l'infini que $\frac{h^{n+1}}{4(n+1)}$ tend vers zéro. La figure 2.1a montre ses polynômes interpolants de degrés 3, 5 et 10 pour une distribution équirepartie des nœuds. Cette absence de convergence est également mise en évidence par les fortes oscillations observées sur le graphe du polynôme d'interpolation (absentes sur le graphe de f), particulièrement au voisinage des extrémités de l'intervalle. Ce comportement est connu sous le nom de *phénomène de* RUNGE. On peut éviter le phénomène de RUNGE en choisissant correctement la distribution des nœuds d'interpolation. Sur un intervalle $[a,b]$, on peut par exemple considérer les nœuds de CHEBYSHEV-GAUSS-LOBATTO (voir figure 2.1b)

$$x_i = \frac{a+b}{2} - \frac{b-a}{2}\cos\left(\frac{\pi}{n}i\right), \qquad \text{pour } i = 0, \ldots, n$$

Pour cette distribution particulière de nœuds, il est possible de montrer que, si f est dérivable sur $[a,b]$, alors P_n converge vers f quand $n \to +\infty$ pour tout $x \in [a,b]$. Les nœuds de CHEBYSHEV-GAUSS-LOBATTO, qui sont les abscisses des nœuds équirépartis sur le demi-cercle unité, se trouvent à l'intérieur de $[a,b]$ et sont regroupés près des extrémités de l'intervalle. Ces figures ont été obtenue

par les instructions suivantes :

```
from matplotlib.pylab import *

def lagrange(t,x,y):
    p = 0
    n = len(x)
    L = [1 for i in range(n)]
    for i in range(n):
        for j in range(n):
            if j!=i:
                L[i] *= (t-x[j])/(x[i]-x[j])
        p += y[i]*L[i]
    return p

def f(x):
    return 1./(1.+x**2)
```

"Noeuds équirépartis"

```
x1 = linspace(-5,5,3)
x2 = linspace(-5,5,5)
x3 = linspace(-5,5,10)
y1 = f(x1)
y2 = f(x2)
y3 = f(x3)

# Calcul des polynomes en plusieurs points d'un intervalle pour affichage
t = arange(-5,5,.1)
l1t = []
l2t = []
l3t = []
for k in t:
    l1t.append(lagrange(k,x1,y1))
    l2t.append(lagrange(k,x2,y2))
    l3t.append(lagrange(k,x3,y3))

plot(t,f(t),'r-',t,l1t,'b:',t,l2t,'m-.',t,l3t,'y--')
legend(['f','p_3','p_5','p_10'],loc='lower center')
axis([-5, 5, -0.5, 1])
show()
```

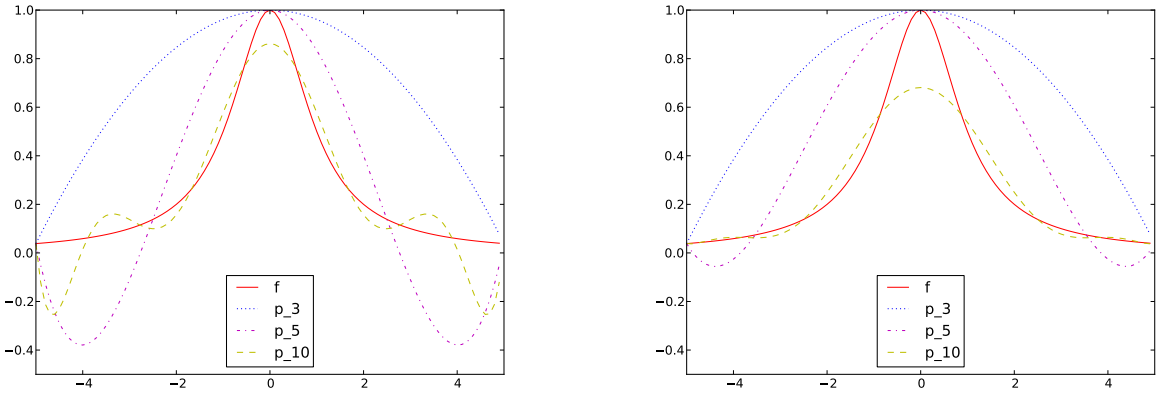
"Noeuds de CHEBYSHEV-GAUSS-LOBATTO"

```
def Tchebychev(a,b,n):
    return [0.5*(a+b)-0.5*(b-a)*cos(pi*i/(n-1)) for i in range(n)]

x1 = Tchebychev(-5,5,3)
x2 = Tchebychev(-5,5,5)
x3 = Tchebychev(-5,5,10)
y1 = [f(x) for x in x1]
y2 = [f(x) for x in x2]
y3 = [f(x) for x in x3]

# Calcul des polynomes en plusieurs points d'un intervalle pour affichage
t = arange(-5,5,.1)
l1t = []
l2t = []
l3t = []
for k in t:
    l1t.append(lagrange(k,x1,y1))
    l2t.append(lagrange(k,x2,y2))
    l3t.append(lagrange(k,x3,y3))

plot(t,f(t),'r-',t,l1t,'b:',t,l2t,'m-.',t,l3t,'y--')
legend(['f','p_3','p_5','p_10'],loc='lower center')
```



(a) Distribution équirepartie des nœuds

(b) Nœuds de CHEBYSHEV-GAUSS-LOBATTO

FIGURE 2.1.: Interpolation de LAGRANGE, exemple de RUNGE

```
axis([-5, 5, -0.5, 1])
show()
```

2.1.3. Stabilité de l'interpolation polynomiale

Soit $f\colon I \to \mathbb{R}$ une fonction de classe $\mathscr{C}^{n+1}(I)$ où I est le plus petit intervalle contenant les nœuds distincts $\{x_i\}_{i=0}^n$. Qu'arrive-t-il aux polynômes d'interpolation si, au lieu des valeurs exactes $f(x_i)$, on considère des valeurs perturbées $\tilde{f}(x_i)$, $i = 0, \dots, n$? Ces perturbations peuvent provenir d'erreurs d'arrondi ou d'incertitudes dans les mesures. Soit P_n le polynôme exact interpolant les valeurs $f(x_i)$ et $\tilde{P}_n$ le polynôme exact interpolant les valeurs $\tilde{f}(x_i)$. En notant $\mathbf{x}$ le vecteur dont les composantes sont les nœuds d'interpolation, on a

$$\max_{x\in I}|P_n(x) - \tilde{P}_n(x)| = \max_{x\in I}|\sum_{i=0}^{n}(f(x_i) - \tilde{f}(x_i))\varphi_i(x)| \leq \Lambda_n(\mathbf{x}) \max_{0\leq i\leq n}|f(x) - \tilde{f}(x)|$$

où

$$\Lambda_n(\mathbf{x}) \equiv \max_{x\in I}\sum_{i=0}^{n}|\varphi_i(x)|$$

est appelée constante de LEBESGUE (noter que cette constante dépend des nœuds d'interpolation). Des petites perturbations sur les valeurs nodales $f(x_i)$ entraînent des petites variations sur le polynôme d'interpolation quand la constante de LEBESGUE est petite. La constante de LEBESGUE mesure donc le *conditionnement* du problème d'interpolation. Pour l'interpolation de LAGRANGE avec des nœuds équirépartis

$$\Lambda_n(\mathbf{x}) \simeq \frac{2^{n+1}}{(\ln(n)+\gamma)ne}$$

où $e \simeq 2.71834$ (nombre de NEPER) et $\gamma \simeq 0.547721$ (constante d'EULER). Quand n est grand, l'interpolation de LAGRANGE sur des nœuds équirépartis peut donc être instable.

Exemple

Dans la Figure 2.2 on a tracé

- la fonction $f(x) = \sin(2\pi x)$,
- le polynôme de LAGRANGE ℓ_{21} qui interpole f en 22 nœuds équirépartis sur l'intervalle $[-1;1]$, c'est-à-dire l'ensemble $\{x_i = -1 + 0.1i, y_i = f(x_i)\}_{i=0}^{21}$,
- le polynôme de LAGRANGE p_{21} qui interpole l'ensemble perturbé $\{(x_i, \tilde{y}_i)\}_{i=0}^{21}$ où $\tilde{y}_i$ est une perturbation aléatoire des valeurs exactes y_i de sorte que

$$\max_{i=0,\dots,21}|y_i - \tilde{y}_i| \leq 10^{-3}.$$

On remarque que la différence entre ces deux polynômes est bien plus grande que la perturbation des données. Plus précisément

$$\max_{x\in I}|P_n(x) - \tilde{P}_n(x)| \simeq 6.212$$

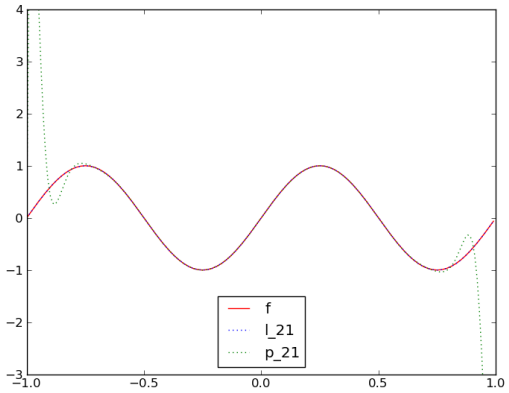


FIGURE 2.2.: Effet de perturbations sur l'interpolation de LAGRANGE en des nœuds équirépartis.

et l'écart est particulièrement important aux extrémités de l'intervalle. Remarquer que dans cet exemple la constante de LEBESGUE est très grande : $\Lambda_n(\mathbf{x}) \simeq 19274$.
Cette figure a été obtenue par les instructions :

```
from matplotlib.pylab import *
import random

def lagrange(t,x,y):
    p = 0
    n = len(x)
    L = [1 for i in range(n)]
    for i in range(n):
        for j in range(n):
            if j!=i:
                L[i] *= (t-x[j])/(x[i]-x[j])
        p += y[i]*L[i]
    return p

def f(x):
    return sin(2*math.pi*x)

x1 = linspace(-1,1,22)
y1 = f(x1)
y2 = [yi+(2.*random.random()-1.)*0.001 for yi in y1]

# Calcul des polynomes en plusieurs points d'un intervalle pour affichage
t = arange(-1,1,.01)
l1t = []
l2t = []
for k in t:
    l1t.append(lagrange(k,x1,y1))
    l2t.append(lagrange(k,x1,y2))

print max(abs(y1-y2))
print max([abs(l1t[i]-l2t[i]) for i in range(len(t))])

plot(t,f(t),'r-',t,l1t,'b:',t,l2t,'g:')
legend(['f','l_21','p_21'],loc='lower center')
axis([-1, 1, -3, 4])
show()
```

2.1.4. Méthode de Newton

On a vu que calculer le polynôme d'interpolation de LAGRANGE dans la base canonique de $\mathbb{R}_n[x]$ comporte la résolution d'un système linéaire d'ordre n. On a alors introduit une autre base de $\mathbb{R}_n[x]$, la base des polynômes de LAGRANGE, qui

permet de calculer directement le polynôme d'interpolation car les coordonnées du polynôme cherché dans cette base ne sont rien d'autres que les valeurs y_i. Cependant, cette méthode n'est pas la plus efficace d'un point de vue pratique. En effet, pour calculer le polynôme d'interpolation d'un ensemble de $n+1$ points on doit calculer les $n+1$ polynômes $\{L_0, L_1, L_2, \ldots, L_n\}$. Si ensuite on ajoute un point d'interpolation, on doit calculer les $n+2$ polynômes $\{\tilde{L}_0, \tilde{L}_1, \tilde{L}_2, \ldots, \tilde{L}_{n+1}\}$ qui diffèrent tous des $n+1$ calculés précédemment. La méthode de NEWTON est basée sur le choix d'une autre base de sort à ce que l'ajout d'un point comporte juste l'ajout d'une fonction de base.

Considérons la famille de polynômes $\{\omega_0, \omega_1, \omega_2, \ldots, \omega_n\}$ où

$$\omega_0(x) = 1,$$
$$\omega_k(x) = \prod_{i=0}^{k-1}(x - x_i) = (x - x_{k-1})\omega_{k-1}(x), \qquad \forall k = 1, \ldots, n.$$

Il est facile de vérifier que

- $\omega_k(x) \in \mathbb{R}_n[x]$,
- la famille $\{\omega_0, \omega_1, \omega_2, \ldots, \omega_n\}$ est génératrice de $\mathbb{R}_n[x]$
- la famille $\{\omega_0, \omega_1, \omega_2, \ldots, \omega_n\}$ est libre.

Par conséquent, la famille $\{\omega_0, \omega_1, \omega_2, \ldots, \omega_n\}$ forme une base de $\mathbb{R}_n[x]$.

Si on choisit comme base de $\mathbb{R}_n[x]$ la famille $\{\omega_0, \omega_1, \omega_2, \ldots, \omega_n\}$, le problème du calcul du polynôme d'interpolation p_n est alors ramené au calcul des coefficients $\{\alpha_0, \alpha_1, \alpha_2, \ldots, \alpha_n\}$ tels que

$$p_n(x) = \sum_{i=0}^{n} \alpha_i \omega_i(x).$$

Si on a calculé les $n+1$ coefficients $\{\alpha_0, \alpha_1, \alpha_2, \ldots, \alpha_n\}$ et on ajoute un point d'interpolation, il n'y a plus a calculer que le coefficient α_{n+1} car la nouvelle base est déduite de l'autre base en ajoutant simplement le polynôme ω_{n+1}.

Commençons par chercher une formule qui permet de calculer ces coefficients. Le polynôme d'interpolation dans la base de NEWTON évalué en x_0 donne

$$p_n(x_0) = \sum_{i=0}^{n} \alpha_i \omega_i(x_0) = \alpha_0$$

donc $\alpha_0 = y_0$. Le polynôme d'interpolation dans la base de NEWTON évalué en x_1 donne

$$p_n(x_1) = \sum_{i=0}^{n} \alpha_i \omega_i(x_1) = \alpha_0 + \alpha_1(x_1 - x_0)$$

donc $\alpha_1 = \frac{y_1 - y_0}{x_1 - x_0}$. Le polynôme d'interpolation dans la base de NEWTON évalué en x_2 donne

$$p_n(x_2) = \sum_{i=0}^{n} \alpha_i \omega_i(x_2) = \alpha_0 + \alpha_1(x_2 - x_0) + \alpha_2(x_2 - x_0)(x_2 - x_1)$$

donc

$$\alpha_2 = \frac{y_2 - \alpha_0 - \alpha_1(x_2 - x_0)}{(x_2 - x_0)(x_1 - x_0)} = \frac{y_2 - y_0 - \frac{y_1 - y_0}{x_1 - x_0}(x_2 - x_0)}{(x_2 - x_0)(x_1 - x_0)} = \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}$$

Pour calculer tous les coefficients on va alors introduire la notion de *différence divisée* :

Définition *Différences divisées*

Soit $\{(x_i, y_i)\}_{i=0}^{n}$ un ensemble de $n+1$ points distincts.

- La différence divisée d'ordre 1 de x_{i-1} et x_i est

$$f[x_{i-1}, x_i] \equiv \frac{y_i - y_{i-1}}{x_i - x_{i-1}}.$$

- La différence divisée d'ordre n des $n+1$ points $x_0, \ldots, x_n$ est définie par récurrence en utilisant deux différences divisées d'ordre $n-1$ comme suit :

$$f[x_0, \ldots, x_n] \equiv \frac{f[x_1, \ldots, x_n] - f[x_0, \ldots, x_{n-1}]}{x_n - x_0}$$

Pour expliciter le processus récursif, les différences divisées peuvent être calculées en les disposant de la manière suivante dans un tableau :

i	x_i	y_i	$f[x_{i-1},x_i]$	$f[x_{i-2},x_{i-1},x_i]$	$f[x_{i-3},x_{i-2},x_{i-1},x_i]$	$f[x_{i-4},x_{i-3},x_{i-2},x_{i-1},x_i]$	$\dots$
0	x_0	$\boxed{y_0}$					
1	x_1	y_1	$\boxed{f[x_0,x_1]}$				
2	x_2	y_2	$f[x_1,x_2]$	$\boxed{f[x_0,x_1,x_2]}$			
3	x_3	y_3	$f[x_2,x_3]$	$f[x_1,x_2,x_3]$	$\boxed{f[x_0,x_1,x_2,x_3]}$		
4	x_4	y_4	$f[x_3,x_4]$	$f[x_2,x_3,x_4]$	$f[x_1,x_2,x_3,x_4]$	$\boxed{f[x_0,x_1,x_2,x_3,x_4]}$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\boxed{\ddots}$

Théorème *Formule de* NEWTON

Soit $\{(x_i,y_i)\}_{i=0}^n$ un ensemble de $n+1$ points distincts. Le polynôme d'interpolation de LAGRANGE p_n sous la forme de NEWTON est donné par

$$p_n(x)=\sum_{i=0}^{n}\omega_i(x)f[x_0,\dots,x_i].$$

Comme le montre la définition des différences divisées, des points supplémentaires peuvent être ajoutés pour créer un nouveau polynôme d'interpolation sans recalculer les coefficients. De plus, si un point est modifié, il est inutile de recalculer l'ensemble des coefficients. Autre avantage, si les x_i sont équirépartis, le calcul des différences divisées devient nettement plus rapide. Par conséquent, l'interpolation polynomiale dans une base de NEWTON est privilégiée par rapport à une interpolation dans la base de LAGRANGE pour des raisons pratiques.

Exemple

On veut calculer le polynôme d'interpolation de de la fonction $f(x)=\sin(x)$ en les 3 points $x_i=\frac{\pi}{2}i$ avec $i=0,\dots,2$. On cherche donc $p_2\in\mathbb{R}_2[x]$ tel que $p_2(x_i)=\sin(x_i)$ pour $i=0,\dots,2$.

Méthode directe. Si on écrit $p_2(x)=\alpha_0+\alpha_1x+\alpha_2x^2$, on cherche $\alpha_0,\alpha_1,\alpha_2$ tels que

$$\begin{pmatrix}1&0&0\\1&\frac{\pi}{2}&\frac{\pi^2}{4}\\1&\pi&\pi^2\end{pmatrix}\begin{pmatrix}\alpha_0\\\alpha_1\\\alpha_2\end{pmatrix}=\begin{pmatrix}0\\1\\0\end{pmatrix}$$

En résolvant ce système linéaire on trouve $\alpha_0=0$, $\alpha_1=\frac{4}{\pi}$ et $\alpha_2=-\frac{4}{\pi^2}$.

Méthode de Lagrange. On a

$$p_2(x)=y_0L_0(x)+y_1L_1(x)+y_2L_2(x)=\frac{x(x-\pi)}{\frac{\pi}{2}\left(\frac{\pi}{2}-\pi\right)}=-\frac{4}{\pi^2}x(x-\pi).$$

Méthode de Newton. On commence par construire le tableau des différences divisées :

i	x_i	y_i	$f[x_{i-1},x_i]$	$f[x_{i-2},x_{i-1},x_i]$
0	0	$\boxed{0}$		
1	$\frac{\pi}{2}$	1	$\boxed{\frac{2}{\pi}}$	
2	π	0	$-\frac{2}{\pi}$	$\boxed{-\frac{4}{\pi^2}}$

On a alors

$$\begin{aligned}p_2(x)&=\sum_{i=0}^{2}\omega_i(x)f[x_0,\dots,x_i]\\&=\omega_0(x)f[x_0]+\omega_1(x)f[x_0,x_1]+\omega_2(x)f[x_0,x_1,x_2]\\&=\frac{2}{\pi}\omega_1(x)-\frac{4}{\pi^2}\omega_2(x)\\&=\frac{2}{\pi}x-\frac{4}{\pi^2}x\left(x-\frac{\pi}{2}\right)\\&=-\frac{4}{\pi^2}x(x-\pi).\end{aligned}$$

Maintenant on veut calculer le polynôme d'interpolation de la même fonction en les 4 points $x_i=\frac{\pi}{2}i$ avec $i=0,\dots,3$, *i.e.* on a juste ajouté le point $x=3\pi/2$. On cherche donc $p_3\in\mathbb{R}_3[x]$ tel que $p_3(x_i)=\sin(x_i)$ pour $i=0,\dots,3$.

Méthode directe. Si on écrit $p_3(x) = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3$, on cherche $\alpha_0, \alpha_1, \alpha_2, \alpha_3$ tels que

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & \frac{\pi}{2} & \frac{\pi^2}{4} & \frac{\pi^3}{8} \\ 1 & \pi & \pi^2 & \pi^3 \\ 1 & \frac{3\pi}{2} & \frac{9\pi^2}{4} & \frac{27\pi^3}{8} \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ -1 \end{pmatrix}$$

En résolvant ce système linéaire on trouve $\alpha_0 = 0$, $\alpha_1 = \frac{16}{3\pi}$, $\alpha_2 = -\frac{8}{\pi^2}$ et $\alpha_3 = \frac{8}{3\pi^3}$.

Méthode de Lagrange. On a

$$p_3(x) = y_0 L_0(x) + y_1 L_1(x) + y_2 L_2(x) + y_3 L_3(x) = \frac{x(x-\pi)\left(x-\frac{3\pi}{2}\right)}{\frac{\pi}{2}\left(\frac{\pi}{2}-\pi\right)\left(\frac{\pi}{2}-\frac{3\pi}{2}\right)} - \frac{x\left(x-\frac{\pi}{2}\right)(x-\pi)}{\frac{3\pi}{2}\left(\frac{3\pi}{2}-\frac{\pi}{2}\right)\left(\frac{3\pi}{2}-\pi\right)}$$
$$= \frac{4}{\pi^3} x(x-\pi)\left(x-\frac{3\pi}{2}\right) - \frac{4}{3\pi^3} x\left(x-\frac{\pi}{2}\right)(x-\pi).$$

Méthode de Newton. Il suffit de calculer une différence divisée en plus, *i.e.* ajouter une ligne au tableau :

i	x_i	y_i	$f[x_{i-1}, x_i]$	$f[x_{i-2}, x_{i-1}, x_i]$	$f[x_{i-3}, x_{i-2}, x_{i-1}, x_i]$
0	0	$\boxed{0}$			
1	$\frac{\pi}{2}$	1	$\boxed{\frac{2}{\pi}}$		
2	π	0	$-\frac{2}{\pi}$	$\boxed{-\frac{4}{\pi^2}}$	
3	$\frac{3\pi}{2}$	-1	$-\frac{2}{\pi}$	0	$\boxed{\frac{8}{3\pi^3}}$

On a alors

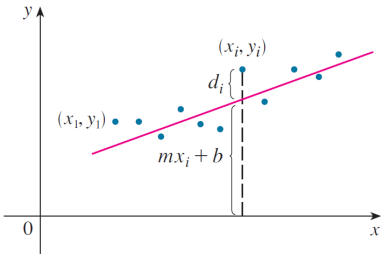
$$\begin{aligned} p_3(x) &= \sum_{i=0}^{3} \omega_i(x) f[x_0, \dots, x_i] \\ &= p_2(x) + \omega_3(x) f[x_0, x_1, x_2, x_3] \\ &= -\frac{4}{\pi^2} x(x-\pi) + \frac{8}{3\pi^3} \omega_3(x) \\ &= -\frac{4}{\pi^2} x(x-\pi) + \frac{8}{3\pi^3} x\left(x-\frac{\pi}{2}\right)(x-\pi) \\ &= \frac{8}{3\pi^3} x(x^2 - 3\pi x + 2\pi^2). \end{aligned}$$

Méthode des moindres carrés : fitting par une relation affine

Lorsqu'un chercheur met au point une expérience (parce qu'il a quelques raisons de croire que les deux grandeurs x et y sont liées par une fonction f), il récolte des données sous la forme de points $\{(x_i, y_i)\}_{i=0}^{n}$. Lorsqu'il en fait une représentation graphique il cherche f pour qu'elle s'ajuste le mieux possible aux points observés.

Nous avons déjà vu que si n est grand, le polynôme d'interpolation de LAGRANGE n'est pas toujours une bonne approximation d'une fonction donnée/cherchée. De plus, si les données sont affectées par des erreurs de mesure, l'interpolation peut être instable. Ce problème peut être résolu avec l'interpolation composite (avec des fonctions linéaires par morceau ou des splines). Néanmoins, aucune de ces méthodes n'est adaptée à l'extrapolation d'informations à partir des données disponibles, c'est-à-dire, à la génération de nouvelles valeurs en des points situés à l'extérieur de l'intervalle contenant les nœuds d'interpolation. On introduit alors la méthode des moindres carrés : soit $d_i = y_i - f(x_i)$ l'écart vertical du point (x_i, y_i) par rapport à la fonction f. La méthode des moindres carrés est celle qui choisit f de sorte que la somme des carrés de ces déviations soit minimale.

Supposons que les deux grandeurs x et y sont liées approximativement par une relation affine, c'est-à-dire de la forme $y = mx + q$ pour certaines valeurs de m et q (autrement dit, lorsqu'on affiche ces points dans un plan cartésien, les points ne sont pas exactement alignés mais cela semble être dû à des erreurs de mesure). On souhaite alors trouver les constantes m et q pour que la droite d'équation $y = mx + q$ s'ajuste *le mieux possible* aux points observés. Pour cela, introduisons $d_i \equiv y_i - (mx_i + q)$ l'écart vertical du point (x_i, y_i) par rapport à la droite.



La méthode des moindres carrés est celle qui choisit m et q de sorte que *la somme des carrés de ces déviations soit minimale*. Pour cela, on doit minimiser la fonction $\mathscr{E} : \mathbb{R}^2 \to \mathbb{R}_+$ définie par

$$\mathscr{E}(m,q) = \sum_{i=0}^{n} d_i^2 = \sum_{i=0}^{n} (y_i - mx_i - q)^2.$$

Pour minimiser $\mathscr{E}$ on cherche d'abord les points stationnaires, *i.e.* les points (m,q) qui vérifient $\frac{\partial \mathscr{E}}{\partial m} = \frac{\partial \mathscr{E}}{\partial q} = 0$. Puisque

$$\frac{\partial \mathscr{E}}{\partial m}(m,q) = -2\left(\sum_{i=0}^{n} (y_i - (mx_i + q))x_i\right), \qquad \frac{\partial \mathscr{E}}{\partial q}(m,q) = -2\left(\sum_{i=0}^{n} (y_i - (mx_i + q))\right),$$

alors

$$\begin{cases} \frac{\partial \mathscr{E}}{\partial m}(m,q) = 0 \\ \frac{\partial \mathscr{E}}{\partial q}(m,q) = 0 \end{cases} \iff \begin{cases} \sum_{i=0}^{n} (y_i - mx_i - q)x_i = 0 \\ \sum_{i=0}^{n} (y_i - mx_i - q) = 0 \end{cases}$$

$$\iff \begin{cases} \left(\sum_{i=1}^{n} x_i^2\right) m + \left(\sum_{i=0}^{n} x_i\right) q = \sum_{i=0}^{n} y_i x_i \\ \left(\sum_{i=1}^{n} x_i\right) m + (n+1) q = \sum_{i=0}^{n} y_i \end{cases} \iff \begin{cases} m = \dfrac{\left(\sum_{i=0}^{n} x_i\right)\left(\sum_{i=0}^{n} y_i\right) - (n+1)\left(\sum_{i=0}^{n} x_i y_i\right)}{\left(\sum_{i=0}^{n} x_i\right)^2 - (n+1)\left(\sum_{i=0}^{n} x_i^2\right)}, \\ q = \dfrac{\left(\sum_{i=0}^{n} x_i\right)\left(\sum_{i=0}^{n} x_i y_i\right) - \left(\sum_{i=0}^{n} y_i\right)\left(\sum_{i=0}^{n} x_i^2\right)}{\left(\sum_{i=0}^{n} x_i\right)^2 - (n+1)\left(\sum_{i=0}^{n} x_i^2\right)}. \end{cases}$$

On a trouvé un seul point stationnaire. On établi sa nature en étudiant la matrice Hessienne :

$$H_{\mathscr{E}}(m,q) = 2\begin{pmatrix} \sum_{i=1}^{n} x_i^2 & \sum_{i=0}^{n} x_i \\ \sum_{i=0}^{n} x_i & (n+1) \end{pmatrix}$$

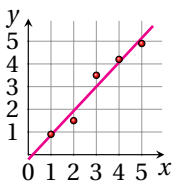
et $\det(H_{\mathscr{E}}(m,q)) = 4\left((n+1)\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=0}^{n} x_i\right)^2\right) > 0$ avec $\partial_{mm}\mathscr{E}(m,q) = \sum_{i=1}^{n} x_i^2 > 0$ donc il s'agit d'un minimum. La droite d'équation $y = mx + q$ ainsi calculée s'appelle *droite de régression de y par rapport à x*.

Exemple

Si on a le points suivantes

x	1	2	3	4	5
y	0.9	1.5	3.5	4.2	4.9

on trouve $m = 1.07$ et $q = -0.21$.



2.2. Polynôme d'Hermite ou polynôme osculateur

On peut généraliser l'interpolation de Lagrange pour prendre en compte, en plus des valeurs nodales, les valeurs de la dérivée du polynôme interpolateur dans ces nœuds.

Considérons $n+1$ triplets (x_i, y_i, y_i'), le problème est de trouver un polynôme $\Pi_m(x) = a_0 + a_1 x + \dots a_m x^m \in \mathbb{R}_m[x]$ tel quel

$$\begin{cases} \Pi_m(x_i) = y_i, \\ \Pi_m'(x_i) = y_i', \end{cases} \qquad i = 0, \dots n.$$

Il s'agit d'un système linéaire de $2(n+1)$ équations et $m+1$ inconnues. Si $m = 2n+1$ on a le résultat suivant :

Théorème

Étant donné $n+1$ points distincts $x_0, \dots, x_n$ et $n+1$ couples correspondantes $(y_0, y_0'), \dots, (y_n, y_n')$, il existe un unique

polynôme $\Pi_{2n+1} \in \mathbb{R}_{2n+1}[x]$ tel que $\Pi_{2n+1}(x_i) = y_i$ et $\Pi'_{2n+1}(x_i) = y'_i$, pour $i = 0, \dots n$ qu'on peut écrire sous la forme

$$Q(x) = \sum_{i=0}^{n} y_i A_i(x) + y'_i B_i(x) \ \in \mathbb{P}_{2n+1} \qquad \text{où} \quad \begin{cases} L_i(x) & = \prod\limits_{\substack{j=0 \\ j\neq i}}^{n} \frac{x-x_j}{x_i-x_j}, \\ c_i & = \sum\limits_{\substack{j=0 \\ j\neq i}}^{n} \frac{1}{x_i-x_j}, \\ A_i(x) & = (1-2(x-x_i)c_i)(L_i(x))^2, \\ B_i(x) & = (x-x_i)(L_i(x))^2, \end{cases}$$

ou encore sous la forme

$$Q(x) = \sum_{i=0}^{n} (y_i D_i(x) + y'_i(x-x_i))(L_i(x))^2 \qquad \text{où} \quad \begin{cases} L_i(x) & = \prod\limits_{\substack{j=0 \\ j\neq i}}^{n} \frac{x-x_j}{x_i-x_j}, \\ c_i & = \sum\limits_{\substack{j=0 \\ j\neq i}}^{n} \frac{1}{x_i-x_j}, \\ D_i(x) & = 1-2(x-x_i)c_i. \end{cases}$$

Cette relation est appelée formule d'interpolation de HERMITE.

Exemple

Pour $n = 2$ le polynôme d'HERMITE s'écrit

$$\begin{aligned} Q(x) = {} & y_0 \left(1-2(x-x_0)\left(\frac{1}{x_0-x_1}+\frac{1}{x_0-x_2}\right)\right)\left(\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}\right)^2 + y'_0(x-x_0)\left(\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}\right)^2 \\ & + y_1 \left(1-2(x-x_1)\left(\frac{1}{x_1-x_0}+\frac{1}{x_1-x_2}\right)\right)\left(\frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}\right)^2 + y'_1(x-x_1)\left(\frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}\right)^2 \\ & + y_2 \left(1-2(x-x_2)\left(\frac{1}{x_2-x_0}+\frac{1}{x_2-x_1}\right)\right)\left(\frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}\right)^2 + y'_2(x-x_2)\left(\frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}\right)^2, \end{aligned}$$

qu'on peut réécrire comme

$$\begin{aligned} Q(x) = {} & \left(y_0 \left(1-2(x-x_0)\left(\frac{1}{x_0-x_1}+\frac{1}{x_0-x_2}\right)\right) + y'_0(x-x_0)\right)\left(\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}\right)^2 \\ & + \left(y_1 \left(1-2(x-x_1)\left(\frac{1}{x_1-x_0}+\frac{1}{x_1-x_2}\right)\right) + y'_1(x-x_1)\right)\left(\frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}\right)^2 \\ & + \left(y_2 \left(1-2(x-x_2)\left(\frac{1}{x_2-x_0}+\frac{1}{x_2-x_1}\right)\right) + y'_2(x-x_2)\right)\left(\frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}\right)^2. \end{aligned}$$

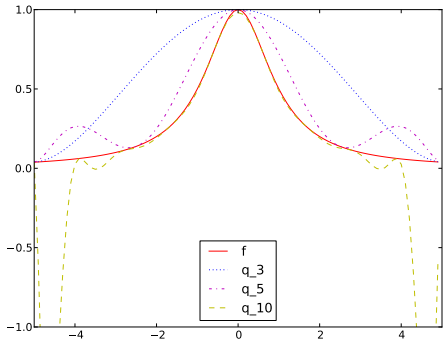
Remarque

Si n est petit on peut calculer directement les coefficients a_0, a_1, …, a_{2n+1} en résolvant le système linéaire de $2n+2$ équations

$$\begin{cases} a_0 + a_1 x_0 + \dots a_{2n+1} x_0^{2n+1} = y_0 \\ a_0 + a_1 x_1 + \dots a_{2n+1} x_1^{2n+1} = y_1 \\ \dots \\ a_n + a_1 x_n + \dots a_{2n+1} x_n^{2n+1} = y_n \\ a_1 + a_2 x_0 + \dots (2n+1) a_{2n+1} x_0^{2n+1-1} = y'_0 \\ a_1 + a_2 x_0 + \dots (2n+1) a_{2n+1} x_1^{2n+1-1} = y'_1 \\ \dots \\ a_n + a_1 x_n + \dots (2n+1) a_{2n+1} x_n^{2n+1-1} = y'_n \end{cases} \quad \text{i.e.} \quad \underbrace{\begin{pmatrix} 1 & x_0 & \dots & x_0^{2n+1} \\ 1 & x_1 & \dots & x_1^{2n+1} \\ \vdots & & \vdots & \vdots \\ 1 & x_n & \dots & x_n^{2n+1} \\ 0 & x_0 & \dots & (2n+1)x_0^{2n+1-1} \\ 0 & x_1 & \dots & (2n+1)x_1^{2n+1-1} \\ \vdots & & \vdots & \vdots \\ 0 & x_n & \dots & (2n+1)x_n^{2n+1-1} \end{pmatrix}}_{(2n+2)\times(2n+2)} \underbrace{\begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ \\ \\ \\ a_{2n+1} \end{pmatrix}}_{(2n+2)\times 1} = \underbrace{\begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \\ y'_0 \\ y'_1 \\ \vdots \\ y'_n \end{pmatrix}}_{(2n+2)\times 1}$$

Exemple RUNGE

On veut voir si avec l'interpolation d'HERMITE on arrive à mieux approcher la fonction de RUNGE. Soit la fonction $f\colon [-5,5] \to \mathbb{R}$ définie par $f(x) = \frac{1}{1+x^2}$. La figure ci-dessous montre les polynômes interpolants de degrés 3, 5 et 10 pour une distribution équirepartie des nœuds.



Cette figure a été obtenue par les instructions :

```
from matplotlib.pylab import *

def hermite(t,x,y,dy):
    p = 0
    n = len(x)
    L = [1 for i in range(n)]
    c = [0 for i in range(n)]
    for i in range(n):
        for j in range(len(x)):
            if j!=i:
                L[i] *= (t-x[j])/(x[i]-x[j])
                c[i] += 1./(x[i]-x[j])
        p += (y[i]*(1.-2.*(t-x[i])*c[i])+dy[i]*(t-x[i]))*L[i]**2
    return p

def f(x):
    return 1./(1.+x**2)

def df(x):
    return -2.*x/(1.+x**2)**2

# INPUT
x1 = linspace(-5,5,3)
x2 = linspace(-5,5,5)
x3 = linspace(-5,5,10)
y1 = f(x1)
y2 = f(x2)
y3 = f(x3)
dy1 = df(x1)
dy2 = df(x2)
dy3 = df(x3)

# Calcul des polynomes en plusieurs points d'un intervalle pour affichage
t = arange(-5,5,.1)
h1t = []
h2t = []
h3t = []
for k in t:
    h1t.append(hermite(k,x1,y1,dy1))
    h2t.append(hermite(k,x2,y2,dy2))
    h3t.append(hermite(k,x3,y3,dy3))

plot(t,f(t),'r-',t,h1t,'b:',t,h2t,'m-.',t,h3t,'y--')
legend(['f','q_3','q_5','q_10'],loc='lower center')
axis([-5, 5, -1, 1])
show()
```

Même avec l'interpolation d'HERMITE on voit que l'erreur tend vers l'infini quand n tend vers l'infini pour une distribution uniforme des nœuds.

Algorithmes

LAGRANGE :

Require: t, n, $\{(x_i, y_i)\}_{i=0}^{n}$
  $p \leftarrow 0$
  for $i = 0$ to n **do**
    $L_i \leftarrow 1$
    for $j = 0$ to n **do**
      if $j \neq i$ **then**
        $L_i \leftarrow \dfrac{t - x_j}{x_i - x_j} \times L_i$
      end if
    end for
    $p \leftarrow p + y_i \times L_i$
  end for
  return p

HERMITE :

Require: t, n, $\{(x_i, y_i, y_i')\}_{i=0}^{n}$
  $p \leftarrow 0$
  for $i = 0$ to n **do**
    $L_i \leftarrow 1$
    for $j = 0$ to n **do**
      if $j \neq i$ **then**
        $L_i \leftarrow \dfrac{t - x_j}{x_i - x_j} \times L_i$
        $c_i \leftarrow \dfrac{1}{x_i - x_j} + c_i$
      end if
    end for
    $p \leftarrow p + \big(y_i \times (1 - 2(t - x_i) \times c_i) + y_i' \times (t - x_i)\big) \times L_i^2$
  end for
  return p

2.3. Splines : interpolation par morceaux

On a mis en évidence le fait qu'on ne peut pas garantir la convergence uniforme du polynôme interpolatoire de LAGRANGE vers f quand les nœuds d'interpolation sont équirépartis. L'interpolation de LAGRANGE de bas degré est cependant suffisamment précise quand elle est utilisée sur des intervalles assez petits, y compris avec des nœuds équirépartis (ce qui est commode en pratique). Il est donc naturel d'introduire une partition de $[a;b]$ en n sous-intervalles $[x_i, x_{i+1}]$, tels que $[a;b] = \cup_{0\leq i\leq n-1}[x_i, x_{i+1}]$ et d'utiliser l'interpolation de LAGRANGE sur chaque sous-intervalles $[x_i, x_{i+1}]$ en utilisant m nœuds équirépartis avec m petit (généralement $m = 1$ ou 3).

Définition

Étant donné $n+1$ points distincts $x_0, \ldots, x_n$ de $[a;b]$ avec $a = x_0 < x_1 < \cdots < x_n = b$, la fonction $s_k \colon [a;b] \to \mathbb{R}$ est une spline de degré k relative aux nœuds $\{x_i\}$ si

$$\begin{cases} s_k(x)|_{[x_i;x_i+1]} \in \mathbb{R}_k[x], \quad i = 0, 1, \ldots, n-1, \\ s_k \in \mathscr{C}^{k-1}([a;b]). \end{cases}$$

Évidemment tout polynôme de degré k est une spline, mais en pratique une spline est constituée de polynômes différents sur chaque sous-intervalle. Il peut donc y avoir des discontinuités de la dérivée k-ième aux nœuds internes $x_1, \ldots, x_{n-1}$.

2.3.1. Interpolation linéaire composite

Étant donné une distribution (non nécessairement uniforme) de nœuds $x_0 < x_1 < \cdots < x_n$, on approche f par une fonction continue qui, sur chaque intervalle $[x_i, x_{i+1}]$, est définie par le segment joignant les deux points $(x_i, f(x_i))$ et $(x_{i+1}, f(x_{i+1}))$. Cette fonction est appelée interpolation linéaire par morceaux (ou *spline* linéaire).

Définition *Splines linéaires*

Étant donné $n+1$ points distincts $x_0, \ldots, x_n$ de $[a;b]$ avec $a = x_0 < x_1 < \cdots < x_n = b$, la fonction $\ell \colon [a;b] \to \mathbb{R}$ est une spline linéaire relative aux nœuds $\{x_i\}$ si

$$\begin{cases} \ell(x)|_{[x_i;x_i+1]} \in \mathbb{R}_1, \quad i = 0, 1, \ldots, n-1, \\ \ell \in \mathscr{C}^0([a;b]). \end{cases}$$

Autrement dit, dans chaque sous-intervalle $[x_i; x_i + 1]$, la fonction $\ell \colon [x_i, x_{i+1}] \to \mathbb{R}$ est le segment qui connecte le point (x_i, y_i) au point (x_{i+1}, y_{i+1}) ; elle s'écrit donc

$$\ell(x)|_{[x_i;x_i+1]} = y_i + \frac{y_{i+1} - y_i}{x_{i+1} - x_i}(x - x_i)$$

Il est intéressant de noter que la commande `plot(x,y)`, utilisée pour afficher le graphe d'une fonction f sur un intervalle donné $[a,b]$, remplace en fait la fonction par une interpolée linéaire par morceaux, les points d'interpolation étant les composantes du vecteur `x`.

Proposition *Erreur*

Si $y_i = f(x_i)$ pour $i = 0, 1, \ldots, n$ et $f : [a;b] \to \mathbb{R}$ est une fonction donnée de classe $\mathscr{C}^2([a;b])$, alors on peut majorer l'erreur d'interpolation au point $x \in [a;b]$ par

$$\max_{x1[a;b]} |f(x) - \ell(x)| \leq \frac{h^2}{8} \max_{x\in[a;b]} |f''(x)|,$$

où $h = \max_{i=0,\ldots,n-1} x_{i+1} - x_i$. Par conséquent, pour tout x dans l'intervalle $[a;b]$, $\ell(x)$ tend vers $f(x)$ quand $n \to +\infty$, à condition que f soit assez régulière.

Le principale défaut de cette interpolation par morceaux est que ℓ n'est que continue. Or, dans des nombreuses applications, il est préférable d'utiliser des fonctions ayant au moins une dérivée continue. On peut construire pour cela une fonction s_3 comme l'interpolation d'Hermite des points $(x_i, f(x_i), f'(x_i))$ et $(x_{i+1}, f(x_{i+1}), f'(x_{i+1}))$ sur chaque $[x_i; x_i + 1]$ pour $i = 0, 1, \ldots, n-1$.

✶✶✶✶✶✶✶✶✶✶✶✶✶ Codes Python ✶✶✶✶✶✶✶✶✶✶✶✶✶

Voici les `function` python des méthodes illustrées dans ce chapitre : `t` est le point où on veut évaluer le polynôme d'interpolation, `x` est une liste qui contient les abscisses des points d'interpolation, `y` est une liste qui contient les ordonnées des points d'interpolation et `dy` est une liste qui contient la valeur de la dérivée aux points d'interpolation. Elles renvoient l'évaluation du polynôme en `t`.

Méthodes numériques.

```
def lagrange(t,x,y):
    p = 0
    n = len(x)
    L = [1 for i in range(n)]
    for i in range(n):
        for j in range(n):
            if j!=i:
                L[i] *= (t-x[j])/(x[i]-x[j])
        p += y[i]*L[i]
    return p

def divided_difference(xx,yy):
    n = len(xx)
    # Initialisation de la matrice vide
    A = []
    for i in range(n):
    A+=[[]] # ajoute n fois une sous-liste vide : [[],[],[],[]]
    for j in range(n):
        A[i]+=[0] # ajoute n lments '0'  chacune des n sous-listes vides
    # On rempli la partie triangulaire inferieure
    for i in range(n):
        A[i][0]=float(yy[i])
        for j in range(1,i+1):
            A[i][j]=(float(A[i][j-1])-float(A[i-1][j-1]))/(float(xx[i])-float(xx[i-j]))
    return [ A[i][i] for i in range(n) ]

def newton(t,xx,yy):
    p = 0
    n = len(xx)
    OMEGA = [1. for i in range(n+1)]
    DD = divided_difference(xx,yy)
    for i in range(n):
        p += DD[i]*OMEGA[i]
        OMEGA[i+1] = OMEGA[i] * float(t-xx[i])
    return p

def hermite(t,x,y,dy):
    p = 0
    n = len(x)
    L = [1 for i in range(n)]
    c = [0 for i in range(n)]
    for i in range(n):
        for j in range(len(x)):
            if j!=i:
                L[i] *= (t-x[j])/(x[i]-x[j])
                c[i] += 1./(x[i]-x[j])
        p += (y[i]*(1.-2.*(t-x[i])*c[i])+dy[i]*(t-x[i]))*L[i]**2
    return p
```

et voici un exemple d'utilisation de ces fonctions :

Cas test.

```
from matplotlib.pylab import *

# INPUT
x = [1,2,3,4,5]
y = [0,1,0,1,0]
dy = [-1,1,0,-1,0]
```

```
55
56 # Calcul des polynomes en un point
57 t = 1.5
58 print "La valeur du polynome de Lagrange en", t, "est", lagrange(t,x,y)
59 print "La valeur du polynome de Newton en", t, "est", newton(t,xx,yy)
60 print "La valeur du polynome d'Hermite en", t, "est", hermite(t,x,y,dy)
61
62 # Calcul des polynomes en plusieurs points d'un intervalle pour affichage
63 axis([0, 6, -2, 2])
64 t = arange(0,6,.1)
65 lt = []
66 nt = []
67 ht = []
68 for k in t:
69 ⟶lt.append(lagrange(k,x,y))
70 ⟶nt.append(newton(k,x,y))
71 ⟶ht.append(hermite(k,x,y,dy))
72 plot(x,y,'ro',t,lt,'b',t,nt,'g.',t,ht,'m')
73 show()
```

Exercices

Exercice 2.1

Construire le polynôme P qui interpole les points $(0,2)$, $(1,1)$, $(2,2)$ et $(3,3)$.

Correction de l'exercice 2.1. On cherche un polynôme de degré au plus 3 tel que $P(0)=2$, $P(1)=1$, $P(2)=2$ et $P(3)=3$. Construire P signifie trouver ses coordonnées dans une base de $\mathbb{R}_3[x]$. On considère trois méthodes qui sont basées sur trois choix différents de bases de $\mathbb{R}_3[x]$:

- **Méthode directe (naïve)**

  On considère $\mathscr{C}=\{1,x,x^2,x^3\}$ la base canonique de $\mathbb{R}_3[x]$ et on cherche $(a_0,a_1,a_2,a_3)=\text{coord}(P,\mathscr{C})$, *i.e.* a_0,a_1,a_2,a_3 tels que $P(x)=\sum_{i=0}^3 a_i x^i$.

  Il s'agit de trouver les 4 coefficients a_0, a_1, a_2 et a_3 solution du système linéaire

$$\begin{cases} p(0)=2\\ p(1)=1\\ p(2)=2\\ p(3)=3 \end{cases} \iff \begin{cases} a_0+a_1\cdot 0+a_2\cdot 0^2+a_3\cdot 0^3=2\\ a_0+a_1\cdot 1+a_2\cdot 1^2+a_3\cdot 1^3=1\\ a_0+a_1\cdot 2+a_2\cdot 2^2+a_3\cdot 2^3=2\\ a_0+a_1\cdot 3+a_2\cdot 3^2+a_3\cdot 3^3=3 \end{cases} \iff \begin{pmatrix} 1&0&0&0\\ 1&1&1&1\\ 1&2&4&8\\ 1&3&9&27 \end{pmatrix}\begin{pmatrix} a_0\\ a_1\\ a_2\\ a_3 \end{pmatrix}=\begin{pmatrix} 2\\ 1\\ 2\\ 3 \end{pmatrix}$$

$$\left(\begin{array}{cccc|c} 10&0&0&2&2\\ 11&1&1&1&1\\ 12&4&8&2&2\\ 13&9&27&3&3 \end{array}\right) \xrightarrow[L_4\leftarrow L_4-L_1]{\substack{L_2\leftarrow L_2-L_1\\ L_3\leftarrow L_3-L_1}} \left(\begin{array}{cccc|c} 1&0&0&0&2\\ 0&1&1&1&-1\\ 0&2&4&8&0\\ 0&3&9&27&1 \end{array}\right) \xrightarrow[L_4\leftarrow L_4-3L_2]{\substack{L_3\leftarrow L_3-2L_2}} \left(\begin{array}{cccc|c} 1&0&0&0&2\\ 0&1&1&1&-1\\ 0&0&2&6&2\\ 0&0&6&24&4 \end{array}\right) \xrightarrow{L_4\leftarrow L_4-3L_3} \left(\begin{array}{cccc|c} 1&0&0&0&2\\ 0&1&1&1&-1\\ 0&0&2&6&2\\ 0&0&0&6&-2 \end{array}\right)$$

  donc $a_3=-\frac{1}{3}$, $a_2=2$, $a_1=-\frac{8}{3}$ et $a_0=2$ et on trouve $P(x)=2-\frac{8}{3}x+2x^2-\frac{1}{3}x^3$.

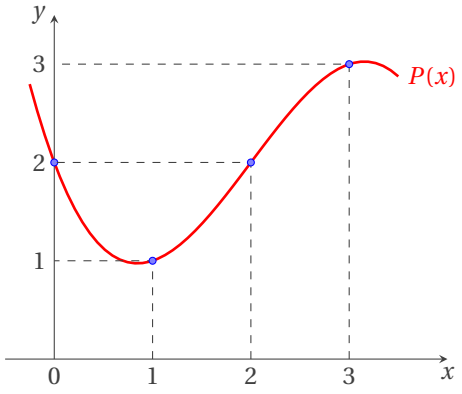
- **Méthode de Lagrange**

  On considère $\mathscr{L}=\{L_0,L_1,L_2,L_3\}$ une base de $\mathbb{R}_3[x]$ telle que $\text{coord}(P,\mathscr{L})=(y_0,y_1,y_2,y_3)$, *i.e.* $P(x)=\sum_{i=0}^3 y_i L_i(x)$. On a

$$L_i(x)=\prod_{\substack{j=0\\ j\neq i}}^{n}\frac{x-x_j}{x_i-x_j}$$

  donc

$$\begin{aligned} P(x)&=y_0\frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)}+y_1\frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)}\\ &\quad+y_2\frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)}+y_3\frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}=\\ &=2\frac{(x-1)(x-2)(x-3)}{(0-1)(0-2)(0-3)}+\frac{(x-0)(x-2)(x-3)}{(1-0)(1-2)(1-3)}\\ &\quad+2\frac{(x-0)(x-1)(x-3)}{(2-0)(2-1)(2-3)}+3\frac{(x-0)(x-1)(x-2)}{(3-0)(3-1)(3-2)}=\\ &=\frac{(x-1)(x-2)(x-3)}{-3}+\frac{x(x-2)(x-3)}{2}\\ &\quad-x(x-1)(x-3)+\frac{x(x-1)(x-2)}{2}=-\frac{1}{3}x^3+2x^2-\frac{8}{3}x+2. \end{aligned}$$



- **Méthode de Newton**

  On considère $\mathscr{N}=\{\omega_0,\omega_1,\omega_2,\omega_3\}$ une base de $\mathbb{R}_3[x]$ telle que $\text{coord}(p,\mathscr{N})=(y_0,f[x_0,x_1],f[x_0,x_1,x_2],f[x_0,x_1,x_2,x_3])$, *i.e.* $P(x)=\sum_{i=0}^3 f[x_0,\dots,x_i]\omega_i(x)$.

  La base de Newton est définie récursivement comme suit :

$$\omega_0(x)=0;\qquad \omega_1(x)=x-x_0;\qquad \text{pour } k=2,\dots,n\quad \omega_k(x)=\omega_{k-1}(x)(x-x_{k-1}).$$

  Les coordonnées sont les valeurs encadrées dans le tableau des différences divisées ci-dessous :

i	x_i	y_i	$f[x_{i-1},x_i]$	$f[x_{i-2},x_{i-1},x_i]$	$f[x_{i-3},x_{i-2},x_{i-1},x_i]$
0	0	$\boxed{2}$			
1	1	1	$\boxed{-1}$		
2	2	2	1	$\boxed{1}$	
3	3	3	1	0	$\boxed{-\frac{1}{3}}$

On a alors

$$\begin{aligned}P_3(x) &= \sum_{i=0}^{3} f[x_0,\dots,x_i]\omega_i(x)\\ &= y_0\omega_0(x) + f[x_0,x_1]\omega_1(x) + f[x_0,x_1,x_2]\omega_2(x) + f[x_0,x_1,x_2,x_3]\omega_3(x)\\ &= 2\omega_0(x) - \omega_1(x) + \omega_2(x) - \frac{1}{3}\omega_3(x)\\ &= 2 - x + x(x-1) - \frac{1}{3}x(x-1)(x-2)\\ &= -\frac{1}{3}x^3 + 2x^2 - \frac{8}{3}x + 2.\end{aligned}$$

Exercice 2.2

1. Calculer le polynôme d'interpolation de la fonction $f(x) = \cos(x)$ en les 3 points $x_i = \frac{\pi}{2}i$ avec $i = 0,\dots,2$.
2. Calculer ensuite le polynôme d'interpolation de la même fonction en les 4 points $x_i = \frac{\pi}{2}i$ avec $i = 0,\dots,3$, *i.e.* en ajoutant le point $x_3 = 3\pi/2$.

CORRECTION DE L'EXERCICE 2.2.

1. On cherche $p_2 \in \mathbb{R}_2[x]$ tel que $p_2(x_i) = \cos(x_i)$ pour $i = 0,\dots,2$. On peut choisir l'une des quatre méthodes ci-dessous (on préférera la méthode de NEWTON car elle permet de réutiliser les calculs de cette question pour répondre à la question suivante).

   Méthode directe (naïve). Si on écrit $p_2(x) = \alpha_0 + \alpha_1 x + \alpha_2 x^2$, on cherche $\alpha_0, \alpha_1, \alpha_2$ tels que

   $$\begin{pmatrix} 1 & 0 & 0 \\ 1 & \frac{\pi}{2} & \frac{\pi^2}{4} \\ 1 & \pi & \pi^2 \end{pmatrix}\begin{pmatrix}\alpha_0\\ \alpha_1\\ \alpha_2\end{pmatrix} = \begin{pmatrix}1\\0\\-1\end{pmatrix}$$

   En résolvant ce système linéaire on trouve $\alpha_0 = 0$, $\alpha_1 = -\frac{2}{\pi}$ et $\alpha_2 = 0$.

   Méthode astucieuse. Le polynôme p_2 s'annule en $\frac{\pi}{2}$, ceci signifie qu'il existe un polynôme $R(x)$ tel que

   $$p_2(x) = R(x)\left(x - \frac{\pi}{2}\right).$$

   Puisque $p_2(x)$ a degré 2, le polynôme $R(x)$ qu'on a mis en facteur a degré 1, autrement dit R est de la forme $ax + b$. On cherche alors a et b tels que

   $$\begin{cases} R(0) = \frac{p_2(0)}{(0-\frac{\pi}{2})}, \\ R(\pi) = \frac{p_2(\pi)}{(\pi-\frac{\pi}{2})}. \end{cases} \iff \begin{cases} b = \frac{1}{(0-\frac{\pi}{2})}, \\ a\pi + b = \frac{-1}{(\pi-\frac{\pi}{2})}. \end{cases} \iff \begin{cases} b = -\frac{2}{\pi}, \\ a = 0. \end{cases}$$

   Ainsi

   $$p_2(x) = R(x)\left(x - \frac{\pi}{2}\right) = -\frac{2}{\pi}\left(x - \frac{\pi}{2}\right) = -\frac{2}{\pi}x + 1.$$

   Méthode de Lagrange. On a

   $$p_2(x) = y_0L_0(x) + y_1L_1(x) + y_2L_2(x) = 1\frac{\left(x-\frac{\pi}{2}\right)(x-\pi)}{\left(0-\frac{\pi}{2}\right)(0-\pi)} - 1\frac{(x-0)\left(x-\frac{\pi}{2}\right)}{(\pi-0)\left(\pi-\frac{\pi}{2}\right)} = 1 - \frac{2}{\pi}x.$$

   Méthode de Newton. On commence par construire le tableau des différences divisées :

i	x_i	y_i	$f[x_{i-1},x_i]$	$f[x_{i-2},x_{i-1},x_i]$
0	0	$\boxed{1}$		
1	$\frac{\pi}{2}$	0	$\boxed{-\frac{2}{\pi}}$	
2	π	-1	$-\frac{2}{\pi}$	$\boxed{0}$

   On a alors

   $$p_2(x) = \sum_{i=0}^{2}\omega_i(x)f[x_0,\dots,x_i]$$

$$
\begin{aligned}
&= \omega_0(x) f[x_0] + \omega_1(x) f[x_0, x_1] + \omega_2(x) f[x_0, x_1, x_2] \\
&= \omega_0(x) - \frac{2}{\pi}\omega_1(x) \\
&= 1 - \frac{2}{\pi}x.
\end{aligned}
$$

2. On cherche donc $p_3 \in \mathbb{R}_3[x]$ tel que $p_3(x_i) = \sin(x_i)$ pour $i = 0, \dots, 3$. On peut choisir l'une des quatre méthodes ci-dessous (on préférera la méthode de NEWTON car elle permet d'utiliser les calculs précédents).

Méthode directe. Si on écrit $p_3(x) = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3$, on cherche $\alpha_0, \alpha_1, \alpha_2, \alpha_3$ tels que

$$
\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & \frac{\pi}{2} & \frac{\pi^2}{4} & \frac{\pi^3}{8} \\ 1 & \pi & \pi^2 & \pi^3 \\ 1 & \frac{3\pi}{2} & \frac{9\pi^2}{4} & \frac{27\pi^3}{8} \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ -1 \\ 0 \end{pmatrix}
$$

En résolvant ce système linéaire on trouve $\alpha_0 = 1$, $\alpha_1 = -\frac{2}{3\pi}$, $\alpha_2 = -\frac{4}{\pi^2}$ et $\alpha_3 = \frac{8}{3\pi^3}$.

Méthode astucieuse. Le polynôme p_3 s'annule en $\frac{\pi}{2}$ et en $\frac{3\pi}{2}$, ceci signifie qu'il existe un polynôme $R(x)$ tel que

$$
p_3(x) = R(x)\left(x - \frac{\pi}{2}\right)\left(x - \frac{3\pi}{2}\right).
$$

Puisque $p_3(x)$ a degré 3, le polynôme $R(x)$ qu'on a mis en facteur a degré 1, autrement dit R est de la forme $ax + b$. On cherche alors a et b tels que

$$
\begin{cases} R(0) = \frac{p_3(0)}{(0-\frac{\pi}{2})(0-\frac{3\pi}{2})}, \\ R(\pi) = \frac{p_3(\pi)}{(\pi-\frac{\pi}{2})(\pi-\frac{3\pi}{2})}. \end{cases} \iff \begin{cases} b = \frac{1}{(0-\frac{\pi}{2})(0-\frac{3\pi}{2})}, \\ a\pi + b = \frac{-1}{(\pi-\frac{\pi}{2})(\pi-\frac{3\pi}{2})}. \end{cases} \iff \begin{cases} b = \frac{4}{3\pi^2}, \\ a = \frac{8}{3\pi^3}. \end{cases}
$$

Ainsi

$$
p_3(x) = R(x)\left(x - \frac{\pi}{2}\right)\left(x - \frac{3\pi}{2}\right) = \left(\frac{8}{3\pi^3}x + \frac{4}{3\pi^2}\right)\left(x - \frac{\pi}{2}\right)\left(x - \frac{3\pi}{2}\right) = 1 - \frac{2}{3\pi}x - \frac{4}{\pi^2}x^2 + \frac{8}{3\pi^3}x^3.
$$

Méthode de Lagrange. On a

$$
\begin{aligned}
p_3(x) &= y_0 L_0(x) + y_1 L_1(x) + y_2 L_2(x) + y_3 L_3(x) = 1\frac{\left(x-\frac{\pi}{2}\right)(x-\pi)\left(x-\frac{3\pi}{2}\right)}{\left(0-\frac{\pi}{2}\right)(0-\pi)\left(0-\frac{3\pi}{2}\right)} - 1\frac{(x-0)\left(x-\frac{\pi}{2}\right)\left(x-\frac{3\pi}{2}\right)}{(\pi-0)\left(\pi-\frac{\pi}{2}\right)\left(\pi-\frac{3\pi}{2}\right)} \\
&= \frac{4}{3\pi^3}\left(x-\frac{\pi}{2}\right)\left(x-\frac{3\pi}{2}\right)(-x+\pi+3x) = 1 - \frac{2}{3\pi}x - \frac{4}{\pi^2}x^2 + \frac{8}{3\pi^3}x^3.
\end{aligned}
$$

Méthode de Newton. Il suffit de calculer une différence divisée en plus, *i.e.* ajouter une ligne au tableau précédant :

i	x_i	y_i	$f[x_{i-1}, x_i]$	$f[x_{i-2}, x_{i-1}, x_i]$	$f[x_{i-3}, x_{i-2}, x_{i-1}, x_i]$
0	0	$\boxed{1}$			
1	$\frac{\pi}{2}$	0	$\boxed{-\frac{2}{\pi}}$		
2	π	-1	$-\frac{2}{\pi}$	$\boxed{0}$	
3	$\frac{3\pi}{2}$	0	$\frac{2}{\pi}$	$\frac{4}{\pi^2}$	$\boxed{\frac{8}{3\pi^3}}$

On a alors

$$
\begin{aligned}
p_3(x) &= \sum_{i=0}^{3} \omega_i(x) f[x_0, \dots, x_i] \\
&= p_2(x) + \omega_3(x) f[x_0, x_1, x_2, x_3] \\
&= 1 - \frac{2}{\pi}x + \frac{8}{3\pi^3}\omega_3(x) \\
&= 1 - \frac{2}{\pi}x + \frac{8}{3\pi^3}x\left(x-\frac{\pi}{2}\right)(x-\pi) \\
&= 1 - \frac{2}{3\pi}x - \frac{4}{\pi^2}x^2 + \frac{8}{3\pi^3}x^3.
\end{aligned}
$$

Exercice 2.3

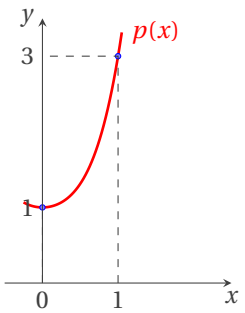
Trouver le polynôme de l'espace vectoriel $\mathrm{Vec}\{1+x^2, x^4\}$ qui interpole les points $(0,1)$ et $(1,3)$.

Correction de l'exercice 2.3.

Il s'agit de trouver un polynôme $p(x)$ qui soit combinaison linéaire des deux polynômes assignés (*i.e.* $p(x) = \alpha(1+x^2) + \beta(x^4)$) et qui interpole les deux points $(0,1)$ et $(1,3)$:

$$\begin{cases} p(0) = 1, \\ p(1) = 3, \end{cases} \quad \Leftrightarrow \quad \begin{cases} \alpha(1+0^2) + \beta(0^4) = 1, \\ \alpha(1+1^2) + \beta(1^4) = 3, \end{cases}$$

d'où $\alpha = 1$ et $\beta = 1$. Le polynôme cherché est donc le polynôme $p(x) = 1 + x^2 + x^4$.



Exercice 2.4

1. Construire le polynôme de Lagrange P qui interpole les points $(-1,2)$, $(0,1)$, $(1,2)$ et $(2,3)$.
2. Soit Q le polynôme de Lagrange qui interpole les points $(-1,2)$, $(0,1)$, $(1,2)$. Montrer qu'il existe un réel λ tel que :

$$Q(x) - P(x) = \lambda(x+1)x(x-1).$$

Correction de l'exercice 2.4. Le polynôme d'interpolation de Lagrange de degré n sur l'ensemble des $n+1$ points $\{(x_i, y_i)\}_{i=0}^n$ s'écrit

$$p_n(x) = \sum_{i=0}^{n} \left(y_i \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{x - x_j}{x_i - x_j} \right).$$

1. Ici $n = 3$ donc on a

$$\begin{aligned} P(x) &= y_0 \frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} + y_1 \frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)} \\ &\quad + y_2 \frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} + y_3 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)} \\ &= \frac{x(x-1)(x-2)}{-3} + \frac{(x+1)(x-1)(x-2)}{2} - (x+1)x(x-2) + \frac{(x+1)x(x-1)}{2} = \\ &= -\frac{1}{3}x^3 + x^2 + \frac{1}{3}x + 1. \end{aligned}$$

2. Par construction

$$\begin{aligned} Q(-1) &= P(-1), \\ Q(0) &= P(0), \\ Q(1) &= P(1), \end{aligned}$$

donc le polynôme $Q(x) - P(x)$ s'annule en -1, en 0 et en 1, ceci signifie qu'il existe un polynôme $R(x)$ tel que

$$Q(x) - P(x) = R(x)(x+1)x(x-1).$$

Puisque $P(x)$ a degré 3 et $Q(x)$ a degré 2, le polynôme $Q(x) - P(x)$ a degré 3, donc le polynôme $R(x)$ qu'on a mis en facteur a degré 0 (*i.e.* $R(x)$ est une constante).

Si on n'a pas remarqué ça, on peut tout de même faire tous les calculs : dans ce cas $n = 2$ donc on a

$$\begin{aligned} Q(x) &= y_0 \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + y_1 \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + y_2 \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \\ &= x(x-1) - (x+1)(x-1) + (x+1)x \end{aligned}$$

$$= x^2 + 1.$$

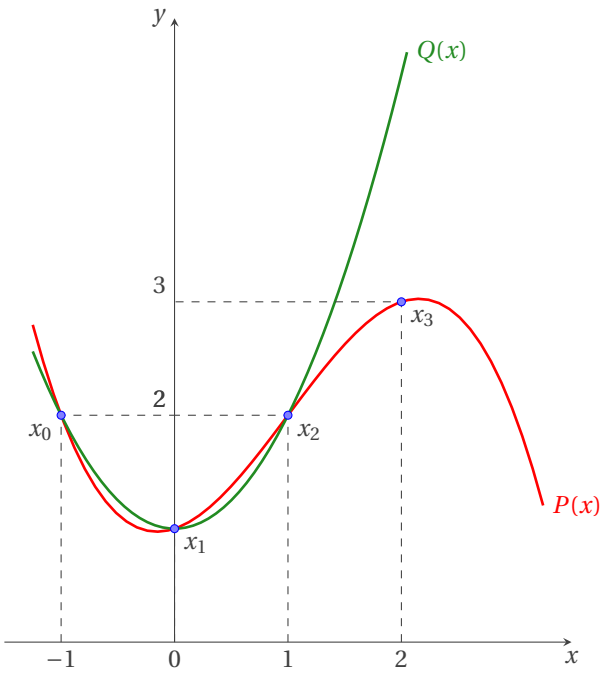
Ainsi

$$\begin{aligned}
Q(x) - P(x) &= y_0 \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}\left[1 - \frac{x-x_3}{x_0-x_3}\right] + y_1 \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}\left[1 - \frac{x-x_3}{x_1-x_3}\right] \\
&+ y_2 \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}\left[1 - \frac{x-x_3}{x_2-x_3}\right] - y_3 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)} \\
&= -y_0 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} - y_1 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)} \\
&- y_2 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} - y_3 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)} \\
&= -\left[\frac{y_0}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} + \frac{y_1}{(x_1-x_0)(x_1-x_2)(x_1-x_3)}\right. \\
&\left. + \frac{y_2}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} + \frac{y_3}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}\right](x-x_0)(x-x_1)(x-x_2) \\
&= \frac{(x+1)x(x-1)}{3}
\end{aligned}$$

et $\lambda = \frac{1}{3}$. Sinon directement

$$Q(x) - P(x) = x^2 + 1 + \frac{1}{3}x^3 - x^2 + \frac{1}{3}x - 1 = \frac{1}{3}x^3 + \frac{1}{3}x = \frac{(x+1)x(x-1)}{3} = \lambda x(x+1)(x-1)$$

avec $\lambda = \frac{1}{3}$.



Exercice 2.5

1. Construire le polynôme de LAGRANGE P qui interpole les trois points $(-1, e)$, $(0, 1)$ et $(1, e)$.
2. Sans faire de calculs, donner l'expression du polynôme de LAGRANGE Q qui interpole les trois points $(-1, -1)$, $(0, 0)$ et $(1, -1)$.
3. Trouver le polynôme de l'espace vectoriel $\mathrm{Vec}\{1, x, x^2\}$ qui interpole les trois points $(-1, -1)$, $(0, 0)$ et $(1, -1)$.

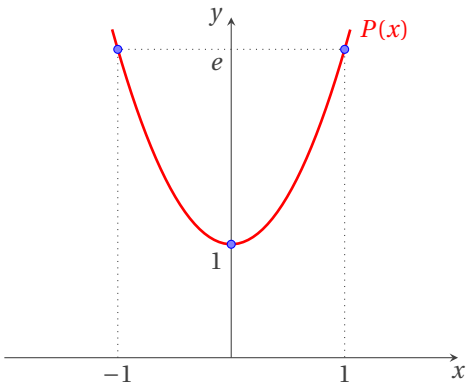
CORRECTION DE L'EXERCICE 2.5.

1. Le polynôme d'interpolation de LAGRANGE de degré n sur l'ensemble des $n+1$ points $\{(x_i, y_i)\}_{i=0}^n$ s'écrit

$$p_n(x) = \sum_{i=0}^{n} \left(y_i \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j} \right).$$

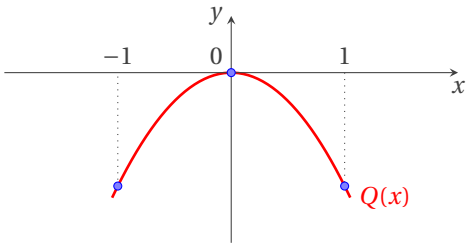
Ici $n = 2$ donc on a

$$P(x) = y_0\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + y_1\frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + y_2\frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} =$$
$$= e\frac{x(x-1)}{2} - (x+1)(x-1) + e\frac{(x+1)x}{2} =$$
$$= (e-1)x^2 + 1.$$



2. Il suffit de changer les coefficients y_i dans l'expression précédente :

$$Q(x) = -\frac{x(x-1)}{2} - \frac{(x+1)x}{2} = -x^2.$$



3. Il s'agit de trouver un polynôme $p(x)$ qui soit combinaison linéaire des deux polynômes assignés (*i.e.* $p(x) = \alpha + \beta x + \gamma x^2$) et qui interpole les trois points $(-1,-1)$, $(0,0)$ et $(1,-1)$:

$$\begin{cases} p(-1) = 1, \\ p(0) = 0, \\ p(1) = -1, \end{cases} \quad \Leftrightarrow \quad \begin{cases} \alpha - \beta + \gamma = -1, \\ \alpha = 0, \\ \alpha + \beta + \gamma = -1, \end{cases}$$

d'où $\alpha = 0$, $\beta = 0$ et $\gamma = -1$. Le polynôme cherché est donc le polynôme $p(x) = -x^2$. En fait, il suffisait de remarquer que le polynôme $Q \in \mathrm{Vec}\{1, x, x^2\}$ pour conclure que le polynôme p cherché est Q lui même.

Exercice 2.6

1. Construire le polynôme de LAGRANGE P qui interpole les points $(-1,1)$, $(0,1)$, $(1,2)$ et $(2,3)$.
2. Soit Q le polynôme de LAGRANGE qui interpole les points $(-1,1)$, $(0,1)$, $(1,2)$. Montrer qu'il existe un réel λ tel que :

$$Q(x) - P(x) = \lambda(x+1)x(x-1).$$

CORRECTION DE L'EXERCICE 2.6. Le polynôme d'interpolation de LAGRANGE de degré n sur l'ensemble des $n+1$ points $\{(x_i, y_i)\}_{i=0}^n$ s'écrit

$$p_n(x) = \sum_{i=0}^{n}\left(y_i \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{x - x_j}{x_i - x_j} \right).$$

1. Ici $n = 3$ donc on a

$$P(x) = y_0\frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} + y_1\frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)}$$
$$+ y_2\frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} + y_3\frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}$$
$$= \frac{x(x-1)(x-2)}{-6} + \frac{(x+1)(x-1)(x-2)}{2} - (x+1)x(x-2) + \frac{(x+1)x(x-1)}{2} =$$
$$= -\frac{1}{6}x^3 + \frac{1}{2}x^2 + \frac{2}{3}x + 1.$$

2. Par construction

$$Q(-1) = P(-1),$$

$$Q(0) = P(0),$$
$$Q(1) = P(1),$$

donc le polynôme $Q(x) - P(x)$ s'annule en -1, en 0 et en 1, ceci signifie qu'il existe un polynôme $R(x)$ tel que

$$Q(x) - P(x) = R(x)(x+1)x(x-1).$$

Puisque $P(x)$ a degré 3 et $Q(x)$ a degré 2, le polynôme $Q(x) - P(x)$ a degré 3, donc le polynôme $R(x)$ qu'on a mis en facteur a degré 0 (*i.e.* $R(x)$ est une constante).
Si on n'a pas remarqué ça, on peut tout de même faire tous les calculs : dans ce cas $n = 2$ donc on a

$$\begin{aligned} Q(x) &= y_0 \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + y_1 \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + y_2 \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \\ &= \frac{x(x-1)}{2} - (x+1)(x-1) + (x+1)x \\ &= \frac{1}{2}x^2 + \frac{1}{2}x + 1. \end{aligned}$$

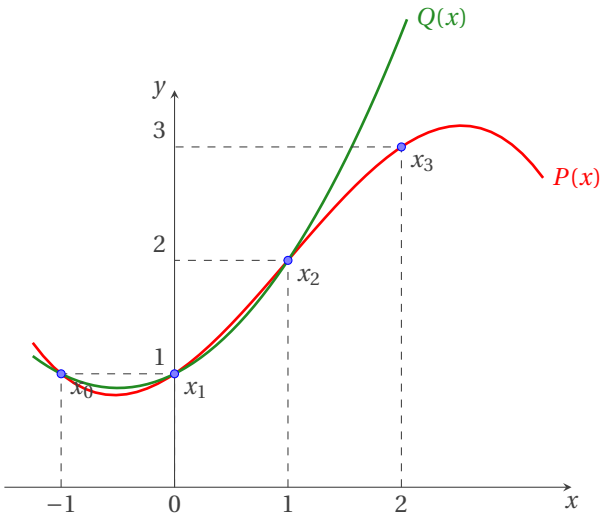
Ainsi

$$\begin{aligned} Q(x) - P(x) &= y_0 \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} \left[1 - \frac{x-x_3}{x_0-x_3}\right] + y_1 \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} \left[1 - \frac{x-x_3}{x_1-x_3}\right] \\ &\quad + y_2 \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \left[1 - \frac{x-x_3}{x_2-x_3}\right] - y_3 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)} \\ &= -y_0 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} - y_1 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)} \\ &\quad - y_2 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} - y_3 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)} \\ &= -\left[\frac{y_0}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} + \frac{y_1}{(x_1-x_0)(x_1-x_2)(x_1-x_3)}\right. \\ &\quad \left. + \frac{y_2}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} + \frac{y_3}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}\right](x-x_0)(x-x_1)(x-x_2) \\ &= \frac{(x+1)x(x-1)}{6} \end{aligned}$$

et $\lambda = \frac{1}{6}$. Sinon directement

$$Q(x) - P(x) = \frac{1}{2}x^2 + \frac{1}{2}x + 1 + \frac{1}{6}x^3 - \frac{1}{2}x^2 - \frac{2}{3}x - 1 = \frac{1}{6}x^3 - \frac{1}{6}x = \frac{1}{6}x(x^2-1) = \lambda x(x+1)(x-1)$$

avec $\lambda = \frac{1}{6}$.



Exercice 2.7

1. Construire le polynôme de LAGRANGE P qui interpole les trois points $(-1,\alpha)$, $(0,\beta)$ et $(1,\alpha)$ où α et β sont des réels.
2. Si $\alpha=\beta$, donner le degré de P.
3. Montrer que P est pair. Peut-on avoir P de degré 1 ?

CORRECTION DE L'EXERCICE 2.7.

1. Construire le polynôme de LAGRANGE P qui interpole les trois points $(-1,\alpha)$, $(0,\beta)$ et $(1,\alpha)$ où α et β sont des réels.
   Le polynôme d'interpolation de LAGRANGE de degré n sur l'ensemble des $n+1$ points $\{(x_i,y_i)\}_{i=0}^n$ s'écrit
   $$p_n(x)=\sum_{i=0}^{n}\left(y_i\prod_{\substack{j=0\\ j\neq i}}^{n}\frac{x-x_j}{x_i-x_j}\right).$$
   Ici $n=2$ donc on a
   $$\begin{aligned}P(x)&=y_0\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}+y_1\frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}+y_2\frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}+\\&=\alpha\frac{x(x-1)}{2}+\beta\frac{(x+1)(x-1)}{-1}+\alpha\frac{(x+1)x}{2}=\\&=\frac{\alpha}{2}x(x-1)-\beta(x+1)(x-1)+\frac{\alpha}{2}x(x+1)\\&=(\alpha-\beta)x^2+\beta.\end{aligned}$$
2. Si $\alpha=\beta$, $P(x)=\alpha$ qui est un polynôme de degré 0.
3. $P(-x)=P(x)$ donc P est pair. Donc P ne peut pas être de degré 1 car un polynôme de degré 1 est de la forme a_0+a_1x qui ne peut pas être pair.

Exercice 2.8

Soit $f:\mathbb{R}\to\mathbb{R}$ la fonction définie par $f(x)=1+x^3$.

1. Calculer le polynôme p_0 qui interpole f au point d'abscisse $x_0=0$.
2. Calculer le polynôme p_1 qui interpole f aux points d'abscisse $\{x_0=0,x_1=1\}$.
3. Calculer le polynôme p_2 qui interpole f aux points d'abscisse $\{x_0=0,x_1=1,x_2=2\}$.
4. Calculer le polynôme p_3 qui interpole f aux points d'abscisse $\{x_0=0,x_1=1,x_2=2,x_3=3\}$.
5. Pour $n>3$, calculer les polynômes p_n qui interpolent f aux points d'abscisse $\{x_0=0,x_1=1,\dots,x_n=n\}$.

CORRECTION DE L'EXERCICE 2.8.

1. On interpole l'ensemble $\{(0,1)\}$ donc $p_0(x)=1$.
2. On interpole l'ensemble $\{(0,1),(1,2)\}$ donc $p_1(x)=1+x$.
3. On interpole l'ensemble $\{(0,1),(1,2),(2,9)\}$ donc $p_2(x)=1-2x+3x^2$.
4. $f\in\mathbb{R}_3[x]$ et comme il existe un seul polynôme de degré au plus 3 qui interpole quatre points ce polynôme coïncide forcement avec f donc $p_3\equiv f$.
5. $f\in\mathbb{R}_n[x]$ pour tout $n\geq 3$ et comme il existe un seul polynôme de degré au plus 3 qui interpole quatre points ce polynôme coïncide forcement avec f donc $p_n\equiv f$ pour $n\geq 3$.

Exercice 2.9

Soit $\mathbb{V}_n$ la matrice de VANDERMONDE :
$$\mathbb{V}_n=\begin{pmatrix}1&a_0&a_0^2&\dots&a_0^n\\1&a_1&a_1^2&\dots&a_1^n\\1&a_2&a_2^2&\dots&a_2^n\\\vdots&\vdots&\vdots&\ddots&\vdots\\1&a_n&a_n^2&\dots&a_n^n\end{pmatrix}.$$
Quel est le lien entre cette matrice et l'interpolation polynomiale de l'ensemble de points $\{(a_i;b_i)\}_{i=0}^n$?

CORRECTION DE L'EXERCICE 2.9. Soit $p(x) = \sum_{i=0}^{n} c_i x^i$ le seul polynôme de degré n qui interpole l'ensemble de $(n+1)$ points $\{(a_i; b_i)\}_{i=0}^{n}$, *i.e.* $b_i = p(a_i) = \sum_{i=0}^{n} c_i a^i$. Alors le vecteur $\mathbf{c} = (c_0, c_1, \ldots, c_n)^T$ est solution du système linéaire $\mathbb{V}\mathbf{c} = \mathbf{b}$.

Exercice 2.10

Vérifier que le polynôme d'interpolation d'HERMITE d'une fonction f en un point coïncide avec le polynôme de TAYLOR d'ordre 1 de f en ce point.

CORRECTION DE L'EXERCICE 2.10. Le polynôme d'interpolation d'HERMITE en un point $(x_0, f(x_0), f'(x_0))$ est l'unique polynôme $q \in \mathbb{R}_1[x]$ qui vérifie $q(x_0) = f(x_0)$ et $q'(x_0) = f'(x_0)$. On cherche alors a_0 et a_1 tels que $q(x) = a_0 + a_1 x$:

$$\begin{cases} q(x_0) = f(x_0), \\ q'(x_0) = f'(x_0), \end{cases} \iff \begin{cases} a_0 + a_1 x_0 = f(x_0), \\ a_1 = f'(x_0), \end{cases} \iff \begin{cases} a_0 = f(x_0) - x_0 f'(x_0), \\ a_1 = f'(x_0), \end{cases}$$

donc $q(x) = f(x_0) + (x - x_0) f'(x_0)$.

Exercice 2.11

Soit f une fonction de classe $\mathscr{C}^1$ et $x_0 \in \mathscr{D}_f$ le domaine de définition de f. Soit ℓ le polynôme d'interpolation de LAGRANGE de f en x_0 et h le polynôme d'interpolation d'HERMITE en x_0. Calculer $h(x) - \ell(x)$.

CORRECTION DE L'EXERCICE 2.11. Le polynôme d'interpolation de LAGRANGE de f en x_0 est l'unique polynôme $\ell \in \mathbb{R}_0[x]$ qui vérifie $\ell(x_0) = f(x_0)$, donc $\ell(x) = f(x_0)$. Le polynôme d'interpolation d'HERMITE de f en x_0 est l'unique polynôme $h \in \mathbb{R}_1[x]$ qui vérifie $h(x_0) = f(x_0)$ et $h'(x_0) = f'(x_0)$. On cherche alors a_0 et a_1 tels que $h(x) = a_0 + a_1 x$:

$$\begin{cases} h(x_0) = f(x_0), \\ h'(x_0) = f'(x_0), \end{cases} \iff \begin{cases} a_0 + a_1 x_0 = f(x_0), \\ a_1 = f'(x_0), \end{cases} \iff \begin{cases} a_0 = f(x_0) - x_0 f'(x_0), \\ a_1 = f'(x_0), \end{cases}$$

donc $h(x) = f(x_0) + (x - x_0) f'(x_0)$ et $h(x) - \ell(x) = f(x_0) + (x - x_0) f'(x_0) - f(x_0) = (x - x_0) f'(x_0)$.

Exercice 2.12

Soit $f : \mathbb{R} \to \mathbb{R}$ une fonction de classe $\mathscr{C}^1(\mathbb{R})$ qui s'annule au moins une fois et dont la dérivée ne s'annule pas. Soit $x_0 \in \mathscr{D}_f$ donné. Pour $i \in \mathbb{N}$ construisons la suite $(x_i)_i$ comme suit : x_{i+1} est la racine du polynôme interpolateur d'HERMITE de f en x_i. Quelle méthode reconnait-on ? Justifier la réponse.

CORRECTION DE L'EXERCICE 2.12. Le polynôme d'HERMITE d'une fonction f en x_i a équation $q(x) = f(x_i) + (x - x_i) f'(x_i)$: il s'agit de la droite tangente au graphe de f en x_i. On cherche x_{i+1} tel que $f(x_i) + (x - x_i) f'(x_i) = 0$, d'où $x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$. On a alors la suite définie par récurrence

$$\begin{cases} x_0 \text{ donnée}, \\ x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}, \end{cases}$$

qui correspond à la méthode de NEWTON pour l'approximation de la racine de f.

Exercice 2.13

Soit f une fonction de classe $\mathscr{C}^1([-1, 1])$ et p le polynôme interpolateur d'HERMITE (de degré ≤ 3) de f vérifiant

$$p(-1) = f(-1), \qquad p'(-1) = f'(-1), \qquad p(1) = f(1), \qquad p'(1) = f'(1).$$

Écrire le polynôme p.

CORRECTION DE L'EXERCICE 2.13. On a deux points d'interpolation ($n = 1$), on cherche alors un polynôme de $\mathbb{R}_3[x]$. On a deux méthodes pour calculer le polynôme interpolateur d'HERMITE :

Première méthode : le polynôme interpolateur d'HERMITE s'écrit

$$p(x) = \sum_{i=0}^{n} \left\{ \left[y_i (1 - 2(x - x_i) c_i) + y_i' (x - x_i) \right] \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{(x - x_j)^2}{(x_i - x_j)^2} \right\} \qquad \text{où} \qquad c_i = \sum_{\substack{j=0 \\ j \neq i}}^{n} \frac{1}{x_i - x_j}.$$

Pour $n=1$ on a alors

$$p(x)=y_0\left(1-2(x-x_0)\left(\frac{1}{x_0-x_1}\right)\right)\left(\frac{(x-x_1)}{(x_0-x_1)}\right)^2+y_0'(x-x_0)\left(\frac{(x-x_1)}{(x_0-x_1)}\right)^2$$
$$+y_1\left(1-2(x-x_1)\left(\frac{1}{x_1-x_0}\right)\right)\left(\frac{(x-x_0)}{(x_1-x_0)}\right)^2+y_1'(x-x_1)\left(\frac{(x-x_0)}{(x_1-x_0)}\right)^2.$$

Dans notre cas $x_0=-1$, $x_1=1$, $y_0=f(-1)$, $y_1=f(1)$, $y_0'=f'(-1)$, $y_1'=f'(1)$ donc

$$\begin{aligned}p(x)&=\frac{1}{4}\left[f(-1)(x+2)(x-1)^2+f'(-1)(x+1)(x-1)^2+f(1)(2-x)(x+1)^2+f'(1)(x-1)(x+1)^2\right]\\&=\frac{1}{4}\Big[f(-1)(x^3-3x+2)+f'(-1)(x^3-x^2-x+1)+f(1)(-x^3+3x+2)+f'(1)(x^3+x^2-x-1)\Big]\\&=\frac{2f(-1)+f'(-1)+2f(1)-f'(1)}{4}+\frac{3f(1)-3f(-1)-f'(-1)-f'(1)}{4}x\\&\quad+\frac{f'(1)-f'(-1)}{4}x^2+\frac{f(-1)+f'(-1)-f(1)+f'(1)}{4}x^3.\end{aligned}$$

Le polynôme interpolateur d'HERMITE est donc le polynôme

$$p(x)=\alpha+\beta x+\gamma x^2+\delta x^3$$

où

$$\alpha=\frac{2f(-1)+2f(1)+f'(-1)-f'(1)}{4},\qquad \beta=\frac{-3f(-1)+3f(1)-f'(-1)-f'(1)}{4},$$
$$\gamma=\frac{-f'(-1)+f'(1)}{4},\qquad \delta=\frac{f(-1)-f(1)+f'(-1)+f'(1)}{4}.$$

Deuxième méthode : le polynôme interpolateur d'HERMITE est un polynôme de degré $2n+1$. On cherche donc un polynôme

$$p(x)=\alpha+\beta x+\gamma x^2+\delta x^3$$

tel que

$$p(-1)=f(-1),\qquad p'(-1)=f'(-1),\qquad p(1)=f(1),\qquad p'(1)=f'(1),$$

c'est-à-dire tel que

$$\begin{cases}\alpha-\beta+\gamma-\delta=f(-1),\\ \alpha+\beta+\gamma+\delta=f(1),\\ \beta-2\gamma+3\delta=f'(-1),\\ \beta+2\gamma+3\delta=f'(1).\end{cases}$$

En utilisant la méthode d'élimination de GAUSS on obtient :

$$[\mathbb{A}|\mathbf{b}]=\left(\begin{array}{cccc|c}1&-1&1&-1&f(-1)\\1&1&1&1&f(1)\\0&1&-2&3&f'(-1)\\0&1&2&3&f'(1)\end{array}\right)\xrightarrow{L_2\leftarrow L_2-L_1}\left(\begin{array}{cccc|c}1&-1&1&-1&f(-1)\\0&2&0&2&f(1)-f(-1)\\0&1&-2&3&f'(-1)\\0&1&2&3&f'(1)\end{array}\right)$$
$$\xrightarrow[L_4\leftarrow L_4-\frac{1}{2}L_3]{L_3\leftarrow L_3-\frac{1}{2}L_2}\left(\begin{array}{cccc|c}1&-1&1&-1&f(-1)\\0&2&0&2&f(1)-f(-1)\\0&0&-2&2&f'(-1)-\frac{f(1)-f(-1)}{2}\\0&0&2&2&f'(1)-\frac{f(1)-f(-1)}{2}\end{array}\right)\xrightarrow{L_4\leftarrow L_4+L_3}\left(\begin{array}{cccc|c}1&-1&1&-1&f(-1)\\0&2&0&2&f(1)-f(-1)\\0&0&-2&2&f'(-1)-\frac{f(1)-f(-1)}{2}\\0&0&0&4&f'(1)+f'(-1)-f(1)+f(-1)\end{array}\right)$$

ainsi

$$\alpha=\frac{2f(-1)+2f(1)+f'(-1)-f'(1)}{4},\qquad \beta=\frac{-3f(-1)+3f(1)-f'(-1)-f'(1)}{4},$$
$$\gamma=\frac{-f'(-1)+f'(1)}{4},\qquad \delta=\frac{f(-1)-f(1)+f'(-1)+f'(1)}{4}.$$

Exercice 2.14

1. Construire le polynôme de LAGRANGE p qui interpole les points $(-1,0)$, $(0,0)$, $(1,0)$ et $(2,0)$.
2. Construire l'ensemble des polynômes de degré 4 qui interpolent les points $(-1,0)$, $(0,0)$, $(1,0)$ et $(2,0)$.
3. Construire le polynôme d'HERMITE Q qui interpole les points $(-1,0,1)$ et $(2,0,-1)$.

CORRECTION DE L'EXERCICE 2.14.

1. Le polynôme d'interpolation de LAGRANGE de degré n sur l'ensemble des $n+1$ points $\{(x_i,y_i)\}_{i=0}^n$ s'écrit
$$p_n(x)=\sum_{i=0}^{n}\left(y_i\prod_{\substack{j=0\\ j\neq i}}^{n}\frac{x-x_j}{x_i-x_j}\right).$$
Ici $n=3$ et $y_i=0$ pour $i=0,1,2,3$ donc $p_3(x)=0$.
2. Comme les points donnés appartiennent tous à la droite d'équation $y=0$, il s'agit de construire les polynômes de degré 4 qui ont 4 racines réelles distinctes $\{x_1,x_2,x_3,x_4\}$. Ils sont tous de la forme $r_a(x)=a(x-x_1)(x-x_2)(x-x_3)(x-x_4)$; ici donc $r_a(x)=a(x+1)x(x-1)(x-2)=a(x^4-2x^3-x^2+2x)$.
3. Étant donné $n+1$ points distincts $x_0,\dots,x_n$ et $n+1$ couples correspondantes $(y_0,y_0'),\dots,(y_n,y_n')$, le polynôme d'HERMITE Q de degré $N=2n+1$ tel que $Q(x_i)=y_i$ et $Q'(x_i)=y_i'$, pour $i=0,\dots n$ s'écrit
$$Q(x)=\sum_{i=0}^{n}y_iA_i(x)+y_i'B_i(x)\ \in\mathbb{R}_N[x]\qquad\text{où}\qquad\begin{cases}L_i(x)=\prod\limits_{\substack{j=0\\ j\neq i}}^{n}\frac{x-x_j}{x_i-x_j},\\ c_i=\sum\limits_{\substack{j=0\\ j\neq i}}^{n}\frac{1}{x_i-x_j},\\ A_i(x)=(1-2(x-x_i)c_i)(L_i(x))^2,\\ B_i(x)=(x-x_i)(L_i(x))^2.\end{cases}$$
Ici $n=1$ et le polynôme d'HERMITE s'écrit
$$\begin{aligned}Q(x)&=y_0A_0+y_0'B_0+y_1A_1+y_0'B_1=B_0-B_1\\&=(x-x_0)\left(\frac{x-x_1}{x_0-x_1}\right)^2-(x-x_1)\left(\frac{x-x_0}{x_1-x_0}\right)^2=(x+1)\left(\frac{x-2}{-3}\right)^2-(x-2)\left(\frac{x+1}{3}\right)^2\\&=\frac{(x-2)^2(x+1)-(x-2)(x+1)^2}{9}=\frac{-3(x-2)(x+1)}{9}=\frac{-x^2+x+2}{3}.\end{aligned}$$
Si on a oublié la formule, il suffit de remarquer qu'on cherche un polynôme de degré 3 qui a comme racines -1 et 2 et donc qui s'écrit $Q(x)=(x+1)(x-2)(ax+b)=ax^3+(-a+b)x^2+(-b-2a)x-2b$; de plus on sait que $Q'(-1)=1$ et $Q'(2)=-1$, on trouve alors a et b en résolvant le système linéaire
$$\begin{cases}3a(-1)^2+2(-a+b)(-1)+(-b-2a)=1,\\ 3a(2)^2+2(-a+b)(2)+(-b-2a)=-1,\end{cases}\iff\begin{cases}3a+2a-2b-b-2a=1,\\ 12a-4a+4b-b-2a=-1,\end{cases}\iff\begin{cases}a=0,\\ b=-1/3.\end{cases}$$
On obtient le polynôme $Q(x)=\frac{-(x+1)(x-2)}{3}$.
Une autre idée pour calculer le polynôme Q sans utiliser la formule ni la remarque précédente est de calculer directement le polynôme selon la définition : on cherche un polynôme de degré 3, donc de la forme $Q(x)=a_0+a_1x+a_2x^2+a_3x^3$, qui vérifie $Q(-1)=0$, $Q(2)=0$, $Q'(-1)=1$ et $Q'(2)=-1$. On doit alors résoudre le système linéaire
$$\begin{cases}a_0&-a_1&+a_2&-a_3x^3&=0\\ a_0&+2a_1&+4a_2&+8a_3x^3&=0\\ &a_1&-2a_2&+3a_3x^3&=1\\ &a_1&+4a_2&+12a_3x^3&=-1\end{cases}$$
qu'on peut réécrire sous la forme $\mathbb{A}\mathbf{a}=\mathbf{b}$ avec
$$\mathbb{A}=\begin{pmatrix}1&-1&1&-1\\1&2&4&8\\0&1&-2&3\\0&1&4&12\end{pmatrix},\qquad\mathbf{a}=\begin{pmatrix}a_0\\a_1\\a_2\\a_3\end{pmatrix}\qquad\text{et}\qquad\mathbf{b}=\begin{pmatrix}0\\0\\1\\-1\end{pmatrix}$$

On utilise la méthode d'élimination de GAUSS :

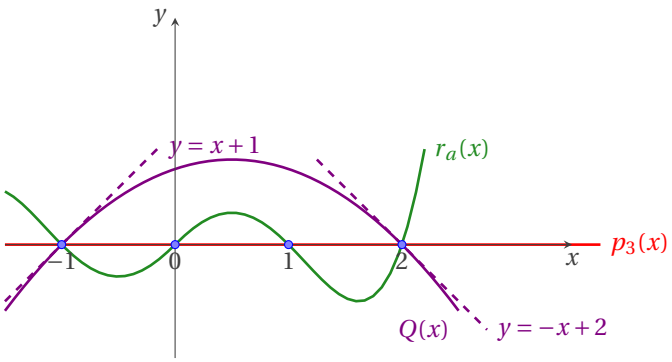
$$(\mathbb{A}|\mathbf{b}) = \left(\begin{array}{cccc|c} 1 & -1 & 1 & -1 & 0 \\ 1 & 2 & 4 & 8 & 0 \\ 0 & 1 & -2 & 3 & 1 \\ 0 & 1 & 4 & 12 & -1 \end{array}\right) \xrightarrow{L_2 \leftarrow L_2 - L_1} \left(\begin{array}{cccc|c} 1 & -1 & 1 & -1 & 0 \\ 0 & 3 & 2 & 9 & 0 \\ 0 & 1 & -2 & 3 & 1 \\ 0 & 1 & 4 & 12 & -1 \end{array}\right)$$

$$\xrightarrow[L_4 \leftarrow L_4 - L_3/3]{L_3 \leftarrow L_3 - L_2/3} \left(\begin{array}{cccc|c} 1 & -1 & 1 & -1 & 0 \\ 0 & 3 & 2 & 9 & 0 \\ 0 & 0 & -3 & 0 & 1 \\ 0 & 0 & 3 & 9 & -1 \end{array}\right) \xrightarrow{L_4 \leftarrow L_4 + L_3} \left(\begin{array}{cccc|c} 1 & -1 & 1 & -1 & 0 \\ 0 & 3 & 2 & 9 & 0 \\ 0 & 0 & -3 & 0 & 1 \\ 0 & 0 & 0 & 9 & 0 \end{array}\right)$$

et finalement on obtient

$$a_3 = 0, \qquad a_2 = -\frac{1}{3}, \qquad a_1 = \frac{1}{3}, \qquad a_0 = \frac{2}{3},$$

d'où $Q(x) = \dfrac{-x^2 + x + 2}{3}$.



Exercice 2.15

Montrer qu'il n'existe aucun polynôme p de $\mathbb{R}_3[x]$ tel que

$$p(-1) = 1, \qquad p'(-1) = 1, \qquad p'(1) = 2, \qquad p(2) = 1.$$

CORRECTION DE L'EXERCICE 2.15. Si on écrit $p(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3$, on cherche quatre coefficients a_0, a_1, a_2, a_3 solution du système linéaire

$$\begin{pmatrix} 1 & -1 & 1 & -1 \\ 0 & 1 & -2 & 3 \\ 0 & 1 & 2 & 3 \\ 1 & 2 & 4 & 8 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 2 \\ 1 \end{pmatrix}$$

Comme

$$\operatorname{rg} \begin{pmatrix} 1 & -1 & 1 & -1 \\ 0 & 1 & -2 & 3 \\ 0 & 1 & 2 & 3 \\ 1 & 2 & 4 & 8 \end{pmatrix} = 3, \qquad \operatorname{rg} \begin{pmatrix} 1 & -1 & 1 & -1 & 1 \\ 0 & 1 & -2 & 3 & 1 \\ 0 & 1 & 2 & 3 & 2 \\ 1 & 2 & 4 & 8 & 1 \end{pmatrix} = 4,$$

le système linéaire n'admet pas de solutions. (On arrive à la même conclusion en utilisant la méthode du pivot de GAUSS).

Exercice 2.16

L'espérance de vie dans un pays a évoluée dans le temps selon le tableau suivant :

Année	1975	1980	1985	1990
Espérance	72,8	74,2	75,2	76,4

Utiliser l'interpolation de LAGRANGE pour estimer l'espérance de vie en 1977, 1983 et 1988. La comparer avec une interpolation linéaire par morceaux.

CORRECTION DE L'EXERCICE 2.16. Le polynôme d'interpolation de LAGRANGE de degré n sur l'ensemble des $n+1$ points $\{(x_i, y_i)\}_{i=0}^{n}$ s'écrit

$$p_n(x) = \sum_{i=0}^{n} \left(y_i \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{x - x_j}{x_i - x_j} \right).$$

Ici $n = 3$ et si on choisit de poser $x_0 = 0$ pour l'année 1975, $x_1 = 5$ pour l'année 1980 etc., on a

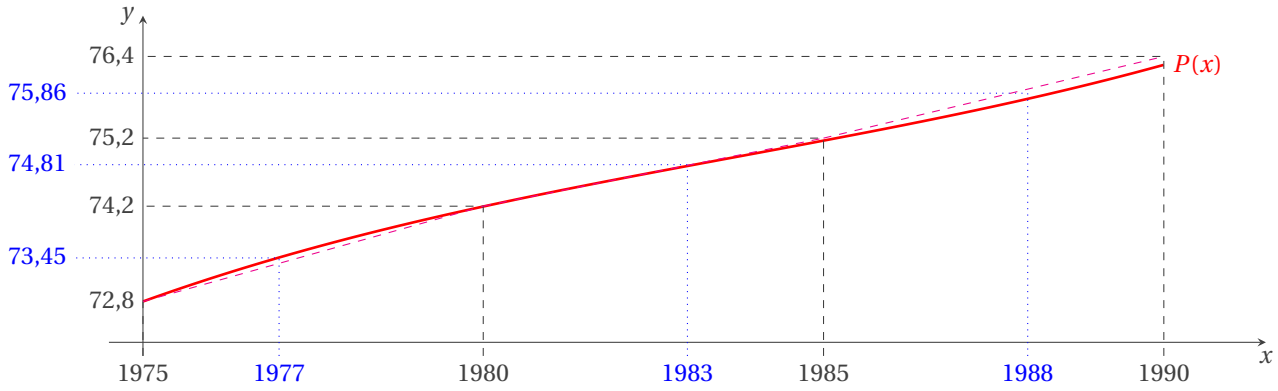
$$\begin{aligned} P(x) &= y_0 \frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} + y_1 \frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)} \\ &\quad + y_2 \frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} + y_3 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)} = \\ &= 72{,}8 \frac{(x-5)(x-10)(x-15)}{(0-5)(0-10)(0-15)} + 74{,}2 \frac{(x-0)(x-10)(x-15)}{(5-0)(5-10)(5-15)} \\ &\quad + 75{,}2 \frac{(x-0)(x-5)(x-15)}{(10-0)(10-5)(10-15)} + 76{,}4 \frac{(x-0)(x-5)(x-10)}{(15-0)(15-5)(15-10)} = \\ &= \frac{-72{,}8(x-5)(x-10)(x-15) + 3\times 74{,}2x(x-10)(x-15) - 3\times 75{,}2x(x-5)(x-15) + 76{,}4x(x-5)(x-10)}{750} \end{aligned}$$

On a alors que

- l'espérance de vie en 1977 correspond à $P(2) = 73{,}45$,
- l'espérance de vie en 1983 correspond à $P(8) = 74{,}81$,
- l'espérance de vie en 1988 correspond à $P(13) = 75{,}86$.

Si on considère une interpolation linéaire par morceaux (splines de degré 1) ; on obtient que l'espérance de vie est sous-estimé en 1977 et sur-estimé en 1988 par rapport à l'interpolation précédente car

- l'espérance de vie en 1977 correspond à $\frac{74{,}2-72{,}8}{5-0} 2 + 72{,}8 = 73{,}36 < P(2)$,
- l'espérance de vie en 1983 correspond à $\frac{75{,}2-74{,}2}{10-5} 8 + 73{,}2 = 74{,}8 \sim P(8)$,
- l'espérance de vie en 1988 correspond à $\frac{76{,}4-74{,}2}{15-10} 13 + 72{,}8 = 75{,}92 > P(13)$.



Exercice 2.17

Pour calculer le zéro d'une fonction $y = f(x)$ inversible sur un intervalle $[a;b]$ on peut utiliser l'interpolation : après avoir évalué f sur une discrétisation x_i de $[a;b]$, on interpole l'ensemble $\{(y_i, x_i)\}_{i=0}^{n}$ et on obtient un polynôme $x = p(y)$ tel que

$$f(x) = 0 \qquad \Longleftrightarrow \qquad x = p(0).$$

Utiliser cette méthode pour évaluer l'unique racine α de la fonction $f(x) = e^x - 2$ dans l'intervalle $[0;1]$ avec trois points d'interpolation.
Comparer ensuite le résultat obtenu avec l'approximation du zéro de f obtenue par la méthode de Newton en 3 itérations à partir de $x_0 = 0$.

CORRECTION DE L'EXERCICE 2.17. Calculons d'abord les valeurs à interpoler

i	x_i	y_i
0	0	-1
1	$\frac{1}{2}$	$\sqrt{e}-2$
2	1	$e-2$

Le polynôme d'interpolation de LAGRANGE de degré n sur l'ensemble des $n+1$ points $\{(y_i, x_i)\}_{i=0}^n$ s'écrit

$$p_n(y) = \sum_{i=0}^{n} \left(x_i \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{y-y_j}{y_i-y_j} \right).$$

Ici $n=2$ donc on a

$$\begin{aligned} p(y) &= x_0 \frac{(y-y_1)(y-y_2)}{(y_0-y_1)(y_0-y_2)} + x_1 \frac{(y-y_0)(y-y_2)}{(y_1-y_0)(y_1-y_2)} + x_2 \frac{(y-y_0)(y-y_1)}{(y_2-y_0)(y_2-y_1)} \\ &= \frac{1}{2} \frac{(y+1)(y-e+2)}{(\sqrt{e}-2+1)(\sqrt{e}-2-e+2)} + \frac{(y+1)(y-\sqrt{e}+2)}{(e-2+1)(e-2-\sqrt{e}+2)}. \end{aligned}$$

Par conséquent une approximation de la racine de f est $p(0) = \frac{1}{2}\frac{-e+2}{(\sqrt{e}-2+1)(\sqrt{e}-2-e+2)} + \frac{-\sqrt{e}+2}{(e-2+1)(e-2-\sqrt{e}+2)} \approx 0.7087486785$.

La méthode de Newton s'écrit

$$\begin{cases} x_0 = 0, \\ x_{k+1} = x_k - \frac{e^{x_k}-2}{e^{x_k}} = x_k - 1 + \frac{2}{e^{x_k}}, \end{cases}$$

on obtient ainsi la suite

k	x_k
0	0
1	1
2	$\frac{2}{e} \approx 0.7357588825$
3	$\frac{2-e}{e} - \frac{2}{e^{\frac{2}{e}}} \approx 0.6940422999$

Remarque : comme il n'y a que trois points d'interpolation, on pourrait calculer directement le polynôme interpolateur de f plutôt que de sa fonction réciproque et chercher les zéros de ce polynôme directement car il s'agit d'un polynôme de degré 2. Cependant cette idée ne peut pas être généralisée au cas de plus de trois points d'interpolation car on ne connait pas de formule générale pour le calcul des zéros d'un polynôme de degré $n \geq 3$.

Exercice 2.18

Soit f une fonction continue dont on connait les valeurs uniquement pour t entier, c'est-à-dire on suppose connues les valeurs $f(\kappa)$ pour tout $\kappa \in \mathbb{Z}$. Si $t \in \mathbb{R} \setminus \mathbb{Z}$, on définit une approximation $p(t)$ de $f(t)$ en interpolant la fonction f par un polynôme de degré 3 aux quatre points entiers les plus proches de t. Calculer $p(t)$ et écrire un algorithme qui fournit $p(t)$.

CORRECTION DE L'EXERCICE 2.18. Soit $\ell = E[t]$ la partie entière[2] de t. Alors $t \in [\ell; \ell+1]$ et il s'agit de définir le polynôme p interpolant les points

$$(\kappa-1, f(\kappa-1)), \qquad (\kappa, f(\kappa)), \qquad (\kappa+1, f(\kappa+1)), \qquad (\kappa+2, f(\kappa+2)),$$

ce qui donne

$$\begin{aligned} P(t) &= \sum_{i=0}^{3} \left(f(\kappa-1+i) \prod_{\substack{j=0 \\ j\neq i}}^{3} \frac{t-(\kappa-1+j)}{(\kappa-1+i)-(\kappa-1+j)} \right) = \sum_{i=0}^{3} \left(f(\kappa-1+i) \prod_{\substack{j=0 \\ j\neq i}}^{3} \frac{t-\kappa+1-j}{i-j} \right) \\ &= -\frac{f(\kappa-1)}{6}(t-\kappa)(t-\kappa-1)(t-\kappa-2) + \frac{f(\kappa)}{2}(t-\kappa+1)(t-\kappa-1)(t-\kappa-2) \\ &\quad - \frac{f(\kappa+1)}{2}(t-\kappa+1)(t-\kappa)(t-\kappa-2) + \frac{f(\kappa+2)}{6}(t-\kappa+1)(t-\kappa)(t-\kappa-1) \end{aligned}$$

Require: $f : \mathbb{Z} \to \mathbb{R}$, t
  $\kappa \leftarrow E[t]$
  $x_0 \leftarrow \kappa - 1$

2. Pour tout nombre réel x, la partie entière notée $E(x)$ est le plus grand entier relatif inférieur ou égal à x. Par exemple, $E(2.3) = 2$, $E(-2) = -2$ et $E(-2.3) = -3$. La fonction partie entière est aussi notée $[x]$ (ou $\lfloor x \rfloor$ par les anglo-saxons). On a toujours $E(x) \leq x < E(x)+1$ avec égalité si et seulement si x est un entier relatif. Pour tout entier relatif k et et pour tout nombre réel x, on a $E(x+k) = E(x)+k$. L'arrondi à l'entier le plus proche d'un réel x peut être exprimé par $E(x+0.5)$.

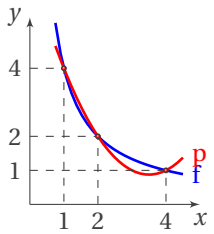
$x_1 \leftarrow \kappa$
$x_2 \leftarrow \kappa + 1$
$x_3 \leftarrow \kappa + 2$
$y \leftarrow 0$
for $i = 0$ to 3 **do**
  $L \leftarrow 1$
  for $j = 0$ to 3 **do**
    if $j \neq i$ **then**
      $L \leftarrow \dfrac{t - x_j}{x_i - x_j} \times L$
    end if
  end for
  $y \leftarrow y + f(x_i) \times L$
end for
return y

Exercice 2.19

1. Calculer le polynôme p de LAGRANGE qui interpole la fonction $f(x) = \frac{4}{x}$ aux points d'abscisse $x_0 = 1$, $x_1 = 2$ et $x_2 = 4$. Esquisser les graphes de f et de p pour $x \in [1,4]$.
2. Vérifier que l'erreur $\varepsilon(x) \equiv f(x) - p(x)$ prend sa valeur maximale en un unique point $\tilde{x}$ dans l'intervalle $[2,4]$. Calculer ensuite $\tilde{x}$ à 10^{-1} près (on pourra utiliser la méthode de dichotomie).
3. Comparer la fonction ε avec l'estimation théorique de l'erreur.

CORRECTION DE L'EXERCICE 2.19.

1. f est une hyperbole et p est la parabole qui passe par les points $(1,4)$, $(2,2)$ et $(4,1)$: $p(x) = \frac{1}{2}x^2 - \frac{7}{2}x + 7$



2. On a $\varepsilon(x) \equiv f(x) - p(x) = \frac{4}{x} - 7 + \frac{7}{2}x - \frac{1}{2}x^2$. Comme $\varepsilon'(x) = \frac{7}{2} - x - \frac{4}{x^2}$, il s'agit de trouver $\tilde{x}$ tel que $\varepsilon'(x) = 0$. Une simple comparaison des graphes des fonctions $u\colon x \mapsto \frac{7}{2} - x$ et $v\colon x \mapsto \frac{4}{x^2}$ montre que $\varepsilon'(x) = 0$ admet une solution dans l'intervalle $[1,2]$ et une solution dans l'intervalle $[2,4]$ (en effet, $\varepsilon'(1) = u(1) - v(1) = 2.5 - 4 < 0$, $\varepsilon'(2) = u(2) - v(2) = 1.5 - 1 > 0$ et $\varepsilon'(4) = u(4) - v(4) < 0$). On a $\varepsilon''(x) = -1 + 8/x^3$: l'erreur étant convexe pour $x < 2$ et concave pour $x > 2$, on conclut qu'elle prend sa valeur maximale pour $x = \tilde{x} \in [2,4]$. On cherche alors $\tilde{x} \in [2,4]$ tel que $\varepsilon'(\tilde{x}) = 0$ par la méthode de dichotomie. Pour que l'erreur soit inférieur à 10^{-1}, il faut $E\left(\log_2\left(\frac{4-2}{10^{-1}}\right)\right) + 1 = E\left(2\log_2(2) + \log_2(5)\right) + 1 = 5$ étapes :

k	0	1	2	3	4	5
$[a_k, b_k]$	$[2,4]$	$[3,4]$	$[3, \frac{7}{2}]$	$[3, \frac{13}{4}]$	$[3, \frac{25}{8}]$	$[\frac{49}{16}, \frac{25}{8}]$
ℓ_k	3	$\frac{7}{2}$	$\frac{13}{4}$	$\frac{25}{8}$	$\frac{49}{16}$	$\frac{99}{32}$
$b_k - a_k$	$2 > 10^{-1}$	$1 > 10^{-1}$	$0.5 > 10^{-1}$	$0.25 > 10^{-1}$	$0.125 > 10^{-1}$	$0.0625 < 10^{-1}$

L'erreur prend sa valeur maximale pour $\tilde{x} \approx \frac{99}{32} = 3.09375$ et vaut $\varepsilon(\tilde{x}) \approx 0.01166653913$.

3. Comparons ce résultat avec l'estimation théorique de l'erreur : $n = 2$ et f est de classe $\mathscr{C}^\infty([1,4])$, donc pour tout $x \in [1,2]$ il existe $\xi_x \in [1,4]$ tel que

$$\varepsilon^{\text{théorique}}(x) = \frac{f'''(\xi_x)}{3!}(x-1)(x-2)(x-4) = -\frac{3}{\xi_x^4}(x^3 - 7x^2 + 14x - 8).$$

Comme $\varepsilon(x) = \frac{4}{x} - 7 + \frac{7}{2}x - \frac{1}{2}x^2$, on obtient $\varepsilon^{\text{théorique}} = \varepsilon$ ssi $\xi_x = \sqrt[4]{6x}$.

3. Quadrature

Calculer $\int_a^b f(x)\,\mathrm{d}x$ où f est une fonction donnée

Dans ce chapitre on va étudier des méthodes pour approcher les intégrales de fonctions. On sait bien qu'il n'est pas toujours possible, pour une fonction arbitraire, de trouver la forme explicite d'une primitive. Par exemple, comment peut-on tracer le graphe de la fonction erf (appelée fonction d'erreur de GAUSS) définie comme suit ?

$$\begin{aligned}\mathrm{erf}\colon \mathbb{R} &\to \mathbb{R}\\ x &\mapsto \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}\,\mathrm{d}t\end{aligned}$$

Mais même quand on la connaît, il est parfois difficile de l'utiliser. C'est par exemple le cas de la fonction $f(x)=\cos(4x)\cos(3\sin(x))$ pour laquelle on a

$$\int_0^\pi f(x)\,\mathrm{d}x = \pi\frac{81}{16}\sum_{k=0}^{\infty}\frac{(-9/4)^k}{k!(k+4)!};$$

on voit que le calcul de l'intégrale est transformé en un calcul, aussi difficile, de la somme d'une série. Dans certains cas, la fonction à intégrer n'est connue que par les valeurs qu'elle prend sur un ensemble fini de points (par exemple, des mesures expérimentales). On se trouve alors dans la même situation que celle abordée au chapitre précédent pour l'approximation des fonctions. Dans tous ces cas, il faut considérer des méthodes numériques afin d'approcher la quantité à laquelle on s'intéresse, indépendamment de la difficulté à intégrer la fonction.

Dans les méthodes d'intégration, l'intégrale d'une fonction f continue sur un intervalle borné $[a,b]$ est remplacée par une somme finie. Le choix de la subdivision de l'intervalle d'intégration et celui des coefficients qui interviennent dans la somme approchant l'intégrale sont des critères essentiels pour minimiser l'erreur. Ces méthodes se répartissent en deux grandes catégories : les méthodes composées dans lesquelles la fonction est remplacée par un polynôme d'interpolation sur chaque intervalle élémentaire $[x_i,x_{i+1}]$ de la subdivision de $[a,b]$ (*i.e.* $[a,b]=\bigcup_i[x_i,x_{i+1}]$) et les méthodes de GAUSS fondées sur les polynômes orthogonaux pour lesquelles les points de la subdivision sont imposés.

3.1. Principes généraux

Soit f une fonction réelle intégrable sur l'intervalle $[a;b]$. Le calcul explicite de l'intégrale définie $I_{[a;b]}(f)\equiv\int_a^b f(x)\mathrm{d}x$ peut être difficile, voire impossible. On appelle *formule de quadrature* ou *formule d'intégration numérique* toute formule permettant de calculer une approximation de $I_{[a;b]}(f)$. Une possibilité consiste à remplacer f par une approximation $\tilde{f}$ et calculer $I_{[a;b]}(\tilde{f})$ au lieu de $I_{[a;b]}(f)$. En posant $\tilde{I}_{[a;b]}(f)\equiv I_{[a;b]}(\tilde{f})$, on définit

$$\tilde{I}_{[a;b]}(f)\equiv\int_a^b\tilde{f}(x)\mathrm{d}x.$$

Si f est de classe $\mathscr{C}^0$ sur $[a;b]$, l'erreur de quadrature $E_{[a;b]}(f)\equiv|\tilde{I}_{[a;b]}(f)-I_{[a;b]}(f)|$ satisfait

$$E_{[a;b]}(f)=\left|\int_a^b f(x)-\tilde{f}(x)\mathrm{d}x\right|\le\int_a^b|f(x)-\tilde{f}(x)|\mathrm{d}x\le(b-a)\max_{x\in[a;b]}|f(x)-\tilde{f}(x)|.$$

L'approximation $\tilde{f}$ doit être facilement intégrable, ce qui est le cas si, par exemple, $\tilde{f}$ est un polynôme.

Une approche naturelle consiste à prendre $\tilde{f}=\sum_{i=0}^n f(x_i)L_i(x)$, le polynôme d'interpolation de LAGRANGE de f sur un ensemble de $n+1$ nœuds distincts $\{x_i\}_{i=0}^{i=n}$. Ainsi on aura

$$I_{[a;b]}(f)\approx\tilde{I}_{[a;b]}(f)=\sum_{i=0}^n\left(f(x_i)\int_a^b L_i(x)\mathrm{d}x\right)\qquad\text{où}\qquad L_i(x)=\prod_{\substack{j=0\\ j\neq i}}^n\frac{x-x_j}{x_i-x_j}.$$

Il s'agit d'un cas particulier de la formule de quadrature suivante

$$\tilde{I}_{[a;b]}(f) = \sum_{i=0}^{n} \alpha_i f(x_i)$$

qui est une somme pondérée des valeurs de f aux points x_i : on dit que ces points sont les nœuds de la formule de quadrature et que les nombres $\alpha_i \in \mathbb{R}$ sont les coefficients ou encore les poids.

La formule de quadrature de LAGRANGE peut être généralisée au cas où on connaît les valeurs de la dérivée de f : ceci conduit à la formule de quadrature d'HERMITE. Les formules de LAGRANGE et d'HERMITE sont toutes les deux des *formules de quadrature interpolatoires*, car la fonction f est remplacée par son polynôme d'interpolation.

Définition *Degré d'exactitude*

On définit le degré de précision (ou d'exactitude) d'une formule de quadrature comme le plus grand entier $r \geq 0$ pour lequel la valeur approchée de l'intégrale (obtenue avec la formule de quadrature) d'un polynôme de degré r est égale à la valeur exacte, *i.e.* $\tilde{I}_{[a;b]}(q) = I_{[a;b]}(q)$ pour tout polynôme $q \in \mathbb{R}_r[x]$. Autrement dit, une formule de quadrature est dite d'ordre r si elle est exacte sur $\mathbb{R}_r[x]$ et inexacte pour au moins un polynôme de degré strictement supérieur à r.

Astuce

Pour vérifier qu'une formule de quadrature est d'ordre r il suffit de vérifier qu'elle est exacte sur une base de $\mathbb{R}_r[x]$ (par exemple la base canonique) et inexacte pour un polynôme de degré $r+1$ (par exemple le polynôme x^{r+1}). En effet, si q est un polynôme de $\mathbb{R}_r[x]$, il existe $\alpha_0, \alpha_1, \ldots, \alpha_r$ tels que $q(x) = \sum_{k=0}^{r} \alpha_k x^k$. Alors

$$I_{[a;b]}(q) = \int_a^b q(x)\,\mathrm{d}x = \sum_{k=0}^{r} \left(\alpha_k \left(\int_a^b x^k\,\mathrm{d}x \right) \right) = \sum_{k=0}^{r} \left(\alpha_k I_{[a;b]}(x^k) \right).$$

Pour vérifier qu'une formule de quadrature $\tilde{I}_{[a;b]}$ a degré de précision r il suffit alors de vérifier que $\tilde{I}_{[a;b]}(x^k) = I_{[a;b]}(x^k)$ pour tout $k = 0 \ldots r$.

Théorème

Toute formule de quadrature interpolatoire utilisant $n+1$ nœuds distincts a un degré de précision au moins égale à n.
En effet, si $f \in \mathbb{R}_n[x]$, alors le polynôme d'interpolation coïncide avec f.

La réciproque aussi est vraie : une formule de quadrature utilisant $n+1$ nœuds distincts et ayant un degré de précision au moins égale à n est nécessairement de type interpolatoire. Le degré de précision peut même atteindre $2n+1$ dans le cas des formules de quadrature de GAUSS.

Définition *Stabilité*

Une formule de quadrature $\tilde{I}_{[a;b]}(f) = \sum_{i=0}^{n} \alpha_i f(x_i)$ est dite stable s'il existe $M \in \mathbb{R}_+^*$ tel que $\sum_{i=0}^{n} |\alpha_i| \leq M$.

Théorème

Une méthode de quadrature de type interpolation est convergente sur $\mathscr{C}[a;b]$ ssi les formules sont stables.

Définition *Formule de quadrature composite*

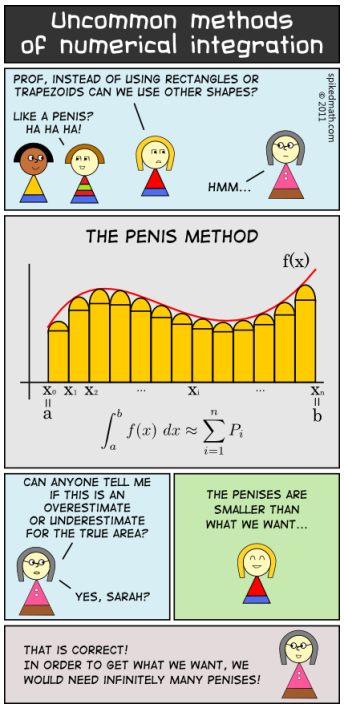
On décompose l'intervalle d'intégration $[a;b]$ en m sous-intervalles $[y_j; y_{j+1}]$ tels que $y_j = a + jH$ où $H = \frac{b-a}{m}$ pour $j = 0, 1, \ldots, m$. On utilise alors sur chaque sous-intervalle une formule interpolatoire de nœuds $\left\{ x_k^{(j)} \right\}_{k=0}^{n}$ et de poids $\left\{ \alpha_k^{(j)} \right\}_{k=0}^{n}$ (généralement la même formule sur chaque sous-intervalle). Puisque

$$I_{[a;b]}(f) = \int_a^b f(x)\mathrm{d}x = \sum_{j=0}^{m-1} \int_{y_j}^{y_{j+1}} f(x)\mathrm{d}x = \sum_{j=0}^{m-1} I_{[y_j;y_{j+1}]}(f),$$

une formule de quadrature interpolatoire composite est obtenue en remplaçant $I_{[a;b]}(f)$ par

$$\sum_{j=0}^{m-1} \tilde{I}^{(j)}_{[y_j;y_{j+1}]}(f) = \sum_{j=0}^{m-1} \sum_{k=0}^{n} \alpha_k^{(j)} f(x_k^{(j)})$$

où $I_{[y_j;y_{j+1}]}(f) \simeq \tilde{I}^{(j)}_{[y_j;y_{j+1}]}(f)$.



Souvent on définit d'abord une formule de quadrature sur l'intervalle $[0;1]$ ou sur l'intervalle $[-1;1]$ et puis on la généralise à l'intervalle $[x_i;x_{i+1}]$ par un changement de variable affine.

Astuce *Changement de variable affine*

Soit $x \in [a;b]$ et soit $y \in [c;d]$. On considère une fonction de classe $\mathscr{C}^1([a;b])$

$$\begin{aligned} g\colon [a;b] &\to [c;d] \\ x &\mapsto y = g(x) \end{aligned}$$

qui envoie l'intervalle $[a;b]$ dans l'intervalle $[c;d]$, c'est-à-dire telle que

$$\begin{cases} g(a) = c, \\ g(c) = d. \end{cases}$$

On a alors

$$\int_c^d f(y)\mathrm{d}y = \int_a^b f(g(x))g'(x)\,\mathrm{d}x.$$

Si $g'(x)$ est une constante, *i.e.* si g est une transformation affine $g(x) = mx + q$, alors

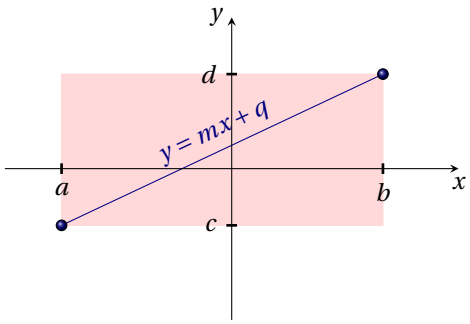
$$\int_c^d f(y)\mathrm{d}y = m\int_a^b f(mx+q)\mathrm{d}x.$$

Par conséquent, si $\int_a^b f(t)\,\mathrm{d}t \approx \sum_{i=0}^n \alpha_i f(t_i)$ alors $\int_a^b f(mx+q)\,\mathrm{d}x \approx \sum_{i=0}^n \alpha_i f(mx_i+q)$.
Pour déterminer cette transformation affine, on doit résoudre le système linéaire

$$\begin{cases} g(a) = c, \\ g(c) = d, \end{cases} \quad \textit{i.e.} \quad \begin{cases} ma + q = c, \\ mb + q = d. \end{cases}$$

On obtient

$$m = \frac{d-c}{b-a}, \qquad q = \frac{cb-ad}{b-a}.$$



Par conséquent $y = \dfrac{d-c}{b-a}x + \dfrac{cb-ad}{b-a}$ d'où

$$\int_c^d f(y)\mathrm{d}y = \frac{d-c}{b-a}\int_a^b f\left(\frac{d-c}{b-a}x + \frac{cb-ad}{b-a}\right)\mathrm{d}x.$$

Exemple

⋆ Transformer l'intervalle $[0;1]$ dans l'intervalle $[x_i;x_{i+1}]$ par un changement de variable affine.
On a $y = (x_{i+1}-x_i)x + x_i$ et

$$\int_{x_i}^{x_{i+1}} f(y)\mathrm{d}y = (x_{i+1}-x_i)\int_0^1 f((x_{i+1}-x_i)x + x_i)\mathrm{d}x.$$

⋆ Transformer l'intervalle $[-1;1]$ dans l'intervalle $[x_i;x_{i+1}]$ par un changement de variable affine.
On a $y = \frac{x_{i+1}-x_i}{2}x + \frac{x_{i+1}+x_i}{2}$, qu'on peut réécrire $y = x_i + (1+x)\frac{x_{i+1}-x_i}{2}$ et

$$\int_{x_i}^{x_{i+1}} f(y)\mathrm{d}y = \frac{x_{i+1}-x_i}{2}\int_{-1}^1 f\left(x_i + (1+x)\frac{x_{i+1}-x_i}{2}\right)\mathrm{d}x.$$

3.2. Exemples de formules de quadrature interpolatoires

Définition *Formule du* rectangle à gauche

- **Formule de base** La formule du *rectangle à gauche* est obtenue en remplaçant f par une constante égale à la valeur de f en la borne gauche de l'intervalle $[a;b]$ (polynôme qui interpole f en le point $(a, f(a))$ et donc de degré 0), ce qui donne

$$I_{[a;b]}(f) \approx I_{[a;b]}(\tilde{f}) = I_{[a;b]}(f(a)) = \int_a^b f(a)\,\mathrm{d}x = (b-a)f(a).$$

Erreur Si $f \in \mathscr{C}^1([a;b])$ alors il existe $\eta \in]a;b[$ tel que $f(x) = f(a) + (x-a)f'(\eta)$ et donc l'erreur de quadrature est majorée par

$$\begin{aligned} E_{[a;b]}(f) &= \left|\int_a^b f(x) - \tilde{f}(x)\,\mathrm{d}x\right| = \left|\int_a^b f(x) - f(b)\,\mathrm{d}x\right| \\ &= \left|\int_a^b (x-a)f'(\eta)\,\mathrm{d}x\right| = \frac{(b-a)^2}{2}|f'(\eta)| \le \frac{(b-a)^2}{2}\max_{[a;b]}|f'(x)|. \end{aligned}$$

Degré Le degré de précision de la formule du rectangle à gauche est 0.

- **Formule composite** On décompose l'intervalle d'intégration $[a;b]$ en m sous-intervalles de largeur $H = \frac{b-a}{m}$ avec $m \ge 1$. En introduisant les nœuds de quadrature $x_k = a + kH$ pour $k = 0, 1, \ldots, m-1$ on obtient la *formule composite du rectangle à gauche*

$$I^m_{[a;b]}(\tilde{f}) = \sum_{k=0}^{m-1} I_{[x_k;x_{k+1}]}(\tilde{f}) = H\sum_{k=0}^{m-1} f(x_k) = H\sum_{k=0}^{m-1} f(a+kH).$$

Erreur Si $f \in \mathscr{C}^1([a;b])$ alors l'erreur de quadrature est majorée par

$$E^m_{[a;b]}(f) = \sum_{k=0}^{m-1} E_{[x_k;x_{k+1}]}(f) \le m\frac{H^2}{2}\max_{[a;b]}|f'(x)| = \frac{b-a}{2}H\max_{[a;b]}|f'(x)|.$$

Définition *Formule du* rectangle à droite

- **Formule de base** La formule du *rectangle à droite* est obtenue en remplaçant f par une constante égale à la valeur de f en la borne droite de l'intervalle $[a;b]$ (polynôme qui interpole f en le point $(b, f(b))$ et donc de degré 0), ce qui donne

$$I_{[a;b]}(f) \approx I_{[a;b]}(\tilde{f}) = I_{[a;b]}(f(b)) = \int_a^b f(b)\,\mathrm{d}x = (b-a)f(b).$$

  Erreur Si $f \in \mathscr{C}^1([a;b])$ alors il existe $\eta \in]a;b[$ tel que $f(x) = f(b) + (x-b)f'(\eta)$ et donc l'erreur de quadrature est majorée par

$$\begin{aligned} E_{[a;b]}(f) &= \left| \int_a^b f(x) - \tilde{f}(x)\,\mathrm{d}x \right| = \left| \int_a^b f(x) - f(b)\,\mathrm{d}x \right| \\ &= \left| \int_a^b (x-b)f'(\eta)\,\mathrm{d}x \right| = \frac{(b-a)^2}{2}|f'(\eta)| \leq \frac{(b-a)^2}{2} \max_{[a;b]} |f'(x)|. \end{aligned}$$

  Degré Le degré de précision de la formule du rectangle à droite est 0.

- **Formule composite** On décompose maintenant l'intervalle d'intégration $[a;b]$ en m sous-intervalles de largeur $H = \frac{b-a}{m}$ avec $m \geq 1$. En introduisant les nœuds de quadrature $x_k = a + (k+1)H$ pour $k = 0, 1, \ldots, m-1$ on obtient la *formule composite du rectangle à droite*

$$I^m_{[a;b]}(\tilde{f}) = \sum_{k=0}^{m-1} I_{[x_k;x_{k+1}]}(\tilde{f}) = H \sum_{k=0}^{m-1} f(x_{k+1}) = H \sum_{k=0}^{m-1} f(a + (k+1)H).$$

  Erreur Si $f \in \mathscr{C}^1([a;b])$ alors l'erreur de quadrature est majorée par

$$E^m_{[a;b]}(f) = \sum_{k=0}^{m-1} E_{[x_k;x_{k+1}]}(f) \leq m\frac{H^2}{2} \max_{[a;b]} |f'(x)| = \frac{b-a}{2} H \max_{[a;b]} |f'(x)|.$$

Définition *Formule du* rectangle *ou du* point milieu

- **Formule de base** La formule du *rectangle* ou du *point milieu* est obtenue en remplaçant f par une constante égale à la valeur de f au milieu de $[a;b]$ (polynôme qui interpole f en le point $\left(\frac{a+b}{2}, f\left(\frac{a+b}{2}\right)\right)$ et donc de degré 0), ce qui donne

$$I_{[a;b]}(f) \approx I_{[a;b]}(\tilde{f}) = I_{[a;b]}\left(f\left(\tfrac{a+b}{2}\right)\right) = \int_a^b f\left(\tfrac{a+b}{2}\right)\mathrm{d}x = (b-a)f\left(\tfrac{a+b}{2}\right).$$

  Erreur Si $f \in \mathscr{C}^2([a;b])$ alors il existe $\eta \in]a;b[$ tel que $f(x) = f\left(\frac{a+b}{2}\right) + \left(x - \frac{a+b}{2}\right) f'\left(\frac{a+b}{2}\right) + \frac{1}{2}\left(x - \frac{a+b}{2}\right) f''(\eta)$ et donc l'erreur de quadrature est majorée par

$$\begin{aligned} E_{[a;b]}(f) &= \left| \int_a^b f(x) - \tilde{f}(x)\,\mathrm{d}x \right| \\ &= \left| \int_a^b f(x) - f\left(\tfrac{a+b}{2}\right)\mathrm{d}x \right| = \left| \int_a^b \left(x - \tfrac{a+b}{2}\right) f'\left(\tfrac{a+b}{2}\right) + \tfrac{1}{2}\left(x - \tfrac{a+b}{2}\right)^2 f''(\eta)\,\mathrm{d}x \right| \\ &= \frac{1}{2}\left|f''(\eta)\right| \int_a^b \left(x - \tfrac{a+b}{2}\right)^2 \mathrm{d}x = \frac{1}{2}\left|f''(\eta)\right| \frac{(b-a)^3}{12} \leq \frac{(b-a)^3}{24} \max_{[a;b]} |f''(x)|. \end{aligned}$$

  Degré Le degré de précision de la formule du point milieu est 1.

- **Formule composite** On décompose maintenant l'intervalle d'intégration $[a;b]$ en m sous-intervalles de largeur $H = \frac{b-a}{m}$ avec $m \geq 1$. En introduisant les nœuds de quadrature $x_k = a + k\frac{H}{2}$ pour $k = 0, 1, \ldots, 2m$ (*i.e.* chaque sous-intervalle $[x_{2k}; x_{2k+2}]$ a largeur H et donc x_{2k+1} est sont point milieu), on obtient la *formule composite du point milieu*

$$I^m_{[a;b]}(\tilde{f}) = \sum_{k=0}^{m-1} I_{[x_{2k};x_{2k+2}]}(\tilde{f}) = \sum_{k=0}^{m-1} (x_{2k+2} - x_{2k}) f(x_{2k+1}) = H \sum_{k=0}^{m-1} f\left(a + (2k+1)\tfrac{H}{2}\right).$$

  Erreur Si $f \in \mathscr{C}^2([a;b])$ alors l'erreur de quadrature est majorée par

$$E^m_{[a;b]}(f) = \sum_{k=0}^{m-1} E_{[x_{2k};x_{2k+2}]}(f) \leq m\frac{H^3}{24} \max_{[a;b]} |f''(x)| = \frac{b-a}{24} H^2 \max_{[a;b]} |f''(x)|.$$

Définition *Formule du* trapèze

- **Formule de base** La formule du *trapèze* est obtenue en remplaçant f par le segment qui relie $(a,f(a))$ à $(b,f(b))$ (polynôme qui interpole f en les points $(a,f(a))$ et $(b,f(b))$ et donc de degré 1), ce qui donne

$$I_{[a;b]}(f) \approx I_{[a;b]}(\tilde{f}) = I_{[a;b]}\left(\tfrac{f(b)-f(a)}{b-a}(x-a)+f(a)\right) = \int_a^b \tfrac{f(b)-f(a)}{b-a}(x-a)+f(a)\,\mathrm{d}x = \frac{b-a}{2}\big(f(a)+f(b)\big).$$

Erreur Si $f \in \mathscr{C}^2([a;b])$ alors il existe $\eta \in]a;b[$ tel que $f(x)-\tilde{f}(x) = \frac{f''(\eta)}{2}\omega_2(x) = \frac{f''(\eta)}{2}(x-a)(x-b)$ et donc l'erreur de quadrature est majorée par

$$\begin{aligned} E_{[a;b]}(f) &= \left|\int_a^b f(x)-\tilde{f}(x)\,\mathrm{d}x\right| = \left|\int_a^b \tfrac{f''(\eta)}{2}(x-a)(x-b)\,\mathrm{d}x\right| \\ &\le \tfrac{1}{2}\left|f''(\eta)\right|\int_a^b (x-a)(x-b)\,\mathrm{d}x = \tfrac{1}{2}\left|f''(\eta)\right|\tfrac{(b-a)^3}{6} \le \frac{(b-a)^3}{12}\max_{[a;b]}|f''(x)|. \end{aligned}$$

Degré Le degré de précision de la formule du point milieu est 1, comme celle du point milieu.

- **Formule composite** Pour obtenir la *formule du trapèze composite*, on décompose l'intervalle d'intégration $[a;b]$ en m sous-intervalles de largeur $H = \frac{b-a}{m}$ avec $m \ge 1$. En introduisant les nœuds de quadrature $x_k = a+kH$ pour $k = 0,1,\dots,m-1$ on obtient la *formule composite des trapèzes*

$$\begin{aligned} I^m_{[a;b]}(\tilde{f}) &= \sum_{k=0}^{m-1} I_{[x_k;x_{k+1}]}(\tilde{f}) = \sum_{k=0}^{m-1}\frac{x_{k+1}-x_k}{2}\big(f(x_k)+f(x_{k+1})\big) \\ &= \frac{H}{2}\sum_{k=0}^{m-1}\big(f(x_k)+f(x_{k+1})\big) = H\left(\frac{1}{2}f(a)+\sum_{k=1}^{m-1}f(a+kH)+\frac{1}{2}f(b)\right). \end{aligned}$$

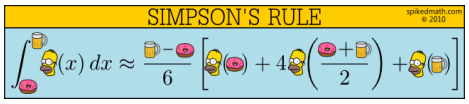
Erreur Si $f \in \mathscr{C}^2([a;b])$ alors l'erreur de quadrature est majorée par

$$E^m_{[a;b]}(f) = \sum_{k=0}^{m-1} E_{[x_k;x_{k+1}]}(f) \le m\frac{H^3}{12}\max_{[a;b]}|f''(x)| = \frac{b-a}{12}H^2\max_{[a;b]}|f''(x)|.$$

Définition *Formule de* Cavalieri-Simpson

- **Formule de base** La formule de *Cavalieri-Simpson* est obtenue en remplaçant f par la parabole qui interpole f en a, en b et en $\frac{a+b}{2}$ (donc un polynôme de degré 2), ce qui donne

$$\begin{aligned} I_{[a;b]}(f) \approx I_{[a;b]}(\tilde{f}) &= I_{[a;b]}\left(\frac{\left(x-\frac{a+b}{2}\right)(x-b)}{\left(a-\frac{a+b}{2}\right)(a-b)}f(a)+\frac{(x-a)(x-b)}{\left(\frac{a+b}{2}-a\right)\left(\frac{a+b}{2}-b\right)}f\left(\tfrac{a+b}{2}\right)+\frac{(x-a)\left(x-\frac{a+b}{2}\right)}{(b-a)\left(b-\frac{a+b}{2}\right)}f(b)\right) \\ &= \int_a^b \frac{\left(x-\frac{a+b}{2}\right)(x-b)}{\left(a-\frac{a+b}{2}\right)(a-b)}f(a)+\frac{(x-a)(x-b)}{\left(\frac{a+b}{2}-a\right)\left(\frac{a+b}{2}-b\right)}f\left(\tfrac{a+b}{2}\right)+\frac{(x-a)\left(x-\frac{a+b}{2}\right)}{(b-a)\left(b-\frac{a+b}{2}\right)}f(b)\,\mathrm{d}x \\ &= \frac{b-a}{6}\left(f(a)+4f\left(\frac{a+b}{2}\right)+f(b)\right). \end{aligned}$$



Erreur Si $f \in \mathscr{C}^4([a;b])$ on peut démontrer que l'erreur de quadrature est majorée par

$$E_{[a;b]}(f) \le \frac{(b-a)^5}{2880}\max_{[a;b]}|f^{(IV)}(x)|.$$

Degré Le degré de précision de la formule du point milieu est 3.

- **Formule composite** On décompose maintenant l'intervalle d'intégration $[a;b]$ en m sous-intervalles de largeur $H = \frac{b-a}{m}$ avec $m \ge 1$. En introduisant les nœuds de quadrature $x_k = a+k\frac{H}{2}$ pour $k = 0,1,\dots,2m$ (*i.e.* chaque sous-intervalle $[x_{2k};x_{2k+2}]$ a largeur H et donc x_{2k+1} est sont point milieu), on obtient la *formule composite de Cavalieri-Simpson*

$$I^m_{[a;b]}(\tilde{f}) = \sum_{k=0}^{m-1} I_{[x_{2k};x_{2k+2}]}(\tilde{f}) = \sum_{k=0}^{m-1}\frac{x_{2k+2}-x_{2k}}{6}\big(f(x_{2k})+4f(x_{2k+1})+f(x_{2k+2})\big)$$

$$= \frac{H}{6}\sum_{k=0}^{m-1}\left(f(x_{2k})+4f(x_{2k+1})+f(x_{2k+2})\right)$$
$$= \frac{H}{6}\left(f(a)+f(b)+\sum_{k=1}^{m-1}f(x_{2k})+4\sum_{k=0}^{m-1}f(x_{2k+1})\right)$$
$$= \frac{H}{6}\left(f(a)+f(b)+\sum_{k=1}^{m-1}f(a+kH)+4\sum_{k=0}^{m-1}f\left(a+\left(k+\tfrac{1}{2}\right)H\right)\right).$$

Erreur Si $f\in\mathscr{C}^4([a;b])$ alors l'erreur de quadrature est majorée par

$$E^m_{[a;b]}(f)=\sum_{k=0}^{m-1}E_{[x_{2k};x_{2k+2}]}(f)\leq m\frac{H^5}{2880}\max_{[a;b]}|f''(x)|=\frac{b-a}{2880}H^4\max_{[a;b]}|f''(x)|.$$

Algorithmes

MÉTHODE DU RECTANGLE À GAUCHE

Require: $a\,;\,b>a\,;\,m>0\,;\,f:[a;b]\to\mathbb{R}$
  $H\leftarrow\frac{b-a}{m}$
  $s\leftarrow 0$
  for $k=0$ to $m-1$ **do**
    $s\leftarrow s+f(a+kH)$
  end for
  return $I\leftarrow Hs$

MÉTHODE DU RECTANGLE À DROITE

Require: $a\,;\,b>a\,;\,m>0\,;\,f:[a;b]\to\mathbb{R}$
  $H\leftarrow\frac{b-a}{m}$
  $s\leftarrow 0$
  for $k=0$ to $m-1$ **do**
    $s\leftarrow s+f(a+(k+1)H)$
  end for
  return $I\leftarrow Hs$

MÉTHODE DU POINT MILIEU

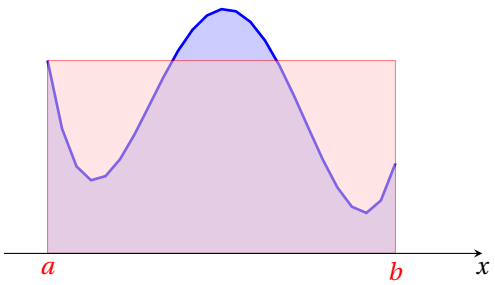
Require: $a\,;\,b>a\,;\,m>0\,;\,f:[a;b]\to\mathbb{R}$
  $H\leftarrow\frac{b-a}{m}$
  $s\leftarrow 0$
  for $k=0$ to $m-1$ **do**
    $s\leftarrow s+f\left(a+\left(k+\frac{1}{2}\right)H\right)$
  end for
  return $I\leftarrow Hs$

MÉTHODE DES TRAPÈZES

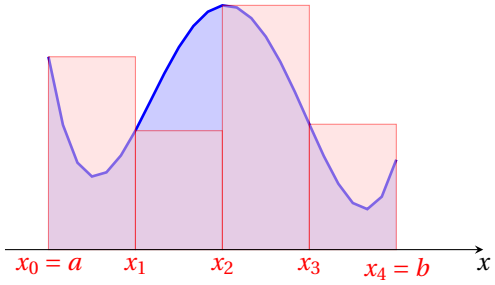
Require: $a\,;\,b>a\,;\,m>0\,;\,f:[a;b]\to\mathbb{R}$
  $H\leftarrow\frac{b-a}{m}$
  $s\leftarrow\frac{f(a)+f(b)}{2}$
  for $k=1$ to $m-1$ **do**
    $s\leftarrow s+f(a+kH)$
  end for
  return $I\leftarrow Hs$

MÉTHODE DE SIMPSON

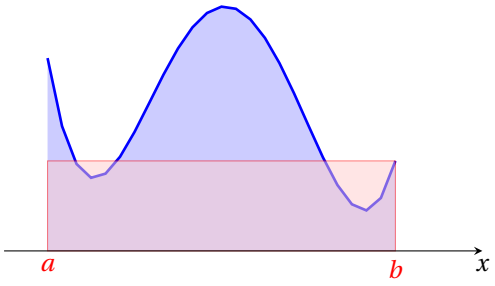
Require: $a\,;\,b>a\,;\,m>0\,;\,f:[a;b]\to\mathbb{R}$
  $H\leftarrow\frac{b-a}{m}$
  $s\leftarrow f(a)+f(b)+4f\left(a+\frac{H}{2}\right)$
  for $k=1$ to $m-1$ **do**
    $s\leftarrow s+f(a+kH)+f\left(a+\left(k+\frac{1}{2}\right)H\right)$
  end for
  return $I\leftarrow\frac{H}{6}s$



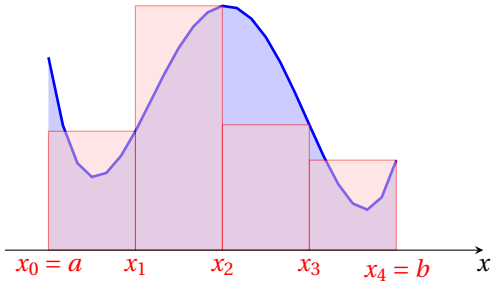
(a) Formule du rectangle à gauche.



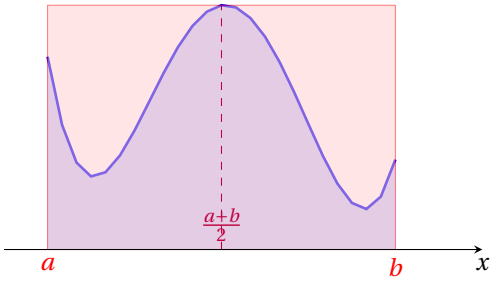
(b) Formule du rectangle à gauche composite.



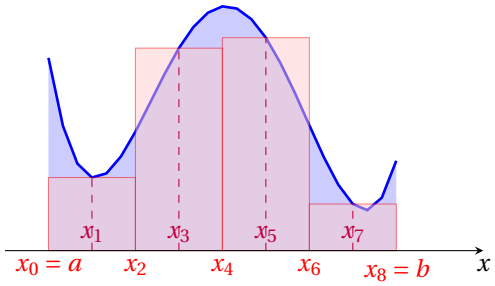
(c) Formule du rectangle à droite.



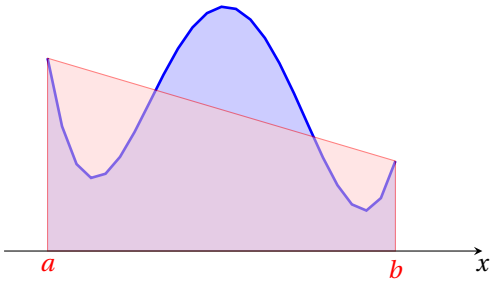
(d) Formule du rectangle à droite composite.



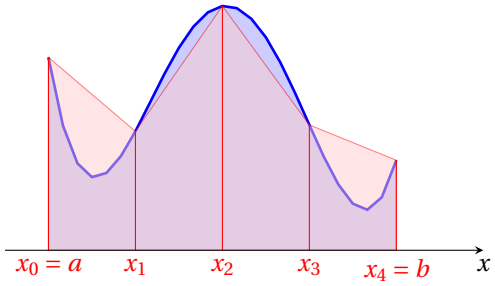
(e) Formule du point milieu.



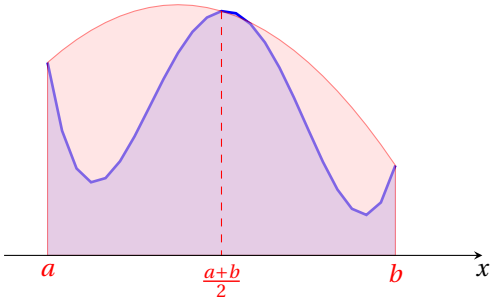
(f) Formule du point milieu composite.



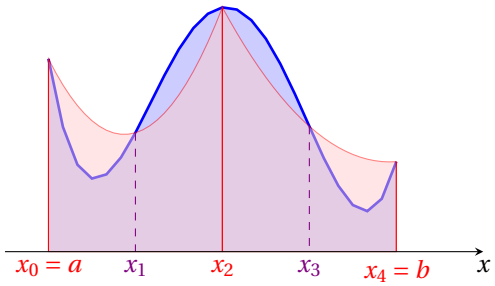
(g) Formule du trapèze.



(h) Formule du trapèze composite.



(i) Formule de Cavalieri-Simpson.



(j) Formule de Cavalieri-Simpson composite.

FIGURE 3.1.: Exemples de formules de quadrature.

3.3. Approximation de dérivées

Soit $f : \mathbb{R} \to \mathbb{R}$ une fonction de classe $\mathscr{C}^1(\mathbb{R})$, $x_i \in \mathbb{R}$ et f' sa dérivée. On sait que

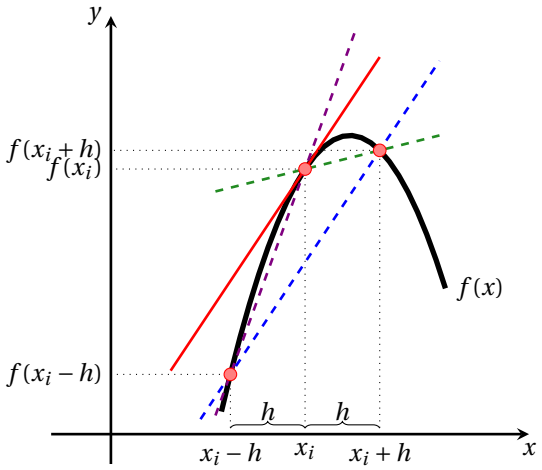
$$\begin{aligned} f'(x_i) &= \lim_{h\to 0} \frac{f(x_i+h)-f(x_i)}{h} \\ &= \lim_{h\to 0} \frac{f(x_i)-f(x_i-h)}{h} \\ &= \lim_{h\to 0} \frac{f(x_i+h/2)-f(x_i-h/2)}{h}. \end{aligned}$$

Une idée naturelle pour calculer numériquement $f'(x_i)$ consiste donc à se donner une valeur de h positive assez petite et à calculer

$$f'(x_i) \approx \delta_h^+ f(x_i) \equiv \frac{f(x_i+h)-f(x_i)}{h}, \tag{3.1}$$

$$f'(x_i) \approx \delta_h^- f(x_i) \equiv \frac{f(x_i)-f(x_i-h)}{h}, \tag{3.2}$$

$$f'(x_i) \approx \delta_h f(x_i) \equiv \frac{f(x_i+h)-f(x_i-h)}{2h}, \tag{3.3}$$



On les appelles *taux d'accroissement* ou *différences finies*

- *à droite (ou progressive)* δ_h^+,
- *à gauche (ou rétrograde)* δ_h^-,
- *centrée* δ_h.

Si f est de classe $\mathscr{C}^3$, en écrivant le développement de TAYLOR de f en x autour du point x_i

$$f(x_i \pm h) = f(x_i) \pm h f'(x_i) + h^2 f''(x_i) + O(h^3),$$

on obtient

$$\frac{f(x_i+h)-f(x_i)}{h} = \frac{f(x_i)+hf'(x_i)+h^2f''(x_i)+O((h)^3)-f(x_i)}{h} = f'(x_i)+O(h),$$

$$\frac{f(x_i)-f(x_i-h)}{h} = \frac{f(x_i)-f(x_i)+hf'(x_i)-h^2f''(x_i)+O(h^3)}{h} = f'(x_i)+O(h),$$

$$\frac{f(x_i+h)-f(x_i-h)}{2h} = \frac{f(x_i)+hf'(x_i)+(h)^2f''(x_i)+O(h^3)-f(x_i)+hf'(x_i)-(h)^2f''(x_i)}{2h} = f'(x_i)+O(h^2).$$

Donc, si f est assez régulière, les différences finies convergent vers $f'(x_i)$ lorsque h tend vers zéro. De plus, pour les différences finies à gauche et à droite la convergence est d'ordre 1 alors que la différence finie centrée converge à l'ordre 2.

Exemple

On compare pour différentes valeurs de h les valeurs données par ces trois formules pour la dérivée de la fonction sinus en 0 :

```
from math import *

def DFgauche(f,x,h):
    return (f(x+h)-f(x))/h

def DFdroite(f,x,h):
    return (f(x)-f(x-h))/h

def DFcentree(f,x,h):
    return (f(x+0.5*h)-f(x-0.5*h))/h

# TEST
def f(x):
    return sin(x)

x = 0
for i in range(1,13):
```

```
    h = 10**(-i)
    dfg = DFgauche(f,x,h)
    dfd = DFdroite(f,x,h)
    dfc = DFcentree(f,x,h)
    print "%5.e %17.15f %17.15f %17.15f" %(h, dfg, dfd, dfc)
```

On constate qu'à partir de $h = 10^{-8}$ la valeur donnée est exacte.

```
1e-01 0.998334166468282 0.998334166468282 0.999583385413567
1e-02 0.999983333416666 0.999983333416666 0.999995833338542
1e-03 0.999999833333342 0.999999833333342 0.999999958333334
1e-04 0.999999998333333 0.999999998333333 0.999999999583333
1e-05 0.999999999983333 0.999999999983333 0.999999999995833
1e-06 0.999999999999833 0.999999999999833 0.999999999999958
1e-07 0.999999999999998 0.999999999999998 1.000000000000000
1e-08 1.000000000000000 1.000000000000000 1.000000000000000
1e-09 1.000000000000000 1.000000000000000 1.000000000000000
1e-10 1.000000000000000 1.000000000000000 1.000000000000000
1e-11 1.000000000000000 1.000000000000000 1.000000000000000
1e-12 1.000000000000000 1.000000000000000 1.000000000000000
```

Définition *Erreur de troncature*

Les différences

$$|f'(x_i) - \delta_h^+|, \qquad |f'(x_i) - \delta_h^-|,$$

sont appelées *erreur de troncature*. Elles sont d'ordre h et on dit que les différences finies sont *consistantes* à l'ordre 1 en h. De même, l'erreur de troncature

$$|f'(x_i) - \delta_h|,$$

est d'ordre h^2 et on dit que la différence finie est consistante à l'ordre 2 en h. Elle est ainsi plus précise que les formules de différences finies progressives et rétrogrades.

Remarque *Erreurs d'arrondis*

Les erreurs de troncature diminuent lorsque h diminue. En revanche, les erreurs d'arrondis augmentent lorsque h diminue. En effet, le calcul de $\delta_h^\pm$ se fait avec une précision absolue de l'ordre de $2 \times \varepsilon \times |f(x)|/h$ où $\varepsilon \approx 10^{-15}$. Par ailleurs, d'après l'inégalité de Taylor-Lagrange, on a $|f'(x_i) - \delta_h^\pm| \leq \frac{h}{2} \max|f''(x)|$. L'inégalité triangulaire entraîne alors $|f'(x_i) - \mathrm{flt}(\delta_h^\pm)| \leq \frac{h}{2} \max|f''(x)| + 2\varepsilon \frac{|f(x)|}{h}$. Une étude rapide de la fonction $h \mapsto \frac{h}{2} \max|f''(x)| + 2\varepsilon \frac{|f(x)|}{h}$ montre que cette fonction possède un minimum absolu sur $\mathbb{R}_+$ atteint en $h = 2\sqrt{\varepsilon \frac{|f(x)|}{\max|f''(x)|}}$. Pour une fonction suffisamment régulière, il est donc judicieux de choisir une valeur de h qui soit de l'ordre de $\sqrt{\varepsilon}$, c'est-à-dire de l'ordre de 10^{-8}.

Exemple

Par exemple, en utilisant le code de l'exemple précédent pour calculer la dérivée première de la fonction $\sqrt{1+x}$ en 0, on obtient

```
1e-01 0.488088481701516 0.513167019494862 0.500156421150636
1e-02 0.498756211208895 0.501256289338003 0.500001562517094
1e-03 0.499875062460964 0.500125062539047 0.500000015625002
1e-04 0.499987500623966 0.500012500624925 0.500000000157597
1e-05 0.499998750003172 0.500001250003379 0.500000000014378
1e-06 0.499999875058776 0.500000124969979 0.499999999958867
1e-07 0.499999988079480 0.500000012504387 0.499999999181711
1e-08 0.499999996961265 0.500000008063495 0.499999996961265
1e-09 0.500000041370185 0.500000041370185 0.500000041370185
1e-10 0.500000041370185 0.500000041370185 0.500000041370185
1e-11 0.500000041370185 0.500000041370185 0.500000041370185
1e-12 0.500044450291171 0.500044450291171 0.500044450291171
1e-13 0.499600361081320 0.500710584105946 0.498490138056695
1e-14 0.488498130835069 0.499600361081320 0.499600361081320
```

Cette fois-ci on voit apparaître très nettement la perte de précision lorsque h est trop petit.

De manière analogue, la dérivée seconde peut être approchée par

$$f''(x_i) \approx \frac{f(x_i+h)-2f(x_i)+f(x_i-h)}{(h)^2}$$

et on a l'estimation d'erreur :

$$\frac{f(x_i+h)-2f(x_i)+f(x_i-h)}{h^2} = \frac{f(x_i)+hf'(x_i)+(h)^2f''(x_i)+O((h)^3)-2f(x_i)+f(x_i)-hf'(x_i)+(h)^2f''(x_i)}{h^2} = f''(x_i)+O(h^2).$$

Codes Python

Voici cinq `function` python qui renvoient la valeur approchée d'une intégrale par les méthodes (composites à n intervalles équirépartis) du rectangle à gauche, du rectangle à droite, du point de milieu, du trapèze et de SIMPSON. En paramètre elles reçoivent `f`, la fonction (mathématique) à intégrer, `a` et `b` sont les extrémités de l'intervalle d'intégration et `n` est le nombre de sous-intervalles de l'intervalle $[a,b]$ (chaque sous-intervalle a largeur $(b-a)/n$). Elles renvoient la valeur approchée de $\int_a^b f(x)\,\mathrm{d}x$.

Méthodes numériques.

```
#!/usr/bin/python
#-*- coding: Utf-8 -*-

import math, sys

def rectangle_gauche_composite(f,a,b,m):
    H = (b-a)/m
    s = 0.
    for k in range(m):
        s += f(a+k*H)
    return H*s

def rectangle_droite_composite(f,a,b,m):
    H = (b-a)/m
    s = 0.
    for k in range(m):
        s += f(a+(k+1)*H)
    return H*s

def milieu_composite(f,a,b,m):
    H = (b-a)/m
    s = 0.
    for k in range(m):
        s += f(a+(2*k+1)*H*0.5)
    return H*s

def trapeze_composite(f,a,b,m):
    H = (b-a)/m
    s = (f(a)+f(b))*0.5
    for k in range(1,m):
        s += f(a+k*H)
    return H*s

def simpson_composite(f,a,b,m):
    H = (b-a)/m
    s = f(a)+f(b)+f(a+H*0.5)
    for k in range(1,m):
        s += f(a+k*H)+f(a+(2*k+1)*H*0.5)
    return H*s/6.
```

et voici quelques exemples d'utilisation de ces méthodes

```
# CHOIX DU CAS TEST
exemple = 1

# DEFINITION DU CAS TEST
if exemple==1:
    n = 100
    a = 0.0
    b = 1.0
    def f(x):
        return x**3
    def primitive(x):
        return x**4/4.
elif exemple==2:
    n = 100
    a = 0.0
```

```
55 ⟶b = 1.0
56 ⟶def f(x):
57 ⟶⟶return x**3
58 ⟶def primitive(x):
59 ⟶⟶return x**4/4.
60 elif exemple==3:
61 ⟶n = 10
62 ⟶a = -10.0
63 ⟶b = 10.0
64 ⟶def f(x):
65 ⟶⟶return math.exp(-x**2)
66 ⟶def primitive(x):
67 ⟶⟶return 0. # on ne connait pas la primitive
68 else:
69 ⟶print "Cas test non defini"
70 ⟶sys.exit(0)
71
72
73 print "** Exacte : ", primitive(b)-primitive(a)
74 print "Formule du rectangle a gauche composite : ", rectangle_gauche_composite(f,a,b,n)
75 print "Formule du rectangle a droite composite : ", rectangle_droite_composite(f,a,b,n)
76 print "Formule du point milieu composite : ", milieu_composite(f,a,b,n)
77 print "Formule des trapezes composite : ", trapeze_composite(f,a,b,n)
78 print "Formule de Simpson composite : ", simpson_composite(f,a,b,n)
79
80 # Dans python il existe un module qui implement deja ces methodes, comparons nos resultats avec ceux du
       ➥module:
81 from scipy import integrate
82 results = integrate.quad(f,a,b)
83 print "Avec scipy.integrate l'integrale est approchee par ", results[0], "avec une erreure de ", results
       ➥[1]
```

Exercices

Exercice 3.1

Estimer $\int_0^{5/2} f(x)\,\mathrm{d}x$ à partir des données

x	0	1/2	1	3/2	2	5/2
$f(x)$	3/2	2	2	1.6364	1.2500	0.9565

en utilisant

1. la méthode des rectangles à gauche composite,
2. la méthode des rectangles à droite composite,
3. la méthode des trapèzes composite.

Correction de l'exercice 3.1. On a $a=0$, $b=\frac{5}{2}$ et $m=5$ donc $h=\frac{b-a}{m}=\frac{1}{2}$.

Méthode	$\int_a^b f(t)dt \simeq$
Méthode 1	$h\sum_{i=0}^{m-1} f(a+ih) = \frac{1}{2}\left(\frac{3}{2}+2+2+1.6364+1.2500\right) = 4.1932$
Méthode 2	$h\sum_{i=0}^{m-1} f(a+(i+1)h) = \frac{1}{2}(2+2+1.6364+1.2500+0.9565) = 3.92145$
Méthode 3	$h\left(\frac{1}{2}f(a)+\sum_{i=1}^{m-1} f(a+ih)+\frac{1}{2}f(b)\right) = \frac{1}{2}\left(\frac{3}{4}+2+2+1.6364+1.2500+\frac{0.9565}{2}\right) = 4.057325$

Exercice 3.2

Étant donnée l'égalité

$$\pi = 4\left(\int_0^{+\infty} e^{-x^2}\,\mathrm{d}x\right)^2 = 4\left(\int_0^{10} e^{-x^2}\,\mathrm{d}x+\epsilon\right)^2,$$

avec $0<\epsilon<10^{-44}$, utiliser la méthode des trapèzes composite à 10 intervalles pour estimer la valeur de π.

Correction de l'exercice 3.2. La méthode des trapèzes composite à m intervalles pour calculer l'intégrale d'une fonction f sur l'intervalle $[a,b]$ s'écrit

$$\int_a^b f(t)dt \simeq h\left(\frac{1}{2}f(a)+\sum_{i=1}^{m-1} f(a+ih)+\frac{1}{2}f(b)\right) \qquad \text{avec } h=\frac{b-a}{m}.$$

Ici on a $f(x)=e^{-x^2}$, $a=0$, $b=10$, $m=10$ d'où $h=1$ et on obtient

$$I \simeq \frac{1}{2}+\sum_{i=1}^{10} e^{-i^2}+\frac{1}{2e^{100}} = \frac{1}{2}+\frac{1}{e}+\frac{1}{e^4}+\frac{1}{e^9}+\frac{1}{e^{16}}+\frac{1}{e^{25}}+\frac{1}{e^{36}}+\frac{1}{e^{49}}+\frac{1}{e^{64}}+\frac{1}{e^{81}}+\frac{1}{2e^{100}},$$

ainsi en utilisant la fonction `trapeze_composite(f,a,b,m)` décrite à la page 24 comme suit

```
import math

def f(x):
    return math.exp(-(x**2))

I = trapeze_composite(f,0,10,10)
print (4.*I**2)
```

on obtient $\pi \approx 4I^2 = 3.14224265994$.

Exercice 3.3

Estimer $\int_0^{\pi} \sin(x)\,\mathrm{d}x$ en utilisant la méthode des trapèzes composite avec 8 et puis 16 sous-intervalles en prenant en compte l'erreur.

Correction de l'exercice 3.3. La méthode des trapèzes composite à $m+1$ points pour calculer l'intégrale d'une fonction f sur l'intervalle $[a,b]$ s'écrit

$$\int_a^b f(t)\,dt \simeq h\left(\frac{1}{2}f(a)+\sum_{i=1}^{m-1} f(a+ih)+\frac{1}{2}f(b)\right) \qquad \text{avec } h=\frac{b-a}{m}$$

et l'erreur est donné par

$$E=-\frac{b-a}{12}h^2 f''(\xi)$$

avec $a<\xi<b$.

Ici on a $a=0$, $b=\pi$. Avec 8 sous-intervalles on a $h=\pi/8$ donc

$$\int_0^\pi \sin(x)\,\mathrm{d}x \simeq \frac{\pi}{8}\left(\frac{\sin(0)}{2}+\sum_{i=1}^{7}\sin(i\pi/8)+\frac{\sin(\pi)}{2}\right)\approx 1.97423$$

et l'erreur est

$$E=\frac{\pi^3}{768}\sin(\xi)$$

pour $\xi\in]0;\pi[$. Comme on ne connait pas la valeur de ξ, on ne peut pas connaitre E mais on peut en déterminer les bornes :

$$E_{\min}=\frac{\pi^3}{768}\sin(0)=0 \qquad\qquad E_{\max}=\frac{\pi^3}{768}\sin(\pi/2)=\frac{\pi^3}{768}\simeq 0.04037$$

ainsi

$$(1.97423-0)\le\int_0^\pi \sin(x)\,\mathrm{d}x\le(1.97423+0.04037)=2.01460$$

La valeur exacte est bien évidemment 2.

Avec 16 sous-intervalles on a $h=\pi/16$ et les nouveaux nœuds se trouvent au milieux des sous-intervalles précédents : $x_j=\pi/16+j\pi/8=(1+2j)\pi/16$ pour $j=0,1,\dots,7$, ainsi

$$\int_0^\pi \sin(x)\,\mathrm{d}x\simeq\frac{1.97423}{2}+\frac{\pi}{16}\sum_{j=0}^{7}\sin((1+2j)\pi/16)\approx 1.99358$$

et le limites de l'erreur deviennent (observons que E est divisé par 4 lorsque h est divisé par 2) :

$$E_{\min}=0 \qquad\qquad E_{\max}\simeq\frac{0.04037}{4}=0.01009$$

ainsi

$$1.99358\le\int_0^\pi \sin(x)\,\mathrm{d}x\le(1.99358+0.01009)=2.00367.$$

Exercice 3.4

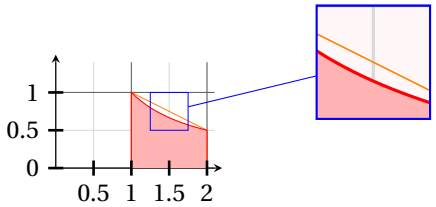
On considère l'intégrale

$$I=\int_1^2\frac{1}{x}\,\mathrm{d}x.$$

1. Calculer la valeur exacte de I.
2. Évaluer numériquement cette intégrale par la méthode des trapèzes avec $m=3$ sous-intervalles.
3. Pourquoi la valeur numérique obtenue à la question précédente est-elle supérieure à $\ln(2)$? Est-ce vrai quelque soit m ? Justifier la réponse. (On pourra s'aider par un dessin.)
4. Quel nombre de sous-intervalles m faut-il choisir pour avoir une erreur inférieure à 10^{-4} ? On rappelle que l'erreur de quadrature associée s'écrit, si $f\in\mathscr{C}^2([a;b])$,

$$|E_m|=\left|\frac{(b-a)^4}{12m^2}f''(\xi)\right|, \qquad \xi\in]a;b[.$$

CORRECTION DE L'EXERCICE 3.4.

1. Une primitive de $\frac{1}{x}$ est $F(x)=\ln(x)$. La valeur exacte est alors $I=\Big[\ln(x)\Big]_{x=1}^{x=2}=\ln(2)$.
2. La méthode des trapèzes composite à $m+1$ points pour calculer l'intégrale d'une fonction f sur l'intervalle $[a,b]$ s'écrit
$$\int_a^b f(t)dt \simeq h\left(\frac{1}{2}f(a)+\sum_{i=1}^{m-1} f(a+ih)+\frac{1}{2}f(b)\right) \qquad \text{avec } h=\frac{b-a}{m}.$$
Ici on a $f(x)=\frac{1}{x}$, $a=1$, $b=2$, $m=3$ d'où $h=\frac{1}{3}$ et on obtient
$$I\simeq\frac{1}{3}\left(\frac{1}{2}f(1)+f(1+1/3)+f(1+2/3)+\frac{1}{2}f(2)\right)=\frac{1}{3}\left(\frac{1}{2}+\frac{3}{4}+\frac{3}{5}+\frac{1}{4}\right)=\frac{21}{30}=0{,}7.$$
3. La valeur numérique obtenue à la question précédente est supérieure à $\ln(2)$ car la fonction $f(x)=\frac{1}{x}$ est convexe. On peut se convaincre à l'aide d'un dessin que les trapèzes sont au-dessus de la courbe $y=1/x$, l'aire sous les trapèzes sera donc supérieure à l'aire sous la courbe. Pour bien visualiser la construction considérons $m=1$:


Cela reste vrai quelque soit le pas h choisi car la fonction est convexe ce qui signifie qu'une corde définie par deux points de la courbe $y=1/x$ sera toujours au-dessus de la courbe et par le raisonnement précédant l'aire sous les trapèzes sera supérieure à l'aire exacte.
4. L'erreur est majorée par
$$|E_m|\leq\frac{(b-a)^4}{12m^2}\sup_{\xi\in]a;b[}|f''(\xi)|.$$
Donc ici on a $f(x)=1/x$, $f'(x)=-1/x^2$ et $f''(x)=2/x^3$, ainsi
$$|E_m|\leq\frac{1}{12m^2}\max_{\xi\in]1;2[}\frac{2}{\xi^3}=\frac{1}{6m^2}.$$
Pour que $|E_m|<10^{-4}$ il suffit que $\frac{1}{6m^2}<10^{-4}$, *i.e.* $m>10^2/\sqrt{6}\approx 40{,}8$. À partir de 41 sous-intervalles, l'erreur de quadrature est inférieure à 10^{-4}.

Exercice 3.5

On considère l'intégrale
$$I=\int_1^2 \ln(x)\,\mathrm{d}x.$$

1. Évaluer numériquement cette intégrale par la méthode des trapèzes composite avec $m=4$ sous-intervalles et comparer le résultat ainsi obtenu avec la valeur exacte. Pourquoi la valeur numérique est-elle inférieure à la valeur exacte ? Est-ce vrai quel que soit m ? (Justifier la réponse.)
2. Quel nombre de sous-intervalles m faut-il choisir pour avoir une erreur E_m inférieure à 10^{-2} ? On rappelle que, pour une fonction f de classe $\mathscr{C}^2$, l'erreur de quadrature E_m associée à la méthode des trapèzes composite avec une discrétisation uniforme de pas $h=(b-a)/m$ de l'intervalle $[a,b]$ en m sous-intervalles vérifie
$$|E_m|=\left|\frac{(b-a)}{12}h^2f''(\xi)\right|, \qquad \xi\in]a;b[.$$

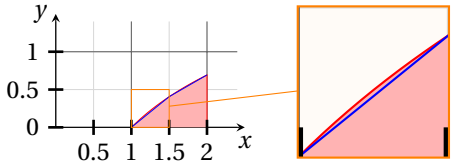
CORRECTION DE L'EXERCICE 3.5.

1. La méthode des trapèzes composite à $m+1$ points (m sous-intervalles) pour calculer l'intégrale d'une fonction f sur l'intervalle $[a,b]$ s'écrit
$$\int_a^b f(t)dt \simeq h\left(\frac{1}{2}f(a)+\sum_{i=1}^{m-1} f(a+ih)+\frac{1}{2}f(b)\right) \qquad \text{avec } h=\frac{b-a}{m}.$$

Ici on a $f(x)=\ln(x)$, $a=1$, $b=2$, $m=4$ d'où $h=\frac{1}{4}$ et on obtient

$$I \simeq \frac{1}{4}\left(\frac{1}{2}f(1)+f\left(\frac{5}{4}\right)+f\left(\frac{3}{2}\right)+f\left(\frac{7}{4}\right)+\frac{1}{2}f(2)\right)=\frac{1}{4}\left(\ln\left(\frac{5}{4}\right)+\ln\left(\frac{3}{2}\right)+\ln\left(\frac{7}{4}\right)+\frac{1}{2}\ln(2)\right)\approx 0.3836995094.$$

Une primitive de $\ln(x)$ est $F(x)=x(\ln(x)-1)$. La valeur exacte est alors $I=[x(\ln(x)-1)]_{x=1}^{x=2}=2\ln(2)-1\approx 0.386294361$. La valeur numérique obtenue est inférieure à celle exacte quelque soit le pas h choisi car la fonction f est concave, ce qui signifie qu'une corde définie par deux points de la courbe $y=\ln(x)$ sera toujours en-dessous de la courbe, donc l'aire sous les trapèzes sera inférieure à l'aire exacte. Pour bien visualiser la construction considérons $m=2$:



2. L'erreur est majorée par

$$|E_m| \leq \frac{(b-a)}{12}h^2 \sup_{\xi\in]a;b[}|f''(\xi)| = \frac{(b-a)^3}{12m^2}\sup_{\xi\in]a;b[}|f''(\xi)|.$$

On a $f(x)=\ln(x)$, $f'(x)=1/x$ et $f''(x)=-1/x^2$, ainsi

$$|E_m| \leq \frac{1}{12m^2}\max_{\xi\in]1;2[}\frac{1}{\xi^2}=\frac{1}{12m^2}.$$

Pour que $|E_m|<10^{-2}$ il suffit que $\frac{1}{12m^2}<10^{-2}$, *i.e.* $m>10/\sqrt{12}\approx 2.886$. À partir de 3 sous-intervalles, l'erreur de quadrature est inférieure à 10^{-2}.

Exercice 3.6

Écrire la méthode de NEWTON pour calculer la solution de l'équation

$$\int_0^x e^{-t^2}\,\mathrm{d}t=\frac{1}{3}$$

en proposant une formule de quadrature.

CORRECTION DE L'EXERCICE 3.6. Soit $f:\mathbb{R}\to\mathbb{R}$ la fonction définie par $f(x)=\int_0^x e^{-t^2}\,\mathrm{d}t-\frac{1}{3}$. On cherche les zéros de cette fonction par la méthode de NEWTON :

$$\begin{cases} x_0=0, \\ x_{n+1}=x_n-\frac{f(x_n)}{f'(x_n)}=x_n-\dfrac{\int_0^{x_n}e^{-t^2}\,\mathrm{d}t-\frac{1}{3}}{e^{-x_n^2}}. \end{cases}$$

À chaque étape n il faut donc calculer $\int_0^{x_n}e^{-t^2}\,\mathrm{d}t$ qu'on peut approcher par exemple par la méthode du trapèze :

$$\int_0^{x_n}e^{-t^2}\,\mathrm{d}t\simeq\frac{x_n}{2}\left(1+e^{-x_n^2}\right)$$

ce qui donne la suite

$$\begin{cases} x_0=0, \\ x_{n+1}=\frac{1}{2}x_n-\frac{1}{2}x_n e^{x_n^2}+\frac{1}{3}e^{x_n^2} \end{cases}$$

Exercice 3.7

Soit f une fonction $\mathscr{C}^\infty(\mathbb{R},\mathbb{R})$.

1. On considère l'approximation

$$\int_{-1}^1 f(x)\,\mathrm{d}x\approx\frac{1}{12}\left(11f\left(-\tfrac{3}{5}\right)+f\left(-\tfrac{1}{5}\right)+f\left(\tfrac{1}{5}\right)+11f\left(\tfrac{3}{5}\right)\right).$$

Quel est le degré de précision de cette formule de quadrature ?

2. On se donne les points $\{x_i\}_{i=0}^{i=n}$ de subdivision de l'intervalle $[a;b]$: $x_i = a + ih$ avec $h = \frac{b-a}{n}$. À l'aide d'un changement de variable affine, en déduire une formule de quadrature pour l'intégrale

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x.$$

En tirer une formule de quadrature composite pour l'intégrale $\int_a^b f(x)\,\mathrm{d}x$.

3. Écrire l'algorithme pour approcher $\int_a^b f(x)\,\mathrm{d}x$.

CORRECTION DE L'EXERCICE 3.7.

1. On a

k	$p_k(x) = x^k$	$\int_{-1}^{1} p_k(x)\,\mathrm{d}x$	$\frac{1}{12}(11p_k(-3/5) + p_k(-1/5) + p_k(1/5) + 11p_k(3/5))$	Degré d'exactitude
0	1	2	2	au moins 0
1	x	0	0	au moins 1
2	x^2	2/3	2/3	au moins 2
3	x^3	0	0	au moins 3
4	x^4	2/5	446/1875	≤ 3

La formule est donc exacte de degré 3.

2. Soit $x = mt + q$, alors

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x = m\int_{-1}^{1} f(mt+q)\,\mathrm{d}t \quad \text{avec} \quad \begin{cases} x_i = -m + q, \\ x_{i+1} = m + q, \end{cases}$$

d'où le changement de variable $x = x_i + (t+1)\frac{x_{i+1}-x_i}{2}$. On déduit la formule de quadrature (exacte sur l'espace des polynôme de degré au plus 3)

$$\begin{aligned}\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x &= \frac{x_{i+1}-x_i}{2}\int_{-1}^{1} f\left(x_i + (t+1)\frac{x_{i+1}-x_i}{2}\right)\mathrm{d}t \\ &\approx \frac{x_{i+1}-x_i}{24}\left[11f\left(x_i + \frac{x_{i+1}-x_i}{5}\right) + f\left(x_i + 2\frac{x_{i+1}-x_i}{5}\right) + f\left(x_i + 3\frac{x_{i+1}-x_i}{5}\right) + 11f\left(x_i + 4\frac{x_{i+1}-x_i}{5}\right)\right].\end{aligned}$$

Soit $h = x_{i+1} - x_i = \frac{b-a}{n}$. La formule précédente se réécrit

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x \approx \frac{h}{24}\left[11f\left(x_i + \frac{h}{5}\right) + f\left(x_i + \frac{2h}{5}\right) + f\left(x_i + \frac{3h}{5}\right) + 11f\left(x_i + \frac{4h}{5}\right)\right].$$

et la formule de quadrature composite déduite de cette approximation est

$$\int_a^b f(x)\,\mathrm{d}x = \sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x \approx \frac{h}{24}\sum_{i=0}^{n-1}\left[11f\left(x_i + \frac{h}{5}\right) + f\left(x_i + \frac{2h}{5}\right) + f\left(x_i + \frac{3h}{5}\right) + 11f\left(x_i + \frac{4h}{5}\right)\right].$$

3. Algorithme d'approximation de $\int_a^b f(x)\,\mathrm{d}x$

Require: $a\,;\, b > a\,;\, n > 0\,;\, f\colon [a;b] \to \mathbb{R}$
  $h \leftarrow \frac{b-a}{n}$
  $s \leftarrow 0$
  for $i = 0$ to $n-1$ **do**
    $x \leftarrow a + ih$
    $s \leftarrow s + 11f\left(x + \frac{h}{5}\right) + f\left(x + \frac{2h}{5}\right) + f\left(x + \frac{3h}{5}\right) + 11f\left(x + \frac{4h}{5}\right)$
  end for
  return $I \leftarrow \frac{h}{24}s$

Exercice 3.8

Soit f une fonction $\mathscr{C}^\infty(\mathbb{R},\mathbb{R})$.

1. On considère l'approximation

$$\int_{-1}^{1} f(x)\,\mathrm{d}x \approx \frac{2}{3}\left(2f\left(-\tfrac{1}{2}\right) - f(0) + 2f\left(\tfrac{1}{2}\right)\right).$$

Quel est le degré de précision de cette formule de quadrature ?

2. On se donne les points $\{x_i\}_{i=0}^{i=n}$ de subdivision de l'intervalle $[a;b]$: $x_i = a + ih$ avec $h = \frac{b-a}{n}$. À l'aide d'un changement de variable affine, en déduire une formule de quadrature pour l'intégrale

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x.$$

En tirer une formule de quadrature composite pour l'intégrale $\int_a^b f(x)\,\mathrm{d}x$.

3. Écrire l'algorithme pour approcher $\int_a^b f(x)\,\mathrm{d}x$.

CORRECTION DE L'EXERCICE 3.8.

1. On a

k	$p_k(x) = x^k$	$\int_{-1}^{1} p_k(x)\,\mathrm{d}x$	$\frac{2}{3}(2p_k(-1/2) - p_k(0) + 2p_k(1/2))$	Degré d'exactitude
0	1	2	2	au moins 0
1	x	0	0	au moins 1
2	x^2	2/3	2/3	au moins 2
3	x^3	0	0	au moins 3
4	x^4	2/5	1/6	3

La formule est donc exacte de degré 3.

2. Soit $x = mt + q$, alors

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x = m\int_{-1}^{1} f(mt+q)\,\mathrm{d}t \quad \text{avec} \quad \begin{cases} x_i = -m + q, \\ x_{i+1} = m + q, \end{cases}$$

d'où le changement de variable $x = x_i + (t+1)\frac{x_{i+1}-x_i}{2}$. On déduit la formule de quadrature (exacte sur l'espace des polynôme de degré au plus 3)

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x = \frac{x_{i+1}-x_i}{2}\int_{-1}^{1} f\left(x_i + (t+1)\frac{x_{i+1}-x_i}{2}\right)\mathrm{d}t \approx \frac{x_{i+1}-x_i}{3}\left[2f\left(\frac{x_i + \frac{x_{i+1}+x_i}{2}}{2}\right) - f\left(\frac{x_{i+1}+x_i}{2}\right) + 2f\left(\frac{\frac{x_{i+1}+x_i}{2} + x_{i+1}}{2}\right)\right].$$

Soit $h = x_{i+1} - x_i = \frac{b-a}{n}$. La formule précédente se réécrit

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x \approx \frac{h}{3}\left[2f\left(x_i + \frac{h}{4}\right) - f\left(x_i + \frac{h}{2}\right) + 2f\left(x_i + \frac{3h}{4}\right)\right].$$

et la formule de quadrature composite déduite de cette approximation est

$$\int_a^b f(x)\,\mathrm{d}x = \sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x \approx \frac{h}{3}\sum_{i=0}^{n-1}\left[2f\left(x_i + \frac{h}{4}\right) - f\left(x_i + \frac{h}{2}\right) + 2f\left(x_i + \frac{3h}{4}\right)\right].$$

3. Algorithme d'approximation de $\int_a^b f(x)\,\mathrm{d}x$

Require: a ; $b > a$; $n > 0$; $f : [a;b] \to \mathbb{R}$
  $h \leftarrow \frac{b-a}{n}$
  $s \leftarrow 0$
  for $i = 0$ to $n-1$ **do**
    $x \leftarrow a + ih$
    $s \leftarrow s + 2f\left(x + \frac{h}{4}\right) - f\left(x + \frac{h}{2}\right) + 2f\left(x + \frac{3h}{4}\right)$
  end for
  return $I \leftarrow \frac{h}{3}s$

Exercice 3.9

Soit f une fonction $\mathscr{C}^\infty(\mathbb{R},\mathbb{R})$. On se donne les points $\{x_i\}_{i=0}^{i=n}$ de subdivision de l'intervalle $[a;b]$: $x_i = a + ih$ avec $h = \frac{b-a}{n}$. Le but de l'exercice est de trouver une formule de quadrature à n points pour approcher l'intégrale

$$\int_a^b f(x)\ln\left(\frac{1}{x}\right)\mathrm{d}x.$$

1. On désire développer une formule d'intégration numérique de la forme
$$\int_0^1 f(x)\ln\left(\frac{1}{x}\right)\mathrm{d}x \approx \alpha f(\beta).$$
Déterminer les valeurs des constantes α et β de telle sorte que le degré d'exactitude de cette formule de quadrature soit le plus élevé possible. Donner ce degré.
Rappel : pour $m \geq 0$,
$$\int_0^1 x^m \ln\left(\frac{1}{x}\right)\mathrm{d}x = \frac{1}{(m+1)^2}.$$

2. À l'aide d'un changement de variable affine, en déduire une formule de quadrature pour l'intégrale
$$\int_{x_i}^{x_{i+1}} f(x)\ln\left(\frac{1}{x}\right)\mathrm{d}x.$$

3. On subdivise l'intervalle $[a;b]$ en n intervalles $[x_i;x_{i+1}]$ de largeur h. En déduire la formule de quadrature composite pour le calcul approché de
$$\int_a^b f(x)\ln\left(\frac{1}{x}\right)\mathrm{d}x.$$

CORRECTION DE L'EXERCICE 3.9.

1. On pose $f(x) = x^m$ et on note $I_m = \int_0^1 x^m \ln\left(\frac{1}{x}\right)\mathrm{d}x$ la valeur exacte de l'intégrale et $\tilde{I}_m = \alpha\beta^m$ la valeur obtenue par la formule de quadrature. On a
   - ⋆ $I_0 = 1$ et $\tilde{I}_0 = \alpha$: pour que le degré soit au moins 0 il faut choisir $\alpha = 1$;
   - ⋆ $I_1 = \frac{1}{4}$ et $\tilde{I}_1 = \alpha\beta = \beta$: pour que le degré soit au moins 1 il faut choisir $x_0 = \frac{1}{4}$;
   - ⋆ $I_2 = \frac{1}{9}$ et $\tilde{I}_2 = \alpha\beta^2 = \frac{1}{16}$: le degré d'exactitude est au plus 1.
2. Soit le changement de variable affine
$$\begin{aligned} x\colon [0;1] &\to [x_i;x_{i+1}] \\ t &\mapsto x = mt+q \end{aligned}$$
tel que
$$\begin{cases} x_i = q, \\ x_{i+1} = m+q, \end{cases} \quad \textit{i.e.} \quad \begin{cases} m = (x_{i+1}-x_i) = h, \\ q = x_i = a+ih, \end{cases}$$
alors
$$\begin{aligned} \int_{x_i}^{x_{i+1}} f(x)\ln\left(\frac{1}{x}\right)\mathrm{d}x &= m\int_0^1 f(mt+q)\ln\left(\frac{1}{mt+q}\right)\mathrm{d}t \\ &\approx mf\left(m\frac{1}{4}+q\right) = hf\left(\frac{h}{4}+x_i\right) = hf\left(a+\left(i+\frac{1}{4}\right)h\right). \end{aligned}$$
3. On trouve la formule de quadrature composite
$$\int_a^b f(x)\ln\left(\frac{1}{x}\right)\mathrm{d}x = \sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} f(x)\ln\left(\frac{1}{x}\right)\mathrm{d}x \approx h\sum_{i=0}^{n-1} f\left(a+\left(i+\frac{1}{4}\right)h\right).$$

Exercice 3.10

Soit f une fonction $\mathscr{C}^\infty(\mathbb{R},\mathbb{R})$. On se donne les points $\{x_i\}_{i=0}^{i=n}$ de subdivision de l'intervalle $[a;b]$: $x_i = a+ih$ avec $h = \frac{b-a}{n}$. Le but de l'exercice est de trouver une formule de quadrature à $2n$ points pour approcher l'intégrale
$$\int_a^b f(x)\,\mathrm{d}x. \tag{3.4}$$
On propose dans un premier temps (question 1 à 4) de construire la formule de quadrature à deux points suivantes :
$$\int_{-1}^1 g(x)\,\mathrm{d}x \approx g(-\alpha)+g(\alpha), \tag{3.5}$$
où $0 < \alpha < 1$ est à déterminer.

1. Choisir α pour rendre la formule de quadrature exacte pour des polynômes de degré le plus élevé possible. Quel est alors le degré de précision de cette formule de quadrature ?
2. À l'aide d'un changement de variable affine, étendre cette formule de quadrature pour l'intégrale suivante :
$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x.$$
3. En déduire une formule de quadrature à $2n$ points, notée F, pour le calcul approché de (3.4). Cette formule de quadrature est-elle stable ?
4. Écrire l'algorithme du calcul de F.

CORRECTION DE L'EXERCICE 3.10.

1. On a

k	$p_k(x)=x^k$	$\int_{-1}^{1} p_k(x)\,\mathrm{d}x$	$p_k(-\alpha)+p_k(\alpha)$	Degré d'exactitude
0	1	2	2	au moins 0
1	x	0	0	au moins 1
2	x^2	2/3	$2\alpha^2$	1 si $\alpha \neq 1/\sqrt{3}$, au moins 2 si $\alpha = 1/\sqrt{3}$
Soit $\alpha = 1/\sqrt{3}$				
3	x^3	0	0	au moins 3
4	x^4	2/5	2/9	3

Donc la formule de quadrature a degré de précision 1 si $\alpha \neq 1/\sqrt{3}$ et degré de précision 3 si $\alpha = 1/\sqrt{3}$.

2. Par le changement de variable $y = x_i + (x+1)\frac{x_{i+1}-x_i}{2}$ on déduit la formule de quadrature
$$\begin{aligned}\int_{x_i}^{x_{i+1}} f(y)\,\mathrm{d}y &= \frac{x_{i+1}-x_i}{2}\int_{-1}^{1} f\left(x_i+(x+1)\frac{x_{i+1}-x_i}{2}\right)\mathrm{d}x\\ &\approx \frac{x_{i+1}-x_i}{2}\left[f\left(x_i+\left(1-\sqrt{\tfrac{1}{3}}\right)\frac{x_{i+1}-x_i}{2}\right)+f\left(x_i+\left(1+\sqrt{\tfrac{1}{3}}\right)\frac{x_{i+1}-x_i}{2}\right)\right].\end{aligned}$$

3. Si $h = x_{i+1}-x_i = \frac{b-a}{n}$ (*i.e.* si on considère une subdivision de l'intervalle $[a;b]$ équirépartie) alors on trouve la formule de quadrature composite (*i.e.* sur n sous-intervalles et à $2n$ points)
$$\int_a^b f(x)\,\mathrm{d}x \approx \frac{h}{2}\sum_{i=0}^{n-1}\left[f\left(x_i+h\left(1-\sqrt{\tfrac{1}{3}}\right)\right)+f\left(x_i+h\left(1+\sqrt{\tfrac{1}{3}}\right)\right)\right] = \frac{h}{2}\sum_{i=0}^{n-1}\left[f\left(a+h\left(i+1-\sqrt{\tfrac{1}{3}}\right)\right)+f\left(a+h\left(i+1+\sqrt{\tfrac{1}{3}}\right)\right)\right].$$
Cette formule de quadrature est stable puisque tous les coefficients sont positifs.

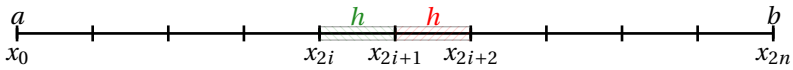
4. Algorithme du calcul de F :

Require: $a\,;\, b>a\,;\, n>0\,;\, f\colon [a;b]\to\mathbb{R}$
  $h \leftarrow \frac{b-a}{n}$
  $\alpha_1 \leftarrow a+\left(1-\sqrt{\frac{1}{3}}\right)h$
  $\alpha_2 \leftarrow a+\left(1+\sqrt{\frac{1}{3}}\right)h$
  for $i=0$ to $n-1$ **do**
    $s \leftarrow s+f(\alpha_1+ih)+f(\alpha_2+ih)$
  end for
  return $I \leftarrow \frac{h}{2}s$

Exercice 3.11 *Interpolation et Intégration*

1. Soit f une fonction de classe $\mathscr{C}^1([-1,1])$ et p le polynôme de LAGRANGE qui interpole f aux points -1, 0 et 1. Écrire le polynôme p.
2. En déduire une méthode de quadrature pour approcher l'intégrale
$$\int_{-1}^{1} f(t)\,\mathrm{d}t.$$
3. Étudier le degré de précision de la formule de quadrature ainsi trouvée.

4. À l'aide d'un changement de variable affine, en déduire une formule de quadrature pour l'intégrale
$$\int_{x_{2i}}^{x_{2i+2}} f(x)\,\mathrm{d}x.$$
5. Soit $h=\frac{b-a}{2n}$ et $x_i=a+ih$ pour $i=0,\dots,2n$. On subdivise l'intervalle $[a;b]$ en n intervalles $[x_{2i};x_{2i+2}]$ de largeur $2h$.



En déduire la formule de quadrature composite pour le calcul approché de
$$\int_a^b f(x)\,\mathrm{d}x.$$
6. Écrire l'algorithme associé à cette formule de quadrature.

CORRECTION DE L'EXERCICE 3.11.

1. On a trois points, donc le polynôme interpolateur de LAGRANGE est un polynôme de $\mathbb{R}_2[x]$. On cherche alors les coefficients α, β et γ du polynôme $p(x)=\alpha+\beta x+\gamma x^2$ tels que
$$\begin{cases} f(-1)=\alpha-\beta+\gamma, & \text{(3.6a)}\\ f(0)=\alpha, & \text{(3.6b)}\\ f(1)=\alpha+\beta+\gamma. & \text{(3.6c)}\end{cases}$$
L'équation (3.6b) donne $\alpha=f(0)$, la somme (3.6c)+(3.6a) donne $\gamma=\frac{f(1)+f(-1)}{2}-f(0)$ et enfin la soustraction (3.6c)−(3.6a) donne $\beta=\frac{f(1)-f(-1)}{2}$.
2. On en déduit la méthode de quadrature
$$\int_{-1}^{1} f(t)\,\mathrm{d}t\approx\int_{-1}^{1} p(t)\,\mathrm{d}t=2(\alpha+\gamma/3)=\frac{f(-1)+4f(0)+f(1)}{3}.$$
3. Par construction, cette formule de quadrature a degré de précision au moins 2. De plus

k	$p_k(x)=x^k$	$\int_{-1}^{1} p_k(x)\,\mathrm{d}x$	$\frac{p_k(-1)+4p_k(0)+p_k(1)}{3}$	Degré d'exactitude
3	x^3	0	0	au moins 3
4	x^4	2/5	2/3	3

La formule est exacte pour les polynômes de degré au plus 3.
4. Soit $x=mt+q$, alors
$$\int_{x_{2i}}^{x_{2i+2}} f(x)\,\mathrm{d}x=m\int_{-1}^{1} f(mt+q)\,\mathrm{d}t \quad\text{avec}\quad \begin{cases} x_{2i}=-m+q,\\ x_{2i+2}=m+q,\end{cases} \quad i.e. \quad \begin{cases} m=\frac{x_{2i+2}-x_{2i}}{2},\\ q=\frac{x_{2i+2}+x_{2i}}{2},\end{cases}$$
d'où le changement de variable $x=x_{2i}+(t+1)\frac{x_{2i+2}-x_{2i}}{2}$. On déduit la formule de quadrature (exacte sur l'espace des polynôme de degré au plus 3)
$$\int_{x_{2i}}^{x_{2i+2}} f(x)\,\mathrm{d}x=\frac{x_{2i+2}-x_{2i}}{2}\int_{-1}^{1} f\left(x_{2i}+(t+1)\frac{x_{2i+2}-x_{2i}}{2}\right)\mathrm{d}t\approx\frac{x_{2i+2}-x_{2i}}{6}\left[f(x_{2i})+4f(x_{2i+1})+f(x_{2i+2})\right].$$
5. $h=\frac{b-a}{2n}=x_{i+1}-x_i$ pour $i=0,\dots,2n-1$. On subdivise l'intervalle $[a;b]$ en n intervalles $[x_{2i};x_{2i+2}]$ de largeur $2h$. On trouve ainsi la formule de quadrature composite
$$\begin{aligned}\int_a^b f(x)\,\mathrm{d}x=\sum_{i=0}^{n-1}\int_{x_{2i}}^{x_{2i+2}} f(x)\,\mathrm{d}x&\approx\sum_{i=0}^{n-1}\frac{x_{2i+2}-x_{2i}}{6}\left[f(x_{2i})+4f(x_{2i+1})+f(x_{2i+2})\right]\\&=\frac{h}{3}\sum_{i=0}^{n-1}\left[f(x_{2i})+4f(x_{2i+1})+f(x_{2i+2})\right]\\&=\frac{h}{3}\left[f(a)+f(b)+2\sum_{i=1}^{n-1}f(x_{2i})+4\sum_{i=0}^{n-1}f(x_{2i}+h)\right]\\&=\frac{h}{3}\left[f(a)+f(b)+2\sum_{i=1}^{n-1}f(a+2ih)+4\sum_{i=0}^{n-1}f(a+(2i+1)h)\right].\end{aligned}$$

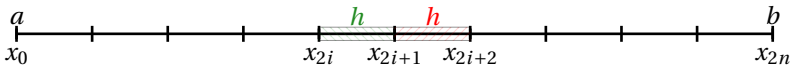
6. Algorithme du calcul associé à cette formule de quadrature

Require: $a\,;\, b>a\,;\, n>0\,;\, f\colon [a;b]\to\mathbb{R}$
$h \leftarrow \frac{b-a}{2n}$
$s_1 \leftarrow 0$
$s_2 \leftarrow s_2 + f(a+h)$
for $i = 1$ to $n-1$ **do**
$s_1 \leftarrow s_1 + f(a+2ih)$
$s_2 \leftarrow s_2 + f(a+(2i+1)h)$
end for
return $\frac{h}{3}\left[f(a)+f(b)+2s_1+4s_2\right]$

Il s'agit de la **méthode de CAVALIERI-SIMPSON composite**.

Exercice 3.12 *Interpolation et Intégration*

1. Soit f une fonction de classe $\mathscr{C}^1([0,2])$ et p le polynôme de LAGRANGE qui interpole f aux points 0, 1 et 2. Écrire le polynôme p.
2. En déduire une méthode de quadrature pour approcher l'intégrale
$$\int_0^2 f(t)\,\mathrm{d}t.$$
3. Étudier le degré de précision de la formule de quadrature ainsi trouvée.
4. À l'aide d'un changement de variable affine, en déduire une formule de quadrature pour l'intégrale
$$\int_{x_{2i}}^{x_{2i+2}} f(x)\,\mathrm{d}x.$$
5. Soit $h = \frac{b-a}{2n}$ et $x_i = a + ih$ pour $i = 0,\dots,2n$. On subdivise l'intervalle $[a;b]$ en n intervalles $[x_{2i};x_{2i+2}]$ de largeur $2h$.



En déduire la formule de quadrature composite pour le calcul approché de
$$\int_a^b f(x)\,\mathrm{d}x.$$

CORRECTION DE L'EXERCICE 3.12.

1. On a trois points, donc le polynôme interpolateur de LAGRANGE est un polynôme de $\mathbb{R}_2[x]$. On cherche alors les coefficients α, β et γ du polynôme $p(x) = \alpha + \beta x + \gamma x^2$ tels que
$$\begin{cases} f(0) = \alpha, \\ f(1) = \alpha + \beta + \gamma, \\ f(2) = \alpha + 2\beta + 4\gamma. \end{cases}$$
Donc $\alpha = f(0)$, $\beta = -\frac{3}{2}f(0) + 2f(1) - \frac{1}{2}f(2)$ et $\gamma = \frac{1}{2}f(0) - f(1) + \frac{1}{2}f(2)$.
2. On en déduit la méthode de quadrature
$$\int_0^2 f(t)\,\mathrm{d}t \approx \int_0^2 p(t)\,\mathrm{d}t = 2\alpha + 2\beta + \frac{8}{3}\gamma = \frac{f(0)+4f(1)+f(2)}{3}.$$
3. Par construction, cette formule de quadrature a degré de précision au moins 2. De plus

k	$p_k(x) = x^k$	$\int_0^2 p_k(x)\,\mathrm{d}x$	$\frac{p_k(0)+4p_k(1)+p_k(2)}{3}$	Degré d'exactitude
3	x^3	4	4	au moins 3
4	x^4	32/5	2/3	3

La formule est exacte pour les polynômes de degré au plus 3.

4. Soit $x = mt + q$, alors

$$\int_{x_{2i}}^{x_{2i+2}} f(x)\,\mathrm{d}x = m\int_0^2 f(mt+q)\,\mathrm{d}t \quad \text{avec} \quad \begin{cases} x_{2i} = q, \\ x_{2i+2} = 2m+q, \end{cases} \qquad \textit{i.e.} \quad \begin{cases} m = \frac{x_{2i+2}-x_{2i}}{2}, \\ q = x_{2i}, \end{cases}$$

d'où le changement de variable $x = \frac{x_{2i+2}-x_{2i}}{2}t + x_{2i}$. On déduit la formule de quadrature (exacte sur l'espace des polynôme de degré au plus 3)

$$\int_{x_{2i}}^{x_{2i+2}} f(x)\,\mathrm{d}x = \frac{x_{2i+2}-x_{2i}}{2}\int_0^2 f\left(\frac{x_{2i+2}-x_{2i}}{2}t + x_{2i}\right)\mathrm{d}t \approx \frac{x_{2i+2}-x_{2i}}{6}\left[f(x_{2i}) + 4f(x_{2i+1}) + f(x_{2i+2})\right].$$

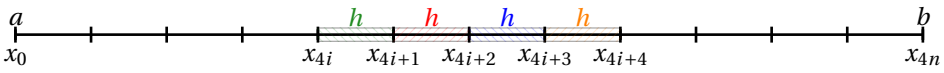
5. $h = \frac{b-a}{2n} = x_{i+1} - x_i$ pour $i = 0,\dots,2n-1$. On subdivise l'intervalle $[a;b]$ en n intervalles $[x_{2i};x_{2i+2}]$ de largeur $2h$. On trouve ainsi la formule de quadrature composite

$$\begin{aligned}
\int_a^b f(x)\,\mathrm{d}x = \sum_{i=0}^{n-1}\int_{x_{2i}}^{x_{2i+2}} f(x)\,\mathrm{d}x &\approx \sum_{i=0}^{n-1}\frac{x_{2i+2}-x_{2i}}{6}\left[f(x_{2i}) + 4f(x_{2i+1}) + f(x_{2i+2})\right] \\
&= \frac{h}{3}\sum_{i=0}^{n-1}\left[f(x_{2i}) + 4f(x_{2i+1}) + f(x_{2i+2})\right] \\
&= \frac{h}{3}\left[f(a) + f(b) + 2\sum_{i=1}^{n-1} f(x_{2i}) + 4\sum_{i=0}^{n-1} f(x_{2i}+h)\right] \\
&= \frac{h}{3}\left[f(a) + f(b) + 2\sum_{i=1}^{n-1} f(a+2ih) + 4\sum_{i=0}^{n-1} f(a+(2i+1)h)\right].
\end{aligned}$$

Il s'agit de la **méthode de CAVALIERI-SIMPSON composite**.

Exercice 3.13 *Interpolation et Intégration*

1. Soit f une fonction de classe $\mathscr{C}^1([0,4])$ et p le polynôme de LAGRANGE qui interpole f aux points 1, 2 et 3. Écrire le polynôme p.
2. En déduire une méthode de quadrature pour approcher l'intégrale
$$\int_0^4 f(t)\,\mathrm{d}t.$$
3. Étudier le degré de précision de la formule de quadrature ainsi trouvée.
4. À l'aide d'un changement de variable affine, en déduire une formule de quadrature pour l'intégrale
$$\int_{x_{4i}}^{x_{4i+4}} f(x)\,\mathrm{d}x.$$
5. Soit $h = \frac{b-a}{4n}$ et $x_i = a + ih$ pour $i = 0,\dots,4n$. On subdivise l'intervalle $[a;b]$ en n intervalles $[x_{4i};x_{4i+4}]$ de largeur $4h$.



En déduire la formule de quadrature composite pour le calcul approché de
$$\int_a^b f(x)\,\mathrm{d}x.$$

CORRECTION DE L'EXERCICE 3.13.

1. On a trois points, donc le polynôme interpolateur de LAGRANGE est un polynôme de $\mathbb{R}_2[x]$. On cherche alors les coefficients α, β et γ du polynôme $p(x) = \alpha + \beta x + \gamma x^2$ tels que
$$\begin{cases} f(1) = \alpha + \beta + \gamma, \\ f(2) = \alpha + 2\beta + 4\gamma, \\ f(3) = \alpha + 3\beta + 9\gamma. \end{cases}$$
Donc $\alpha = 3f(1) - 3f(2) + f(3)$, $\beta = -\frac{5}{2}f(1) + 4f(2) - \frac{3}{2}f(3)$ et $\gamma = \frac{1}{2}f(1) - f(2) + \frac{1}{2}f(3)$.

2. On en déduit la méthode de quadrature

$$\int_0^4 f(t)\,\mathrm{d}t \approx \int_0^4 p(t)\,\mathrm{d}t = 4\alpha + 8\beta + \frac{64}{3}\gamma = \frac{4}{3}(2f(1) - f(2) + 2f(3)).$$

3. Par construction, cette formule de quadrature a degré de précision au moins 2.
De plus, $\int_0^4 x^3\,\mathrm{d}x = 64$ et $\frac{4}{3}(2\times 1^3 - 2^3 + 2\times 3^3 = 64$ donc la formule est exacte pour les polynômes de degré au moins 3.
Enfin, $\int_0^4 x^4\,\mathrm{d}x = \frac{592}{3}$ et tandis que $\frac{8\times(1)^4 - 4\times(2)^4 + 8\times(3)^4}{3} = \frac{1024}{3}$ donc la formule est exacte pour les polynômes de degré au plus 3.

4. Soit le changement de variable affine

$$\begin{aligned} x\colon [0;4] &\to [x_{4i}; x_{4i+4}] \\ t &\mapsto x = mt + q \end{aligned}$$

tel que

$$\begin{cases} x_{4i} = q, \\ x_{4i+4} = 4m + q, \end{cases} \quad \textit{i.e.} \quad \begin{cases} m = \frac{x_{4i+4} - x_{4i}}{4} = h, \\ q = x_{4i} = a + 4ih, \end{cases}$$

alors

$$\begin{aligned} \int_{x_{4i}}^{x_{4i+4}} f(x)\,\mathrm{d}x = m\int_0^4 f(mt+q)\,\mathrm{d}t &\approx \frac{4}{3}m\big(2f(m+q) - f(2m+q) + 2f(3m+q)\big) \\ &= \frac{x_{4i+4} - x_{4i}}{3}\left[2f(x_{4i+1}) - f(x_{4i+2}) + f(x_{4i+3})\right] = \frac{4}{3}h\left[2f(a+(4i+1)h) - f(a+(4i+2)h) + f(a+(4i+3)h)\right]. \end{aligned}$$

5. $h = \frac{b-a}{4n} = x_{j+1} - x_j$ pour $j = 0, \dots, 4n-1$. On subdivise l'intervalle $[a;b]$ en n intervalles $[x_{4i}; x_{4i+4}]$ de largeur $4h$. On trouve ainsi la formule de quadrature composite

$$\int_a^b f(x)\,\mathrm{d}x = \sum_{i=0}^{n-1}\int_{x_{4i}}^{x_{4i+4}} f(x)\,\mathrm{d}x \approx \frac{4h}{3}\sum_{i=0}^{n-1}\left[2f(a+(4i+1)h) - f(a+(4i+2)h) + 2f(a+(4i+3)h)\right].$$

Exercice 3.14

Soit f une fonction $\mathscr{C}^\infty(\mathbb{R},\mathbb{R})$. On se donne les points $\{x_i\}_{i=0}^{i=2n}$ de subdivision de l'intervalle $[a;b]$: $x_i = a + ih$ avec $h = \frac{b-a}{2n}$.
Le but de l'exercice est de trouver une formule de quadrature à $2n+1$ points basée sur la formule de SIMPSON pour approcher

$$\int_a^b f(x)\,\mathrm{d}x. \tag{3.7}$$

On propose dans un premier temps (question 1 à 3) de construire la formule de quadrature à 3 points de Simpson :

$$\int_{-1}^1 g(x)\,\mathrm{d}x \approx \alpha g(-1) + \beta g(0) + \alpha g(1), \tag{3.8}$$

où les réels α et β sont à déterminer.

1. Déterminer α et β pour que la formule de quadrature (3.8) ait degré de précision maximale.
2. À l'aide d'un changement de variable affine, en déduire une formule de quadrature exacte sur l'espace des polynôme de degré au plus 3 pour l'intégrale suivante :

$$\int_{x_{2i}}^{x_{2i+2}} f(x)\,\mathrm{d}x.$$

3. En déduire une formule de quadrature à $2n$ points, notée F, pour le calcul approché de (3.7). Cette formule de quadrature est-elle stable ?
4. Écrire l'algorithme du calcul de F.
5. Soit x un élément de $[x_i; x_{i+1}]$. Écrire une formule de TAYLOR $f(x) = P_i(x) + R_i(x)$ à l'ordre 3 pour f en x, avec $P_i \in \mathbb{P}_3$. Majorer R_i sur $[x_i; x_{i+1}]$ en fonction de h.
6. En déduire une estimation d'erreur entre (3.7) et F.

CORRECTION DE L'EXERCICE 3.14.

1. On a

k	$p_k(x)=x^k$	$\int_{-1}^{1} p_k(x)\,\mathrm{d}x$	$\alpha p_k(-1)+\beta p_k(0)+\alpha p_k(1)$	Degré d'exactitude
0	1	2	$2\alpha+\beta$	même pas 0 si $\beta\neq 2(1-\alpha)$, au moins 0 si $\beta=2(1-\alpha)$
Soit $\beta=2(1-\alpha)$				
1	x	0	0	au moins 1
2	x^2	2/3	2α	1 si $\alpha\neq 1/3$, au moins 2 si $\alpha=1/3$
Soit $\alpha=1/3$				
3	x^3	0	0	au moins 3
4	x^4	2/5	2/3	3

Si $\beta\neq 2(1-\alpha)$ la formule de quadrature n'est même pas exacte pour une constante, si $\beta=2(1-\alpha)$ mais $\alpha\neq 1/3$, elle est exacte pour les polynômes de degré au plus 1, si $\alpha=1/3$ et $\beta=4/3$ la formule est exacte pour les polynômes de degré au plus 3.

2. Soit $x=mt+q$, alors
$$\int_{x_{2i}}^{x_{2i+2}} f(x)\,\mathrm{d}x = m\int_{-1}^{1} f(mt+q)\,\mathrm{d}t \quad \text{avec} \quad \begin{cases} x_{2i}=-m+q, \\ x_{2i+2}=m+q, \end{cases}$$
d'où le changement de variable $x=x_{2i}+(t+1)\frac{x_{2i+2}-x_{2i}}{2}$. On déduit la formule de quadrature (exacte sur l'espace des polynôme de degré au plus 3)
$$\int_{x_{2i}}^{x_{2i+2}} f(x)\,\mathrm{d}x = \frac{x_{2i+2}-x_{2i}}{2}\int_{-1}^{1} f\left(x_{2i}+(t+1)\frac{x_{2i+2}-x_{2i}}{2}\right)\mathrm{d}t \approx \frac{x_{2i+2}-x_{2i}}{6}\left[f(x_{2i})+4f(x_{2i+1})+f(x_{2i+2})\right].$$

3. On trouve ainsi la formule de quadrature composite (*i.e.* sur n sous-intervalles)
$$\int_a^b f(x)\,\mathrm{d}x = \sum_{i=0}^{n-1}\int_{x_{2i}}^{x_{2i+2}} f(x)\,\mathrm{d}x \approx \sum_{i=0}^{n-1}\frac{x_{2i+2}-x_{2i}}{6}\left[f(x_{2i})+4f(x_{2i+1})+f(x_{2i+2})\right].$$
Si $h=\frac{x_{i+1}-x_i}{2}=\frac{b-a}{2n}$ (*i.e.* si on considère une subdivision de l'intervalle $[a;b]$ équirépartie) alors on a
$$\begin{aligned}\int_a^b f(x)\,\mathrm{d}x \approx \frac{h}{3}\sum_{i=0}^{n-1}\left[f(x_{2i})+4f(x_{2i+1})+f(x_{2i+2})\right] &= \frac{h}{3}\left[f(a)+f(b)+2\sum_{i=1}^{n-1}f(x_{2i})+4\sum_{i=0}^{n-1}f(x_{2i}+h)\right] \\ &= \frac{h}{3}\left[f(a)+f(b)+2\sum_{i=1}^{n-1}f(a+2ih)+4\sum_{i=0}^{n-1}f(a+(2i+1)h)\right].\end{aligned}$$
Cette formule de quadrature est stable puisque tous les coefficients sont positifs et on a
$$\frac{h}{3}\left[1+1+2\sum_{i=1}^{n-1}1+4\sum_{i=0}^{n-1}1\right] = \frac{b-a}{6n}\left[2+2(n-1)+4n\right] = \frac{b-a}{6n}6n = (b-a).$$

4. Algorithme du calcul de F :
Require: $f\colon [a,b]\to\mathbb{R}$, a, $b>a$, $n>0$
  $H \leftarrow \frac{b-a}{n}$
  $s_1 \leftarrow 0$
  $s_2 \leftarrow s_2 + f(a+H/2)$
  for $i=1$ to $n-1$ **do**
    $s_1 \leftarrow s_1 + f(a+iH)$
    $s_2 \leftarrow s_2 + f(a+(i+1)H/2)$
  end for
  return $I \leftarrow \frac{H}{6}\left[f(a)+f(b)+2s_1+4s_2\right]$

5. Soit x un élément de $[x_{2i};x_{2i+2}]$. Une formule de TAYLOR à l'ordre 3 pour f en x s'écrit
$$f(x) = P_i(x) + R_i(x),$$
avec
$$P_i(x) = f(x_{2i}) + (x-x_{2i})f'(x_{2i}) + (x-x_{2i})^2\frac{f''(x_{2i})}{2} + (x-x_{2i})^3\frac{f'''(x_{2i})}{6} \qquad \in \mathbb{P}_3$$

et le reste de LAGRANGE

$$R_i(x) = (x - x_{2i})^4 \frac{f^{IV}(\xi)}{24} \qquad \text{avec} \quad \xi \in]x_{2i}; x_{2i+2}[.$$

On peut majorer R_i sur $[x_{2i}; x_{2i+2}]$ en fonction de $H = x_{2i+2} - x_{2i}$:

$$|R_i(x)| \leq \frac{H^4}{24} \max|f^{IV}(\xi)| = \frac{b-a}{n} \frac{H^3}{24} \max|f^{IV}(\xi)|.$$

6. On en déduit l'estimation d'erreur entre (3.7) et F suivante [1]

$$\begin{aligned}
\left|\int_a^b f(x)\,\mathrm{d}x - F\right| &\leq \left|\sum_{i=0}^{n-1} \int_{x_{2i}}^{x_{2i+2}} P_i(x)\,\mathrm{d}x - F\right| + \left|\sum_{i=0}^{n-1} \int_{x_{2i}}^{x_{2i+2}} R_i(x)\,\mathrm{d}x\right| \\
&\leq nH|R_i(x_{2i+2})| + \left|\sum_{i=0}^{n-1} \int_{x_{2i}}^{x_{2i+2}} R_i(x)\,\mathrm{d}x\right| \\
&\leq nH \frac{b-a}{n} \frac{H^3}{24} \max|f^{IV}(\xi)| + nH \frac{b-a}{n} \frac{H^3}{24} \max|f^{IV}(\xi)| \\
&= (b-a) \frac{H^4}{12} \sup|f^{IV}(\xi)|.
\end{aligned}$$

Exercice 3.15

Soit f une fonction $\mathscr{C}^\infty(\mathbb{R},\mathbb{R})$. On se donne les points $\{x_i\}_{i=0}^{i=n}$ de subdivision de l'intervalle $[a;b]$: $x_i = a + ih$ avec $h = \frac{b-a}{n}$. Le but de l'exercice est de trouver une formule de quadrature à $3n$ points pour approcher l'intégrale

$$\int_a^b f(x)\,\mathrm{d}x. \tag{3.9}$$

On propose dans un premier temps de construire la formule de quadrature à trois points suivantes :

$$\int_{-1}^{1} g(x)\,\mathrm{d}x \approx \frac{2}{3}\big(g(-\alpha) + g(0) + g(\alpha)\big), \tag{3.10}$$

où le réel $0 < \alpha < 1$ sera à déterminer par la suite.

1. Déterminer α pour que la formule de quadrature (3.10) ait degré de précision maximale. Quel est alors le degré de précision de cette formule de quadrature ?
2. À l'aide d'un changement de variable affine, étendre cette formule de quadrature pour l'intégrale suivante :

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x.$$

3. En déduire une formule de quadrature à $3n$ points, notée F, pour le calcul approché de (3.9). Cette formule de quadrature est-elle stable ?

CORRECTION DE L'EXERCICE 3.15.

1. On a

k	$p_k(x) = x^k$	$\int_{-1}^{1} p_k(x)\,\mathrm{d}x$	$\frac{2}{3}(p_k(-\alpha) + \beta p_k(0) + p_k(\alpha))$	Degré d'exactitude
0	1	2	2	au moins 0
1	x	0	0	au moins 1
2	x^2	2/3	$2\alpha^2/3$	1 si $\alpha \neq 1/\sqrt{2}$, au moins 2 si $\alpha = 1/\sqrt{2}$
			Soit $\alpha = 1/\sqrt{2}$	
3	x^3	0	0	au moins 3
4	x^4	2/5	1/3	3

Si $\alpha \neq 1/\sqrt{2}$ la formule de quadrature est exacte pour les polynômes de degré au plus 1, si $\alpha = 1/\sqrt{2}$ la formule est exacte pour les polynômes de degré au plus 3.

2. Par le changement de variable $y = x_i + (x+1)\frac{x_{i+1}-x_i}{2}$ on déduit la formule de quadrature

$$\begin{aligned}
\int_{x_i}^{x_{i+1}} f(y)\,\mathrm{d}y &= \frac{x_{i+1}-x_i}{2} \int_{-1}^{1} f\left(x_i + (x+1)\frac{x_{i+1}-x_i}{2}\right)\mathrm{d}x \\
&\approx \frac{x_{i+1}-x_i}{3}\left[f\left(x_i + \left(1 - \sqrt{\tfrac{1}{2}}\right)\frac{x_{i+1}-x_i}{2}\right) + f\left(x_i + \frac{x_{i+1}-x_i}{2}\right) + f\left(x_i + \left(1 + \sqrt{\tfrac{1}{2}}\right)\frac{x_{i+1}-x_i}{2}\right)\right].
\end{aligned}$$

1. N.B. : le polynôme P_i n'est pas le polynôme d'interpolation en x_{2i}, x_{2i+2} et $(x_{2i} + x_{2i+2})/2$ donc $\int_{x_{2i}}^{x_{2i+2}} P_i(x)\,\mathrm{d}x - F \neq 0$.

3. Si $H = x_{i+1} - x_i = \frac{b-a}{n}$ (*i.e.* si on considère une subdivision de l'intervalle $[a;b]$ équirépartie) alors on trouve la formule de quadrature composite (*i.e.* sur n sous-intervalles et à $3n$ points)

$$\begin{aligned}\int_a^b f(x)\,\mathrm{d}x &\approx \frac{H}{3}\sum_{i=0}^{n-1}\left[f\left(x_i + H\left(1-\tfrac{1}{\sqrt{2}}\right)\right) + f\left(x_i+\tfrac{H}{2}\right) + f\left(x_i + H\left(1+\tfrac{1}{\sqrt{2}}\right)\right)\right]\\ &= \frac{H}{3}\sum_{i=0}^{n-1}\left[f\left(a + H\left(i+1-\tfrac{1}{\sqrt{2}}\right)\right) + f\left(x_i+\tfrac{H}{2}\right) + f\left(a + H\left(i+1+\tfrac{1}{\sqrt{2}}\right)\right)\right].\end{aligned}$$

Cette formule de quadrature est stable puisque tous les coefficients sont positifs.

Exercice 3.16

Soit f une fonction $\mathscr{C}^\infty(\mathbb{R},\mathbb{R})$. On se donne les points $\{x_i\}_{i=0}^{i=n}$ de subdivision de l'intervalle $[a;b]$: $x_i = a + ih$ avec $h = \frac{b-a}{n}$. Le but de l'exercice est de trouver une formule de quadrature à n points pour approcher l'intégrale définie

$$\int_a^b f(x)\,\mathrm{d}x. \tag{3.11}$$

On propose dans un premier temps (question 1 à 2) de construire la formule de quadrature à deux points :

$$\int_{-1}^{1} g(x)\,\mathrm{d}x \approx \frac{4}{3}g\left(-\frac{w}{2}\right) + \frac{2}{3}g(w), \tag{3.12}$$

où $0 < w \leq 1$ est à déterminer.

1. Déterminer w pour que la formule de quadrature (3.12) soit exacte pour toute fonction g polynomiale de degré $m > 1$ et donner la plus grande valeur de m.
2. À l'aide d'un changement de variable affine, en déduire une formule de quadrature pour l'intégrale suivante :
$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x.$$
3. En déduire une formule de quadrature à $2n$ points, notée F, pour le calcul approché de (3.11). Cette formule de quadrature est-elle stable ?
4. Écrire l'algorithme du calcul de F.

Correction de l'exercice 3.16.

1. On a

k	$p_k(x) = x^k$	$\int_{-1}^{1} p_k(x)\,\mathrm{d}x$	$\frac{2}{3}\left(2p_k\left(-\frac{w}{2}\right) + p_k(w)\right)$	Degré d'exactitude
0	1	2	2	au moins 0
1	x	0	0	au moins 1
2	x^2	2/3	w^2	1 si $w \neq \sqrt{2/3}$, au moins 2 si $w = \sqrt{2/3}$
			Soit $w = \sqrt{2/3}$	
3	x^3	0	$w^3/2$	2

Si $w \neq \sqrt{2/3}$ la formule de quadrature est exacte pour les polynômes de degré au plus 1, si $w = \sqrt{2/3}$ la formule est exacte pour les polynômes de degré au plus 2.

2. Par le changement de variable $y = x_i + (x+1)\frac{x_{i+1}-x_i}{2}$ on déduit la formule de quadrature

$$\begin{aligned}\int_{x_i}^{x_{i+1}} f(y)\,\mathrm{d}y &= \frac{x_{i+1}-x_i}{2}\int_{-1}^{1} f\left(x_i + (x+1)\frac{x_{i+1}-x_i}{2}\right)\mathrm{d}x\\ &\approx \frac{x_{i+1}-x_i}{3}\left[f\left(x_i + (1+\sqrt{\tfrac{2}{3}})\frac{x_{i+1}-x_i}{2}\right) + 2f\left(x_i + (1-\sqrt{\tfrac{1}{6}})\frac{x_{i+1}-x_i}{2}\right)\right].\end{aligned}$$

3. Si $H = x_{i+1} - x_i = \frac{b-a}{n}$ (*i.e.* si on considère une subdivision de l'intervalle $[a;b]$ équirépartie) alors on trouve la formule de quadrature composite (*i.e.* sur n sous-intervalles et à $2n$ points)

$$\begin{aligned}\int_a^b f(x)\,\mathrm{d}x &\approx \frac{H}{3}\sum_{i=0}^{n-1}\left[f\left(x_i + H\left(1+\sqrt{\tfrac{2}{3}}\right)\right) + 2f\left(x_i + H\left(1-\sqrt{\tfrac{1}{6}}\right)\right)\right]\\ &= \frac{H}{3}\sum_{i=0}^{n-1}\left[f\left(a + H\left(i+1+\sqrt{\frac{2}{3}}\right)\right) + 2f\left(a + H\left(i+1-\sqrt{\frac{1}{6}}\right)\right)\right].\end{aligned}$$

Cette formule de quadrature est stable puisque tous les coefficients sont positifs.

4. Algorithme du calcul de F :

Require: a ; $b > a$; $n > 0$; $f : [a;b] \to \mathbb{R}$
$H \leftarrow \frac{b-a}{n}$
$\alpha_1 \leftarrow a + H(1+\sqrt{2/3})$
$\alpha_2 \leftarrow a + H(1-\sqrt{1/6})$
for $i = 0$ to $n-1$ **do**
$s \leftarrow s + f(\alpha_1 + iH) + 2f(\alpha_2 + iH)$
end for
return $I \leftarrow \frac{H}{3}s$

Exercice 3.17

Soit $0 < \alpha \leq 1$ un nombre réel donné et soit $\omega_1, \omega_2, \omega_3$ trois nombres réels. Considérons la formule de quadrature

$$\int_{-1}^{1} g(t)\,\mathrm{d}t \approx \omega_1 g(-\alpha) + \omega_2 g(0) + \omega_3 g(\alpha).$$

1. Calculer $\alpha, \omega_1, \omega_2, \omega_3$ pour que le degré de précision de la formule de quadrature soit de 5.
2. À l'aide d'un changement de variable affine, en déduire une formule de quadrature pour l'intégrale
$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x.$$
3. Soit $h = \frac{b-a}{n}$ et $x_i = a + ih$ pour $i = 0, \ldots, n$. On subdivise l'intervalle $[a;b]$ en n intervalles $[x_i; x_{i+1}]$ de largeur h. En déduire la formule de quadrature composite pour le calcul approché de
$$\int_a^b f(x)\,\mathrm{d}x.$$
4. Écrire l'algorithme associé à cette formule de quadrature.

CORRECTION DE L'EXERCICE 3.17.

1. On a

k	$p_k(x) = x^k$	$\int_{-1}^{1} p_k(x)\,\mathrm{d}x$	$\omega_1 p_k(-\alpha) + \omega_2 p_k(0) + \omega_3 p_k(\alpha)$
0	1	2	$\omega_1 + \omega_2 + \omega_3$
Soit $\omega_2 = 2 - \omega_1 - \omega_3$			
1	x	0	$-\alpha\omega_1 + \alpha\omega_3$
Soit $\omega_3 = \omega_1$ et donc $\omega_2 = 2 - 2\omega_1$			
2	x^2	$\frac{2}{3}$	$2\alpha^2\omega_1$
Soit $\omega_1 = \frac{1}{3\alpha^2}$ et donc $\omega_3 = \frac{1}{3\alpha^2}$ et $\omega_2 = 2 - \frac{2}{3\alpha^2}$			
3	x^3	0	0
4	x^4	$\frac{2}{5}$	$\frac{2}{3}\alpha^2$
Soit $\alpha = \sqrt{\frac{3}{5}}$ et donc $\omega_1 = \omega_3 = \frac{5}{9}$ et $\omega_2 = \frac{8}{9}$			
5	x^5	0	0
6	x^6	$\frac{2}{7}$	$\frac{6}{25}$

Si $\alpha = \sqrt{\frac{3}{5}}$, $\omega_1 = \omega_3 = \frac{5}{9}$ et $\omega_2 = \frac{8}{9}$ alors la formule est exacte pour les polynômes de degré au plus 5 (il s'agit de la formule de GAUSS-LEGENDRE à 3 points).

Remarquons que si on choisit $\alpha = 1$ on retrouve la formule de SIMPSON.

2. Soit $x = mt + q$, alors
$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x = m\int_{-1}^{1} f(mt+q)\,\mathrm{d}t \quad \text{avec} \quad \begin{cases} x_i = -m + q, \\ x_{i+1} = m + q, \end{cases}$$

d'où le changement de variable $x = x_i + (t+1)\frac{x_{i+1}-x_i}{2}$. On déduit la formule de quadrature (exacte sur l'espace des polynôme de degré au plus 5)

$$\begin{aligned}\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x &= \frac{x_{i+1}-x_i}{2}\int_{-1}^{1} f\left(x_i+(t+1)\frac{x_{i+1}-x_i}{2}\right)\mathrm{d}t \\ &\approx \frac{x_{i+1}-x_i}{18}\left[5f\left(x_i+\left(1-\sqrt{\frac{3}{5}}\right)\frac{x_{i+1}-x_i}{2}\right)+8f\left(\frac{x_{i+1}+x_i}{2}\right)+5f\left(x_i+\left(1+\sqrt{\frac{3}{5}}\right)\frac{x_{i+1}-x_i}{2}\right)\right].\end{aligned}$$

3. $h = \frac{b-a}{n} = x_{i+1} - x_i$ pour $i = 0,\dots,n$. On subdivise l'intervalle $[a;b]$ en n intervalles $[x_i;x_{i+1}]$ de largeur h. On trouve ainsi la formule de quadrature composite

$$\begin{aligned}\int_a^b f(x)\,\mathrm{d}x &= \sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x \\ &\approx \sum_{i=0}^{n-1}\frac{x_{i+1}-x_i}{18}\left[5f\left(x_i+\left(1-\sqrt{\frac{3}{5}}\right)\frac{x_{i+1}-x_i}{2}\right)+8f\left(\frac{x_{i+1}+x_i}{2}\right)+5f\left(x_i+\left(1+\sqrt{\frac{3}{5}}\right)\frac{x_{i+1}-x_i}{2}\right)\right] \\ &= \frac{h}{18}\sum_{i=0}^{n-1}\left[5f\left(x_i+\left(1-\sqrt{\frac{3}{5}}\right)h\right)+8f\left(x_i+\frac{h}{2}\right)+5f\left(x_i+\left(1+\sqrt{\frac{3}{5}}\right)h\right)\right] \\ &= \frac{h}{18}\sum_{i=0}^{n-1}\left[5f\left(a+\left(i+1-\sqrt{\frac{3}{5}}\right)h\right)+8f\left(a+\left(i+\frac{1}{2}\right)h\right)+5f\left(a+\left(i+1+\sqrt{\frac{3}{5}}\right)h\right)\right].\end{aligned}$$

4. Algorithme du calcul associé à cette formule de quadrature

Require: $a\,;\,b>a\,;\,n>0\,;\,f\colon[a;b]\to\mathbb{R}$
  $h \leftarrow \frac{b-a}{n}$
  $c_1 \leftarrow a+\left(1-\sqrt{3/5}\right)h$
  $c_2 \leftarrow a+\frac{1}{2}h$
  $c_3 \leftarrow a+\left(1+\sqrt{3/5}\right)h$
  $s \leftarrow 0$
  for $i = 0$ to $n-1$ **do**
    $s \leftarrow s+5f(c_1+ih)+8f(c_2+ih)+5f(c_3+ih)$
  end for
  return $\frac{h}{18}s$

Exercice 3.18 *Interpolation et quadratures*

Soit f une fonction $\mathscr{C}^\infty(\mathbb{R},\mathbb{R})$. On se donne les points $\{x_i\}_{i=0}^{i=n}$ de subdivision uniforme de l'intervalle $[a;b]$ définis par $x_i = a+ih$ avec $h = \frac{b-a}{n}$. Le but de l'exercice est de trouver une formule de quadrature composite pour approcher l'intégrale

$$\int_a^b f(x)\,\mathrm{d}x.$$

1. Écrire le polynôme p qui interpole f aux points 0 et 1.
2. En déduire une formule de quadrature basée sur l'approximation

$$\int_0^1 f(x)\,\mathrm{d}x \approx \int_0^1 p(x)\,\mathrm{d}x$$

et étudier le degré de précision de cette formule de quadrature.
3. À l'aide d'un changement de variable affine, en déduire une formule de quadrature pour l'intégrale

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x.$$

4. En utilisant le résultat au point précédent, proposer une formule de quadrature composite pour le calcul approché de l'intégrale

$$\int_a^b f(x)\,\mathrm{d}x.$$

Quelle méthode de quadrature reconnait-on ?

CORRECTION DE L'EXERCICE 3.18.

1. On a deux points d'interpolation, donc le polynôme d'interpolation est un polynôme de $\mathbb{R}_1[x]$. On cherche alors les coefficients α et β du polynôme $p(x) = \alpha + \beta x$ tels que

$$\begin{cases} f(0) = p(0) = \alpha, \\ f(1) = p(1) = \alpha + \beta. \end{cases}$$

   On obtient $\alpha = f(0)$ et $\beta = f(1) - f(0)$.

2. On en déduit la méthode de quadrature

$$\int_0^1 f(t)\,\mathrm{d}t \approx \int_0^1 p(t)\,\mathrm{d}t = \int_0^1 \alpha + \beta t\,\mathrm{d}t = \alpha + \frac{\beta}{2} = \frac{f(0)+f(1)}{2}.$$

   Par construction, cette formule de quadrature a degré de précision au moins 1. Soit $f(x) = x^2$, alors $\int_0^1 f(x)\,\mathrm{d}x = 1/3$ tandis que $\frac{f(0)+f(1)}{2} = 1/2$: la formule est exacte pour les polynômes de degré au plus 1.

3. Soit $x = mt + q$, alors

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x = m\int_0^1 f(mt+q)\,\mathrm{d}t \quad \text{avec} \quad \begin{cases} x_i = q, \\ x_{i+1} = m + q, \end{cases}$$

   d'où le changement de variable $x = x_i + (x_{i+1} - x_i)t$. On en déduit la formule de quadrature (exacte sur l'espace des polynôme de degré au plus 1)

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x = (x_{i+1} - x_i)\int_0^1 f x_i + (x_{i+1} - x_i)t\,\mathrm{d}t \approx (x_{i+1} - x_i)\frac{f(x_i) + f(x_{i+1})}{2}.$$

4. On subdivise l'intervalle $[a;b]$ en n intervalles $[x_i; x_{i+1}]$ de largeur $h = \frac{b-a}{n} = \frac{x_{i+1}-x_i}{2}$ pour $i = 0, \dots, n$. On trouve ainsi la formule de quadrature composite

$$\begin{aligned} \int_a^b f(x)\,\mathrm{d}x = \sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x \approx \sum_{i=0}^{n-1}(x_{i+1} - x_i)\frac{f(x_i)+f(x_{i+1})}{2} &= \frac{h}{2}\sum_{i=0}^{n-1}\left[f(x_i) + f(x_{i+1})\right] \\ &= \frac{h}{2}\left[f(a) + f(b) + 2\sum_{i=1}^{n-1} f(x_i)\right] = \frac{h}{2}\left[f(a) + f(b) + 2\sum_{i=1}^{n-1} f(a+ih)\right]. \end{aligned}$$

   Il s'agit de la **méthode des trapèzes composite**.

Exercice 3.19 *Interpolation de* LAGRANGE *et quadratures*

Soit f une fonction $\mathscr{C}^\infty(\mathbb{R},\mathbb{R})$. On se donne les points $\{x_i\}_{i=0}^{i=n}$ de subdivision uniforme de l'intervalle $[a;b]$ définis par $x_i = a + ih$ avec $h = \frac{b-a}{n}$. Le but de l'exercice est de trouver une formule de quadrature composite pour approcher l'intégrale

$$\int_a^b f(x)\,\mathrm{d}x.$$

1. Soit p le polynôme de LAGRANGE qui interpole f aux points -1 et 1. Écrire le polynôme p, en déduire une formule de quadrature basée sur l'approximation

$$\int_{-1}^1 f(x)\,\mathrm{d}x \approx \int_{-1}^1 p(x)\,\mathrm{d}x$$

   et étudier le degré de précision de cette formule de quadrature.

2. À l'aide d'un changement de variable affine, en déduire une formule de quadrature pour l'intégrale

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x.$$

3. En utilisant le résultat au point précédent, proposer une formule de quadrature composite pour le calcul approché de l'intégrale

$$\int_a^b f(x)\,\mathrm{d}x.$$

   Quelle méthode de quadrature reconnait-on ?

CORRECTION DE L'EXERCICE 3.19.

1. On a deux points d'interpolation, donc le polynôme interpolateur de LAGRANGE est un polynôme de $\mathbb{R}_1[x]$. On cherche alors les coefficients α et β du polynôme $p(x) = \alpha + \beta x$ tels que

$$\begin{cases} f(-1) = p(-1) = \alpha - \beta, & \text{(3.13a)} \\ f(1) = p(1) = \alpha + \beta. & \text{(3.13b)} \end{cases}$$

   La somme des équations (3.13b)+(3.13a) donne $\alpha = \frac{f(1)+f(-1)}{2}$ et la soustraction des équations (3.13b)−(3.13a) donne $\beta = \frac{f(1)-f(-1)}{2}$. On en déduit la méthode de quadrature

$$\int_{-1}^{1} f(t)\,\mathrm{d}t \approx \int_{-1}^{1} p(t)\,\mathrm{d}t = \int_{-1}^{1} \alpha + \beta t\,\mathrm{d}t = 2\alpha = f(-1) + f(1).$$

   Par construction, cette formule de quadrature a degré de précision au moins 1. Soit $f(x) = x^2$, alors $\int_{-1}^{1} f(x)\,\mathrm{d}x = 2/3$ tandis que $f(-1) + f(1) = 2$: la formule est exacte pour les polynômes de degré au plus 1.

2. Soit $x = mt + q$, alors

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x = m\int_{-1}^{1} f(mt+q)\,\mathrm{d}t \quad \text{avec} \quad \begin{cases} x_i = -m + q, \\ x_{i+1} = m + q, \end{cases}$$

   d'où le changement de variable $x = x_i + (t+1)\frac{x_{i+1}-x_i}{2}$. On en déduit la formule de quadrature (exacte sur l'espace des polynôme de degré au plus 1)

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x = \frac{x_{i+1}-x_i}{2}\int_{-1}^{1} f\left(x_i + (t+1)\frac{x_{i+1}-x_i}{2}\right)\mathrm{d}t \approx \frac{x_{i+1}-x_i}{2}\left[f(x_i) + f(x_{i+1})\right].$$

3. On subdivise l'intervalle $[a;b]$ en n intervalles $[x_i; x_{i+1}]$ de largeur $h = \frac{b-a}{n} = x_{i+1} - x_i$ pour $i = 0, \dots, n-1$. On trouve ainsi la formule de quadrature composite

$$\begin{aligned} \int_a^b f(x)\,\mathrm{d}x = \sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x \approx \sum_{i=0}^{n-1}\frac{x_{i+1}-x_i}{2}\left[f(x_i)+f(x_{i+1})\right] &= \frac{h}{2}\sum_{i=0}^{n-1}\left[f(x_i)+f(x_{i+1})\right] \\ &= \frac{h}{2}\left[f(a)+f(b)+2\sum_{i=1}^{n-1} f(x_i)\right] = \frac{h}{2}\left[f(a)+f(b)+2\sum_{i=1}^{n-1} f(a+ih)\right]. \end{aligned}$$

   Il s'agit de la **méthode des trapèzes composite**.

Exercice 3.20 *Interpolation d'*HERMITE *et quadratures*

Soit f une fonction de classe $\mathscr{C}^1([-1,1])$ et p le polynôme interpolateur d'HERMITE (de degré ≤ 3) de f vérifiant

$$p(-1) = f(-1), \qquad p'(-1) = f'(-1), \qquad p(1) = f(1), \qquad p'(1) = f'(1).$$

1. Écrire le polynôme p.
2. En déduire la méthode d'intégration numérique élémentaire

$$\int_{-1}^{1} f(s)\,\mathrm{d}s \approx f(-1) + f(1) + \frac{1}{3}\left(f'(-1) - f'(1)\right).$$

3. Connaissant la formule sur $[-1;1]$, en déduire la formule de quadrature des trapèzes-HERMITE sur l'intervalle $[a;b]$ par exemple grâce au changement de variable $y = a + (x+1)\frac{b-a}{2}$.

CORRECTION DE L'EXERCICE 3.20.

1. Cf. exercice 2.13.
2. En intégrant le polynôme ainsi trouvé on en déduit

$$\begin{aligned} \int_{-1}^{1} p(x)\,\mathrm{d}x &= \left[\alpha x + \frac{\beta}{2}x^2 + \frac{\gamma}{3}x^3 + \frac{\delta}{4}x^4\right]_{-1}^{1} \\ &= 2\alpha + \frac{2}{3}\gamma = \frac{2f(-1)+2f(1)+f'(-1)-f'(1)}{2} + \frac{-f'(-1)+f'(1)}{6} \end{aligned}$$

$$= \frac{6f(-1)+6f(1)+3f'(-1)-3f'(1)-f'(-1)+f'(1)}{6}$$
$$= f(-1)+f(1)+\frac{1}{3}(f'(-1)-f'(1)).$$

Remarque : la formule est au moins exacte de degré 3 par construction. Elle n'est pas exacte de degré supérieure à 3 car si $f(x)=x^4$ alors

$$\int_{-1}^{1} f(x)\,\mathrm{d}x = \left[\frac{1}{5}x^5\right]_{-1}^{1} = \frac{2}{5} = \frac{6}{15}$$
$$\neq$$
$$f(-1)+f(1)+\frac{1}{3}(f'(-1)-f'(1)) = 1+1+\frac{1}{3}(4+4) = \frac{14}{3} = \frac{70}{15}$$

3. Connaissant la formule sur $[-1;1]$, on en déduit la formule sur un intervalle $[a;b]$ quelconque par le changement de variable $y = a+(x+1)\frac{b-a}{2}$ qui donne [2]

$$\begin{aligned}\int_a^b f(y)\mathrm{d}y &= \frac{b-a}{2}\int_{-1}^{1} f\left(a+(x+1)\frac{b-a}{2}\right)\mathrm{d}x\\ &= \frac{b-a}{2}\left[f(a)+f(b)+\frac{b-a}{6}(f'(a)-f'(b))\right]\\ &= \frac{b-a}{2}(f(a)+f(b))+\frac{(b-a)^2}{12}(f'(a)-f'(b)).\end{aligned}$$

Exercice 3.21

Soit f une fonction $\mathscr{C}^\infty([0;1],\mathbb{R})$. On se donne les points $\{x_i\}_{i=0}^{i=n}$ de subdivision de l'intervalle $[0;1]$: $x_i = ih$ avec $h=\frac{1}{n}$. Le but de l'exercice est de trouver une formule de quadrature pour approcher

$$\int_0^1 f(x)\,\mathrm{d}x. \tag{3.14}$$

1. Soit i un entier fixé ($1 \le i \le n-1$). Trouver m_i un point du segment $[x_i;x_{i+1}]$ et a, b et c trois coefficients réels tels que la formule de quadrature suivante, sur l'intervalle $[x_i;x_{i+1}]$, soit exacte pour p un polynôme de degré le plus haut possible :
$$\int_{x_i}^{x_{i+1}} p(x)\,\mathrm{d}x = ap(x_i)+bp(m_i)+cp(x_{i+1}).$$
2. En déduire en fonction de a, b et c la formule de quadrature $Q(f)$
$$Q(f) = \sum_{i=0}^{n}\alpha_i f(x_i)+\sum_{i=0}^{n-1}\beta_i f(m_i)$$
pour le calcul approché de 3.14 construite sur la formule de quadrature précédente pour chaque intervalle du type $[x_i;x_{i+1}]$. Cette formule de quadrature est-elle stable ?
3. On rappelle que si p interpole f en k points $y_1<y_2<\cdots<y_k$, on a l'estimation d'erreur
$$\forall x\in[y_1;y_k],\qquad |f(x)-p(x)| \le \frac{\sup_{\xi\in[y_1;y_k]}|f^{(k)}(\xi)|}{k!}\prod_{j=1}^{k}(x-y_j).$$
En déduire une estimation de l'erreur de quadrature entre (3.14) et Q
$$E(h) = \int_0^1 f(x)\,\mathrm{d}x - Q(f).$$
La dépendance en h dans cette estimation d'erreur est-elle optimale ?
4. Écrire l'algorithme qui calcule $Q(f)$.

Correction de l'exercice 3.21.

2. Rappel : si $y = a+(x+1)\frac{b-a}{2}$ alors $\mathrm{d}y = \frac{b-a}{2}\mathrm{d}x$ et $f'(y) = \frac{b-a}{2}f'(x)$.

1. Pour simplifier le calcul, on se ramène à l'intervalle $[0;1]$. Soit x un élément de l'intervalle $[x_i;x_{i+1}]$ et y un élément de l'intervalle $[0;1]$. On transforme l'intervalle $[x_i;x_{i+1}]$ dans l'intervalle $[0;1]$ par le changement de variable affine $y = \frac{1}{x_{i+1}-x_i}x - \frac{x_i}{x_{i+1}-x_i}$. On note $h = x_{i+1} - x_i$. Alors $y = \frac{x-x_i}{h}$ et on a $\int_0^1 f(y)\,\mathrm{d}y = \frac{1}{h}\int_{x_i}^{x_{i+1}} f(\frac{x-x_i}{h})dx$. Comme $\int_{x_i}^{x_{i+1}} f(t)\,\mathrm{d}t \approx af(x_i) + bf(m_i) + cf(x_{i+1})$, alors $\int_0^1 f(y)\,\mathrm{d}y = \frac{1}{h}\int_{x_i}^{x_{i+1}} f(\frac{x-x_i}{h})\,\mathrm{d}x \approx \frac{1}{h}\left(af(0) + bf(\frac{m_i-x_i}{h}) + cf(1)\right)$. On note alors $A = \frac{a}{h}$, $B = \frac{b}{h}$, $C = \frac{c}{h}$, $M = \frac{m_i-x_i}{h}$ d'où $m_i = (1-M)x_i + Mx_{i+1}$. Rechercher a, b, c et m_i revient à chercher A, B, C et M avec

$$\begin{cases} m_i = (1-M)x_i + Mx_{i+1}, \\ a = Ah, \\ b = Bh, \\ c = Ch \end{cases}$$

tels que

$$\int_0^1 p(x)\,\mathrm{d}x = Ap(0) + Bp(M) + Cp(1),$$

où $p(x)$ est un polynôme. Si $p \in \mathbb{P}^3$ (*i.e.* si $p(x) = d_0 + d_1x + d_2x^2 + d_3x^3$) on a

$$\begin{array}{ccc} \int_0^1 p(x)\,\mathrm{d}x & = & Ap(0) + Bp(M) + Cp(1) \\ \| & & \| \\ \left[d_0x + \frac{d_1}{2}x^2 + \frac{d_2}{3}x^3 + \frac{d_3}{4}x^4\right]_0^1 & & Ad_0 + Bd_0 + Cd_0 \\ \| & & \| \\ d_0 + \frac{d_1}{2} + \frac{d_2}{3} + \frac{d_3}{4} & & (A+B+C)d_0 + (BM+C)d_1 + (BM^2+C)d_2 + (BM^3+C)d_3 \end{array}$$

Par conséquent, pour que la formule soit exacte de degré au moins 3 il faut que

$$\begin{cases} A+B+C = 1 \\ BM+C = \frac{1}{2} \\ BM^2+C = \frac{1}{3} \\ BM^3+C = \frac{1}{4} \end{cases} \iff \begin{cases} A+B+C = 1 \\ BM = \frac{1}{2} - C \\ (\frac{1}{2}-C)M = \frac{1}{3} - C \\ (\frac{1}{3}-C)M = \frac{1}{4} - C \end{cases} \iff \begin{cases} A = \frac{1}{6}, \\ B = \frac{2}{3}, \\ C = \frac{1}{6}, \\ M = \frac{1}{2}. \end{cases}$$

La méthode

$$\int_0^1 f(x)\,\mathrm{d}x = \frac{1}{6}f(0) + \frac{2}{3}f\left(\frac{1}{2}\right) + \frac{1}{6}f(1),$$

est exacte pour tout polynôme de degré au moins 3.
Soit maintenant $f(x) = x^4$. On a

$$\int_0^1 f(x)\,\mathrm{d}x = \left[\frac{x^5}{5}\right]_0^1 = \frac{1}{5}$$

mais

$$\frac{1}{6}f(0) + \frac{2}{3}f\left(\frac{1}{2}\right) + \frac{1}{6}f(1) = \frac{1}{6} + \frac{2}{3}\left(\frac{1}{2}\right)^4 + \frac{1}{6} = \frac{5}{24},$$

donc la formule de quadrature est exacte de degré 3.
Si on revient aux variables initiales, on trouve

$$\begin{cases} m_i = \frac{1}{2}x_i + \frac{1}{2}x_{i+1}, \\ a = \frac{1}{6}h, \\ b = \frac{2}{3}h, \\ c = \frac{1}{6}h \end{cases}$$

2. L'intégrale

$$\int_0^1 f(x)\mathrm{d}x = \sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x$$

peut être calculée numériquement en utilisant la formule précédente pour approcher chaque intégrale

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x \approx \frac{h}{6}\left[f(x_i) + 4f\left(\frac{x_i+x_{i+1}}{2}\right) + f(x_{i+1})\right].$$

On obtient ainsi

$$\begin{aligned}
\int_0^1 f(x)\mathrm{d}x &= \sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x \\
&\approx \sum_{i=0}^{n-1}\frac{h}{6}\left[f(x_i)+4f\left(\frac{x_i+x_{i+1}}{2}\right)+f(x_{i+1})\right] \\
&= \frac{h}{6}\left[\sum_{i=0}^{n-1}f(x_i)+\sum_{i=0}^{n-1}f(x_{i+1})+4\sum_{i=0}^{n-1}f\left(\frac{x_i+x_{i+1}}{2}\right)\right] \\
&= \frac{h}{6}\left[f(a)+f(b)+2\sum_{i=1}^{n-1}f(x_i)+4\sum_{i=0}^{n-1}f\left(\frac{x_i+x_{i+1}}{2}\right)\right] \\
&= \sum_{i=0}^{n}\alpha_i f(x_i)+\sum_{i=0}^{n-1}\beta_i f(m_i) = Q(f) \qquad \text{avec} \quad \beta_i=\frac{2h}{3}, \quad \alpha_i=\begin{cases}\frac{h}{3} & \text{si } i=1,\dots,n-1,\\ \frac{h}{6} & \text{sinon.}\end{cases}
\end{aligned}$$

Cette formule de quadrature est stable puisque tous les coefficients α_i et β_i sont positifs et on a

$$\sum_{i=0}^{n}\alpha_i+\sum_{i=0}^{n-1}\beta_i=\frac{h}{6}+\sum_{i=1}^{n-1}\frac{h}{3}+\frac{h}{6}+\sum_{i=0}^{n-1}\frac{2h}{3}=\frac{1}{n}\left(\frac{1}{6}+\frac{1}{3}\sum_{i=1}^{n-1}1+\frac{1}{6}+\frac{2}{3}\sum_{i=0}^{n-1}1\right)=1.$$

3. On reconnait la formule de Cavalieri-Simpson : remarquons alors que $Q(f)=\sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}}p(x)\mathrm{d}x$ avec p le polynôme qui interpole $(x_i,f(x_i))$, $(m_i,f(m_i))$ et $(x_{i+1},f(x_{i+1}))$. Par conséquent l'erreur de quadrature entre (3.14) et Q est

$$\begin{aligned}
|E(h)| &= \left|\int_0^1 f(x)\,\mathrm{d}x - Q(f)\right| \\
&= \left|\sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x - \sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} p(x)\mathrm{d}x\right| \\
&\le \sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} \left|f(x)-p(x)\right|\mathrm{d}x \\
&\le \sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} \frac{\sup_{\xi\in[x_i;x_{i+1}]}|f'''(\xi)|}{6}(x-x_i)(x-m_i)(x-x_{i+1})\mathrm{d}x \\
&\le Dh^4.
\end{aligned}$$

4. Algorithme

Require: $x\mapsto f$
Require: $n>0$
  $a\leftarrow\frac{1}{6n}$
  $b\leftarrow\frac{2}{3n}$
  $c\leftarrow\frac{1}{6n}$
  $I\leftarrow af(0)$
  for $i=1$ to $n-1$ **do**
    $I\leftarrow I+(a+c)f\left(\frac{i}{n}\right)+bf\left(\frac{i-\frac{1}{2}}{n}\right)$
  end for
  return $I\leftarrow I+cf(1)+bf\left(\frac{n-\frac{1}{2}}{n}\right)$

4. Équations différentielles ordinaires

Calculer la fonction $t \mapsto y(t)$ qui vérifie l'EDO $y'(t) = \varphi(t, y(t))$ et la condition $y(t_0) = y_0$

Les équations différentielles décrivent l'évolution de nombreux phénomènes dans des domaines variés. Une équation différentielle est une équation impliquant une ou plusieurs dérivées d'une fonction inconnue. Si toutes les dérivées sont prises par rapport à une seule variable, on parle d'équation différentielle ordinaire. Une équation mettant en jeu des dérivées partielles est appelée équation aux dérivées partielles.

4.1. Généralités

Une équation différentielle (EDO) est une équation exprimée sous la forme d'une relation

$$F(y, y', y'', \ldots, y^{(n)}) = g(t)$$

- ⋆ dont l'inconnue est une fonction $y\colon I \subset \mathbb{R} \to \mathbb{R}$ définie sur un intervalle I (à déterminer)
- ⋆ dans laquelle cohabitent à la fois y et ses dérivées y', y'', …, $y^{(p)}$ (p est appelé l'ordre de l'équation).

Si la fonction g, appelée «second membre» de l'équation, est nulle, on dit que l'équation en question est homogène.

Nous pouvons nous limiter aux équations différentielles du premier ordre, car une équation d'ordre $p > 1$ peut toujours se ramener à un système de p équations d'ordre 1.

4.1.1. Position du problème

Résoudre une équation c'est chercher toutes les valeurs de l'inconnue qui satisfont l'égalité. Dans les équations rencontrées jusqu'à présent, les inconnues étaient des nombres. Par exemple, résoudre l'équation $2x + 4 = 10$ signifie chercher toutes les valeurs de $x \in \mathbb{R}$ telles que $2x + 4 = 10$.

Dans les équations différentielles, les inconnues sont des fonctions. *Résoudre une équation différentielle*, c'est chercher toutes les fonctions, définies sur un intervalle $I \subset \mathbb{R}$, qui satisfont l'équation (on dit aussi intégrer l'équation différentielle).

Exemple

Résoudre l'équation différentielle $y'(t) = -y(t)$ signifie chercher toutes les fonctions

$$\begin{aligned} y\colon I \subset \mathbb{R} &\to \mathbb{R} \\ t &\mapsto y = f(t) \end{aligned}$$

telles que $f'(t) = -f(t)$ pour tout $t \in I$. On peut vérifier que $y(t) = 0$ pour tout $t \in \mathbb{R}$ est une solution de l'EDO mais aussi $y(t) = ce^{-ct}$ pour tout $t \in \mathbb{R}$ (où c est constante réelle quelconque).

4.1.2. Condition initiale

Une EDO admet généralement une infinité de solutions. Pour en sélectionner quelques unes (parfois juste une), on doit imposer une condition supplémentaire qui correspond à la valeur prise par la solution en un point de l'intervalle d'intégration.

Définition *Condition initiale*

Une condition initiale (CI) est une relation du type $y(t_0) = y_0$ qui impose en t_0 la valeur y_0 de la fonction inconnue.

En pratique, se donner une CI revient à se donner le point (t_0, y_0) par lequel doit passer le graphe de la fonction solution.

Exemple *Évolution d'une population-1*

Soit N le nombre d'individu d'une population à l'instant t. La population N a un taux de naissance saisonnier ; le taux de décès est proportionnel au nombre d'individu au carré (par surpopulation, dus par exemple au manque de nourriture). On considère enfin un terme indépendant de la taille et du temps (par exemple, si cette EDO modélise l'élevage de saumons, ce terme représente les

saumons péchés). On a alors l'équation différentielle

$$N' = (2-\cos(t))N - \frac{1}{2}N^2 - 1.$$

On aura donc deux types de questions :
1. trouver toutes les solutions de l'EDO ;
2. trouver la ou les solutions qui vérifient une CI

4.1.3. Représentation graphique

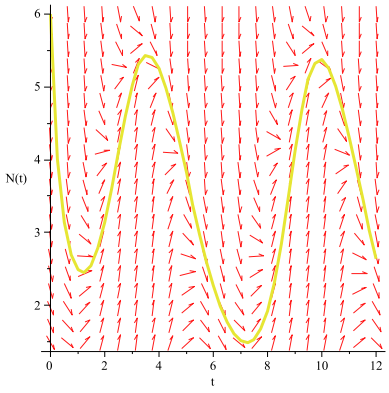
On va expliquer comment tracer l'allure des solutions d'une EDO du type

$$y' = \varphi(t, y).$$

Soit $y = f(t)$ la fonction inconnue solution de cette EDO. Si (t, y) est un point du graphe de f, cette égalité dit que la tangente au graphe de f au point (t, y) a pour pente $\varphi(t, y)$. Dessinons alors, en (presque) chaque point (t, y) du plan un vecteur $V_{t,y}$ de pente $\varphi(t, y)$: le graphe de f est tangent en chaque point (t, y) au vecteur $V_{t,y}$. Remarquer qu'on n'a pas besoin d'avoir résolu l'équation (analytiquement) pour pouvoir dessiner le champ de tangentes, et ceci permet parfois d'avoir une idée du comportement des solutions.

Exemple *Évolution d'une population-2*

Considérons à nouveau l'exemple de l'évolution d'une population traçons l'allure des solutions. Si on démarre l'élevage avec 6 saumons, on voit qu'une et une seule courbe passe par le point $(0,6)$ et si on suit cette solution on peut prédire par exemple le nombre d'individu de la population dans dix ans : la courbe tracée en jaune donne $N(10) \approx 5.5$.



4.1.4. Théorème d'existence et unicité, intervalle de vie et solution maximale

Le couple EDO-CI porte le nom de problème de CAUCHY :

Définition *Problème de* CAUCHY

Soit $\varphi\colon \mathbb{R}\times\mathbb{R}\to\mathbb{R}$ une fonction donnée et y' la dérivée de y par rapport à t. On appelle *problème de* CAUCHY le problème trouver une fonction $y\colon I\subset\mathbb{R}\to\mathbb{R}$ définie sur un intervalle I telle que

$$\begin{cases} y'(t) = \varphi(t, y(t)), & \forall t \in I, \\ y(t_0) = y_0, \end{cases} \tag{4.1}$$

avec t_0 un point de I et y_0 une valeur donnée.

Exemple *Existence et unicité sur* $\mathbb{R}$ *de la solution d'un problème de* CAUCHY

On se donne $\varphi(t, y(t)) = 3t - 3y(t)$ et $y_0 = \alpha$ (un nombre quelconque). On cherche une fonction $y\colon t\in\mathbb{R}^+ \mapsto y(t)\in\mathbb{R}$ qui satisfait

$$\begin{cases} y'(t) = 3t - 3y(t), & \forall t > 0, \\ y(0) = \alpha. \end{cases}$$

Sa solution, définie sur $\mathbb{R}$, est donnée par $y(t) = (\alpha + 1/3)e^{-3t} + t - 1/3$. En effet on a bien

$$y(0) = (\alpha+1/3)e^0 + 0 - 1/3 = \alpha, \qquad y'(t) = -3(\alpha+1/3)e^{-3t} + 1 = -3(\alpha+1/3)e^{-3t} + 1 - 3t + 3t = -3y(t) + 3t.$$

Cet exemple montre le cas où il existe une et une seule solution du problème de CAUCHY définie sur $\mathbb{R}$. Les choses ne se passent pas toujours si bien. Les exemples ci-dessous montrent que l'étude mathématique de l'existence et de l'unicité des solutions d'un problème de CAUCHY peut être une affaire délicate.

Exemple *Existence et unicité sur $I \subset \mathbb{R}$ (mais non existence sur $\mathbb{R}$) de la solution d'un problème de* CAUCHY

On se donne $\varphi(t, y(t)) = (y(t))^3$ et $y_0 = 1$. On cherche une fonction $y\colon t \in \mathbb{R}^+ \mapsto y(t) \in \mathbb{R}$ qui satisfait

$$\begin{cases} y'(t) = (y(t))^3, & \forall t > 0, \\ y(0) = 1. \end{cases}$$

On vérifie que la solution y est donnée par $y(t) = \frac{1}{\sqrt{1-2t}}$ qui n'est définie que pour $t \in [0;1/2[$. Cet exemple montre qu'*un problème de* CAUCHY *n'a pas toujours une solution pour tout* $t \in [0;+\infty[$ puisqu'ici la solution explose lorsque t tend vers la valeur $1/2$ (en effet, nous avons $\lim_{t \to (1/2)^-} y(t) = +\infty$) : le graphe de la solution a une asymptote verticale en $t = 1/2$. On parle d'explosion de la solution en temps fini ou encore de barrière.

On verra que ceci est un phénomène général : pour une solution d'une EDO, la seule façon de ne pas être définie sur $\mathbb{R}$ est d'avoir un asymptote verticale.

Exemple *Non unicité de la solution d'un problème de* CAUCHY

On se donne $\varphi(t, y(t)) = \sqrt[3]{y(t)}$ et $y_0 = 0$. On cherche une fonction $y\colon t \in \mathbb{R}^+ \mapsto y(t) \in \mathbb{R}$ qui satisfait

$$\begin{cases} y'(t) = \sqrt[3]{y(t)}, & \forall t > 0, \\ y(0) = 0. \end{cases}$$

On vérifie que les fonctions $y_1(t) = 0$ et $y_{2,3}(t) = \pm\sqrt{8t^3/27}$, pour tout $t \geq 0$, sont toutes les trois solution du problème de CAUCHY donné. Cet exemple montre qu'*un problème de* CAUCHY *n'a pas nécessairement de solution unique.*

Il y a un résultat qui garantit que, sous certaines hypothèses très générales, deux graphes de fonctions qui sont des solutions de la même EDO ne se rencontrent jamais. Le théorème garantit aussi l'existence des solutions ; pour donner un énoncé précis, il faut d'abord définir la notion de solution maximale.

De façon générale, lorsqu'on se donne une équation différentielle et une condition initiale $y(t_0) = y_0$, on cherche un intervalle I, contenant t_0, sur lequel une solution existe, et qui soit «le plus grand possible» : il n'existe pas d'intervalle plus grand sur lequel l'équation différentielle ait une solution. Cet intervalle s'appelle intervalle de vie de la solution. Une solution définie sur cet intervalle le plus grand possible s'appelle solution maximale.

Définition *Solution maximale*

On se donne une équation différentielle $y'(t) = \varphi(t, y(t))$ avec une condition initiale $y(t_0) = y_0$. Une *solution maximale* pour ce problème est une fonction $y = f(t)$, définie sur un intervalle I appelé intervalle de vie, telle que

- f est solution de l'équation différentielle et vérifie la condition initiale ;
- il n'existe pas de solution $\tilde{f}$ de la même équation, vérifiant la même condition initiale et définie sur un intervalle J contenant I et plus grand que I.

Dans ce chapitre, nous nous contentons de rappeler un résultat d'existence et d'unicité global, au sens où on peut intégrer le problème de CAUCHY jusqu'à $t = \infty$.

Théorème *de* CAUCHY-LIPSCHITZ, *Existence et unicité des solutions*

Considérons une fonction $(x, y) \mapsto \varphi(x, y)$

- définie pour tout t dans un intervalle I et pour tout y dans un intervalle J
- de classe $\mathscr{C}^1$

alors pour toute CI $y(t_0) = y_0$ avec $t_0 \in I$ et $y_0 \in J$ il existe une unique solution maximale $y = y(t)$ de l'EDO $y'(t) = \varphi(t, y(t))$.

Attention

La fonction φ est une fonction de deux variables donc vérifier que φ, $\partial_t\varphi$ et $\partial_y\varphi$ sont continues signifie utiliser la notion de limite en deux variables. Les limites unilatérales (*i.e.* de la gauche et de la droite) perdent leur sens et sont remplacées par les nombreuses limites directionnelles possibles. En effet, dès que le domaine se situe dans un espace à deux dimensions au moins, les chemins qui mènent à un point donné peuvent suivre divers axes. Ainsi, l'ensemble des points en lesquels une limite peut être considérée, doit être défini en tenant compte de toutes les possibilités d'accès.

Les fonctions continues de plusieurs variables jouissent des mêmes propriétés que les fonctions continues d'une seule variable. Les fonctions élémentaires telles que les polynômes, les fonctions exponentielles, logarithmiques et trigonométriques sont continues dans leurs domaines de définition respectifs. La continuité des autres fonctions s'établit, le cas échéant, en tant que somme, produit, composée, le quotient (lorsque le dénominateur ne s'annule pas) etc., de fonctions

continues. Voici quelques exemples :
1. $\varphi(x,y) = x^2 + y^2 - xy + y$ est continue dans $\mathbb{R}^2$ (polynôme du second degré à deux variables).
2. $\varphi(x,y,z) = e^y + xy^2$ est continue dans $\mathbb{R}^2$ (somme d'une exponentielle et d'un polynôme).
3. $\varphi(x,y) = \ln(x+y^2) - 3$ est continue dans $\{(x,y) \in \mathbb{R}^2 \mid x + y^2 > 0\}$ comme somme du logarithme d'un polynôme (fonction composée) et d'une constante.

Théorème

Soit $y = f(t)$ une solution maximale définie sur un intervalle de vie $I =]a;b[$. Si $b \neq +\infty$ alors $\lim_{t \to b-} y(t) = \infty$, *i.e.* le graphe de la solution a une asymptote verticale en $t = b$. Même chose si $a \neq -\infty$.

On utilise souvent le théorème sous forme contraposée : si les solutions ne peuvent pas «exploser», alors elles sont définies sur $\mathbb{R}$.

Remarque *Applications du théorème de* CAUCHY-LIPSCHITZ

D'un point de vue pratique, cet énoncé nous aidera à faire des dessins, en garantissant que les graphes des solutions ne se rencontrent jamais. On peut en déduire quelques remarques plus subtiles :
- si l'EDO admet comme solution la solution nulle mais $y_0 \neq 0$, alors la solution du problème de CAUCHY est du signe de y_0 pour tout $t \in I$;
- si l'EDO admet deux solutions constantes $y(t) = \kappa_1$ et $y(t) = \kappa_2$ pour tout $t \in I$ et $y_0 \in]\kappa_1;\kappa_2[$, alors la solution du problème de CAUCHY vérifie $y(t) \in]\kappa_1;\kappa_2[$ pour tout $t \in \mathbb{R}$.

4.2. Schémas numériques

En pratique, on ne peut expliciter les solutions que pour des équations différentielles ordinaires très particulières. Dans certains cas, on ne peut exprimer la solution que sous forme implicite. Dans d'autres cas, on ne parvient même pas à représenter la solution sous forme implicite. Pour ces raisons, on cherche des méthodes numériques capables d'approcher la solution de toutes les équations différentielles qui admettent une solution.

Considérons le problème de CAUCHY (4.1) et supposons que l'on ait montré l'existence d'une solution y. Le principe des méthodes numériques est de subdiviser l'intervalle $I = [t_0, T]$, avec $T < +\infty$, en N_h intervalles de longueur $h = (T - t_0)/N_h = t_{n+1} - t_n$; h est appelé le pas de discrétisation. Si nous intégrons l'EDO $y'(t) = \varphi(t, y(t))$ entre t_n et t_{n+1} nous obtenons

$$y(t_{n+1}) - y(t_n) = \int_{t_n}^{t_{n+1}} \varphi(t, y(t))\, dt.$$

Pour chaque nœud $t_n = t_0 + nh$ $(1 \leq n \leq N_h)$ on cherche la valeur inconnue u_n qui approche $y(t_n)$. L'ensemble des valeurs $\{u_0 = y_0, u_1, \ldots, u_{N_h}\}$ représente la solution numérique.

4.2.1. Schémas numériques classiques

On peut construire différentes schémas selon la formule de quadrature utilisée pour approcher le membre de droite. Les schémas qu'on va construire permettent de calculer u_{n+1} à partir de u_n et il est donc possible de calculer successivement u_1, u_2,..., en partant de u_0 par une formule de récurrence de la forme

$$\begin{cases} u_0 = y_0, \\ u_{n+1} = \varphi(u_n), \quad \forall n \in \mathbb{N}. \end{cases}$$

- Si on utilise la formule de quadrature du rectangle à gauche, *i.e.*

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t))\, dt \approx h\varphi(t_n, y(t_n))$$

on obtient le ***schéma d'*EULER *progressif***

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} = u_n + h\varphi(t_n, u_n) \quad n = 0, 1, 2, \ldots N_h - 1 \end{cases}$$

Il s'agit d'un schéma explicite car il permet d'expliciter u_{n+1} en fonction de u_n.

⋆ Si on utilise la formule de quadrature du rectangle à droite, *i.e.*

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t))\,dt \approx h\varphi(t_{n+1}, y(t_{n+1}))$$

on obtient le ***schéma d'*EULER *rétrograde***

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} - h\varphi(t_{n+1}, u_{n+1}) = u_n \quad n = 0, 1, 2, \dots N_h - 1 \end{cases}$$

Il s'agit d'un schéma implicite car il ne permet pas d'expliciter directement u_{n+1} en fonction de u_n lorsque la fonction f n'est pas triviale.

⋆ Si on utilise la formule de quadrature du point du milieu, *i.e.*

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t))\,dt \approx h\varphi\left(t_n + \frac{h}{2}, y\left(t_n + \frac{h}{2}\right)\right)$$

on obtient un nouveau schéma :

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} = u_n + h\varphi\left(t_n + \frac{h}{2}, u_{n+1/2}\right) \quad n = 0, 1, 2, \dots N_h - 1 \end{cases}$$

où $u_{n+1/2}$ est une approximation de $y(t_n + h/2)$. Nous pouvons utiliser une prédiction d'EULER progressive pour approcher le $u_{n+1/2}$ dans le terme $\varphi(t_n + h/2, u_{n+1/2})$ par $\tilde{u}_{n+1/2} = u_n + (h/2)\varphi(t_n, u_n)$. Nous avons construit ainsi un nouveau schéma appelé ***schéma d'*EULER *modifié*** qui s'écrit

$$\begin{cases} u_0 = y(t_0) = y_0, \\ \tilde{u}_{n+1/2} = u_n + (h/2)\varphi(t_n, u_n), \\ u_{n+1} = u_n + h\varphi\left(t_n + \frac{h}{2}, \tilde{u}_{n+1/2}\right) \quad n = 0, 1, 2, \dots N_h - 1 \end{cases}$$

Il s'agit d'un schéma explicite car il permet d'expliciter u_{n+1} en fonction de u_n.
Si on utilise la formule de quadrature du point milieu sur l'intervalle $[t_n; t_{n+2}]$, i.e.

$$\int_{t_n}^{t_{n+2}} \varphi(t, y(t))\,dt \approx 2h\varphi\left(t_{n+1}, y(t_{n+1})\right)$$

on obtient

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 = \tilde{y}(t_1) \\ u_{n+1} = u_{n-1} + 2h\varphi(t_n, u_n) \quad n = 1, 2, \dots N_h - 1 \end{cases}$$

*où u_1 est une approximation de $y(t_1)$. Nous pouvons utiliser une prédiction d'*EULER *progressive pour approcher u_1. Nous avons construit ainsi un nouveau schéma appelé* **schéma du point milieu** *qui s'écrit*

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 = u_0 + h\varphi(t_0, u_0), \\ u_{n+1} = u_{n-1} + 2h\varphi(t_n, u_n) \quad n = 1, 2, \dots N_h - 1 \end{cases}$$

Il s'agit d'un schéma explicite car il permet d'expliciter u_{n+1} en fonction de u_n et de u_{n-1}.

⋆ Si on utilise la formule du trapèze, *i.e.*

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t))\,dt \approx \frac{h}{2}\left(\varphi(t_n, y(t_n)) + \varphi(t_{n+1}, y(t_{n+1}))\right)$$

on obtient le ***schéma du trapèze ou de* CRANK-NICOLSON**

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} - \frac{h}{2}\varphi(t_{n+1}, u_{n+1}) = u_n + \frac{h}{2}\varphi(t_n, u_n) \quad n = 0, 1, 2, \dots N_h - 1 \end{cases}$$

Il s'agit à nouveau d'un schéma implicite car il ne permet pas d'expliciter directement u_{n+1} en fonction de u_n lorsque la fonction f n'est pas triviale. En fait, ce schéma fait la moyenne des schémas d'EULER progressif et rétrograde.

⋆ Pour éviter le calcul implicite de u_{n+1} dans le schéma du trapèze, nous pouvons utiliser une prédiction d'EULER progressive et remplacer le u_{n+1} dans le terme $\varphi(t_{n+1},u_{n+1})$ par $\tilde{u}_{n+1}=u_n+h\varphi(t_n,u_n)$. Nous avons construit ainsi un nouveau schéma appelé ***schéma de* HEUN**. Plus précisément, la méthode de HEUN s'écrit

$$\begin{cases} u_0=y(t_0)=y_0,\\ \tilde{u}_{n+1}=u_n+h\varphi(t_n,u_n),\\ u_{n+1}=u_n+\frac{h}{2}\left(\varphi(t_n,u_n)+\varphi(t_{n+1},\tilde{u}_{n+1})\right) \quad n=0,1,2,\dots N_h-1 \end{cases}$$

Remarque

Considérons le schéma d'EULER rétrograde. Si nous voulons calculer u_{n+1}, nous définissons la fonction

$$g(x)=x-h\varphi(t_{n+1},x)-u_n$$

et nous cherchons un zéro de $g(x)$ en prenant par exemple la méthode de NEWTON. Ainsi nous pouvons poser $x_0=u_0$ et $x_{m+1}=x_m-g(x_m)/g'(x_m)$, $m=0,1,\dots$. Puisque $g'(x)=1-h\partial_x\varphi(t_{n+1},x)$, nous obtenons donc dans ce cas le schéma

$$\begin{cases} x_0=u_n,\\ x_{m+1}=x_m-\frac{x_m-h\varphi(t_{n+1},x)-u_n}{1-h\partial_x\varphi(t_{n+1},x)} \quad m=0,1,2,\dots \end{cases}$$

et $u_{n+1}=\lim_{m\to\infty}x_m$ pour autant que f soit suffisamment régulière et que x_0 soit suffisamment proche de u_{n+1}, ce qui est le cas si le pas h est suffisamment petit.

4.2.2. Schémas numériques d'Adams

Les schémas d'ADAM approchent l'intégrale $\int_{t_n}^{t_{n+1}}\varphi(t,y(t))dt$ par l'intégrale d'un polynôme p interpolant f en des points donnés. On peut construire différentes schémas selon les points d'interpolation choisis. Ils se divisent en deux familles : les méthodes d'ADAMS-BASHFORTH qui sont explicites et les méthodes d'ADAMS-MOULTON qui sont implicites. Voici quelques exemples :

Méthodes d'Adams-Bashforth : il s'agit de méthodes explicites à j pas notées AB_j. Le polynôme p interpole f en les points $\{t_n,t_{n-1},\dots,t_{n-j+1}\}$ où $j\geq 0$ est fixé. Elles permettent de calculer u_{n+1} à partir de l'ensemble $\{u_n,u_{n-1},\dots,u_{n-j+1}\}$ et il est donc possible de calculer successivement u_j, u_{j+1},..., en partant de $u_0,u_1,\dots,u_{j-1}$ (qui doivent donc être initialisés par des approximations adéquates car seul u_0 est donné).

⋆ $\boxed{j=1}$ On a

$$p(t)=\varphi(t_n,y(t_n))$$
$$\int_{t_n}^{t_{n+1}}p(t)\,\mathrm{d}t=h\varphi(t_n,y(t_n))$$

et on obtient le schéma

$$\begin{cases} u_0=y(t_0)=y_0,\\ u_{n+1}=u_n+h\varphi(t_n,u_n) \quad n=0,1,\dots N-1 \end{cases}$$

La méthode AB_1 coïncide avec la méthode d'EULER progressive.

⋆ $\boxed{j=2}$ On a

$$p(t)=\frac{\varphi(t_n,y(t_n))-\varphi(t_{n-1},y(t_{n-1}))}{h}(t-t_{n-1})+\varphi(t_{n-1},y(t_{n-1}))$$
$$\int_{t_n}^{t_{n+1}}p(t)\,\mathrm{d}t=\frac{h}{2}\left(3\varphi(t_n,y(t_n))-\varphi(t_{n-1},y(t_{n-1}))\right)$$

et on obtient le schéma

$$\begin{cases} u_0=y(t_0)=y_0,\\ u_1=u_0+h\varphi(t_0,u_0)\approx y(t_1)\\ u_{n+1}=u_n+\frac{h}{2}\left(3\varphi(t_n,u_n)-\varphi(t_{n-1},u_{n-1})\right) \quad n=1,2,\dots N-1 \end{cases}$$

où u_1 est une approximation de $y(t_1)$ obtenue en utilisant une prédiction AB_1.

⋆ $\boxed{j=3}$ On a

$$p(t)=\frac{\varphi(t_{n-2},y(t_{n-2}))}{2h^2}(t-t_{n-1})(t-t_n)-\frac{\varphi(t_{n-1},y(t_{n-1}))}{h^2}(t-t_{n-2})(t-t_n)+\frac{\varphi(t_n,y(t_n))}{2h^2}(t-t_{n-2})(t-t_{n-1})$$
$$\int_{t_n}^{t_{n+1}}p(t)\,\mathrm{d}t=\frac{h}{12}\big(23\varphi(t_n,y(t_n))-16\varphi(t_{n-1},y(t_{n-1}))+5\varphi(t_{n-2},y(t_{n-2}))\big)$$

et on obtient le schéma

$$\begin{cases}u_0=y(t_0)=y_0,\\ u_1=u_0+h\varphi(t_0,u_0)\approx y(t_1)\\ u_2=u_1+\frac{h}{2}\big(3\varphi(t_1,u_1)-\varphi(t_0,u_0)\big)\approx y(t_2)\\ u_{n+1}=u_n+\frac{h}{12}\big(23\varphi(t_n,u_n)-16\varphi(t_{n-1},u_{n-1}+5\varphi(t_{n-2},u_{n-2})\big)\quad n=2,3,\dots N-1\end{cases}$$

où u_1 est une approximation de $y(t_1)$ obtenue en utilisant une prédiction AB_1 et u_2 est une approximation de $y(t_2)$ obtenue en utilisant la méthode AB_2.

Méthode d'Adams-Moulton : il s'agit de méthodes implicites à $j\geq 0$ pas notées AM_{j+1}. Le polynôme p interpole f en les points $\{t_{n+1},t_n,t_{n-1},\dots,t_{n-j+1}\}$ où j est fixé. Elles permettent de calculer u_{n+1} de façon implicite à partir de l'ensemble $\{u_n,u_{n-1},\dots,u_{n-j+1}\}$ et il est donc possible de calculer successivement u_j, u_{j+1},..., en partant de $u_0,u_1,\dots,u_{j-1}$ (qui doivent donc être initialisés par des approximations adéquates car seul u_0 est donné).

⋆ $\boxed{j=0}$ On a

$$p(t)=\varphi(t_{n+1},y(t_{n+1}))$$
$$\int_{t_n}^{t_{n+1}}p(t)\,\mathrm{d}t=h\varphi(t_{n+1},y(t_{n+1}))$$

et on obtient le schéma

$$\begin{cases}u_0=y(t_0)=y_0,\\ u_{n+1}=u_n+h\varphi(t_{n+1},u_{n+1})\quad n=0,1,\dots N-1\end{cases}$$

La méthode AM_1 coïncide avec la méthode d'Euler régressive.

⋆ $\boxed{j=1}$ On a

$$p(t)=\frac{\varphi(t_{n+1},y(t_{n+1}))-\varphi(t_n,y(t_n))}{h}(t-t_n)+\varphi(t_n,y(t_n))$$
$$\int_{t_n}^{t_{n+1}}p(t)\,\mathrm{d}t=\frac{h}{2}\big(\varphi(t_n,y(t_n))+\varphi(t_{n+1},y(t_{n+1}))\big)$$

et on obtient le schéma

$$\begin{cases}u_0=y(t_0)=y_0,\\ u_{n+1}=u_n+\frac{h}{2}\big(\varphi(t_n,u_n)+\varphi(t_{n+1},u_{n+1})\big)\quad n=1,2,\dots N-1\end{cases}$$

La méthode AM_2 coïncide avec la méthode de Crank-Nicolson.

⋆ $\boxed{j=2}$ On a

$$p(t)=\frac{\varphi(t_{n-1},y(t_{n-1}))}{2h^2}(t-t_n)(t-t_{n+1})-\frac{\varphi(t_n,y(t_n))}{h^2}(t-t_{n-1})(t-t_{n+1})+\frac{\varphi(t_{n+1},y(t_{n+1}))}{2h^2}(t-t_{n-1})(t-t_n)$$
$$\int_{t_n}^{t_{n+1}}p(t)\,\mathrm{d}t=\frac{h}{12}\big(5\varphi(t_{n+1},y(t_{n+1}))+8\varphi(t_n,y(t_n))-\varphi(t_{n-1},y(t_{n-1}))\big)$$

et on obtient le schéma

$$\begin{cases}u_0=y(t_0)=y_0,\\ u_1=u_0+h\varphi(t_0,u_0)\approx y(t_1)\\ u_{n+1}=u_n+\frac{h}{12}\big(5\varphi(t_{n+1},u_{n+1})+8\varphi(t_n,u_n)-\varphi(t_{n-1},u_{n-1}\big)\quad n=1,2,\dots N-1\end{cases}$$

où u_1 est une approximation de $y(t_1)$ obtenue en utilisant une prédiction AB_1.

4.2.3. Schémas multi-pas de type *predictor-corrector*

Lorsqu'on utilise une méthode implicite, pour calculer u_{n+1} on doit résoudre une équation non-linéaire, par exemple avec la méthode de NEWTON. Une approche différente qui permet de s'affranchir de cette étape est donnée par les méthodes *predictor-corrector*. Une méthode *predictor-corrector* est une méthode qui permet de calculer u_{n+1} de façon explicite à partir d'une méthode implicite comme suit :

1. **predictor** : on calcule $\tilde{u}_{n+1}$ une approximation de u_{n+1} par une méthode explicite ;
2. **corrector** : on écrit une méthode implicite et on approche $\varphi(t_{n+1}, u_{n+1})$ par $\varphi(t_{n+1}, \tilde{u}_{n+1})$.

Exemple 1 : on a déjà rencontré une méthode de type *predictor-corrector* lorsqu'on a construit les schémas classiques : la méthode de HEUN

$$\begin{cases} u_0 = y_0 & \\ \tilde{u}_{n+1} = u_n + h\varphi(t_n, u_n) & n = 1, 2, \dots N-1 \\ u_{n+1} = u_n + \frac{h}{2}\left(\varphi(t_n, u_n) + \varphi(t_{n+1}, \tilde{u}_{n+1})\right) & n = 1, 2, \dots N-1 \end{cases}$$

est construite par les deux étapes suivantes :

⋆ predictor : méthode d'EULER explicite (ou AB_1) $\tilde{u}_{n+1} = u_n + h\varphi(t_n, u_n)$

⋆ corrector : méthode de CRANK-NICOLSON (ou AM_1) $\tilde{u}_{n+1} = u_n + \frac{h}{2}\left(\varphi(t_n, u_n) + \varphi(t_{n+1}, u_{n+1})\right)$

Exemple 2 : des méthodes de type *predictor-corrector* sont souvent construites en utilisant une prédiction d'ADAMS-BASHFORTH suivie d'une correction d'ADAMS-MOULTON. Par exemple, si on considère les deux étapes suivantes

⋆ predictor : méthode AB_2 $\tilde{u}_{n+1} = u_n + \frac{h}{2}\left(3\varphi(t_n, u_n) - \varphi(t_{n-1}, u_{n-1})\right)$

⋆ corrector : méthode AM_2 $\tilde{u}_{n+1} = u_n + \frac{h}{12}\left(5\varphi(t_{n+1}, u_{n+1}) + 8\varphi(t_n, u_n) - \varphi(t_{n-1}, u_{n-1}\right)$

on obtient la méthode AB_2-AM_2 :

$$\begin{cases} u_0 = y_0 & \\ u_1 = u_0 + h\varphi(t_0, u_0), & \\ \tilde{u}_{n+1} = u_n + \frac{3}{2}h\left(\varphi(t_n, u_n) - \varphi(t_{n-1}, u_{n-1})\right) & n = 2, 3, \dots N-1 \\ u_{n+1} = u_n + \frac{h}{12}\left(5\varphi(t_{n+1}, \tilde{u}_{n+1}) + 8\varphi(t_n, u_n) - \varphi(t_{n-1}, u_{n-1}\right) & n = 2, 3, \dots N-1 \end{cases}$$

4.3. Conditionnement

En générale, il ne suffit pas qu'un schéma numérique soit convergent pour qu'il donne des bons résultats sur n'importe quelle équation différentielle. Encore faut-il que le problème soit mathématiquement bien posé (existence et unicité de la solution), qu'il soit numériquement bien posé (continuité suffisamment bonne par rapport aux conditions initiales) et qu'il soit bien conditionné. Voyons dans l'exemple suivant ce que cela signifie.

Exemple *Problème de* CAUCHY *numériquement mal posé*

Une fois calculée la solution numérique $\{u_n\}_{n=1}^{N_h}$, il est légitime de chercher à savoir dans quelle mesure l'erreur $|y(t_n) - u_n|$ est petite pour $n = 1, 2, \dots$. Nous essayons de répondre à cette question en reprenant le premier exemple du chapitre. On se donne $\varphi(t, y) = 3t - 3y$ et $y_0 = \alpha$ (un nombre quelconque). On cherche une fonction $y\colon t \in \mathbb{R}^+ \mapsto y(t) \in \mathbb{R}$ qui satisfait

$$\begin{cases} y'(t) = 3t - 3y(t), & \forall t > 0, \\ y(0) = \alpha. \end{cases}$$

Nous avons vu que sa solution est donnée par $y(t) = (\alpha - 1/3)e^{3t} + t + 1/3$. Si nous cherchons à résoudre le problème de CAUCHY jusqu'à $t = 10$ avec $\alpha = 1/3$, nous obtenons $y(10) = 10 + 1/3 = 31/3$. Par contre, si nous faisons le calcul avec l'approximation $\alpha = 0.333333$ au lieu de $1/3$, nous avons $y(10) = (0.333333 - 1/3)e^{30} + 10 + 1/3 = -e^{30}/3000000 + 31/3$ ce qui représente une différence avec la précédente valeur de $e^{30}/3000000 \approx 10^7/3$. Cet exemple nous apprend qu'une petite erreur sur la condition initiale (erreur relative d'ordre 10^{-6}) peut provoquer une très grande erreur sur $y(10)$ (erreur relative d'ordre 10^6). Ainsi, si le calculateur mis à notre disposition ne calcul qu'avec 6 chiffres significatifs (en virgule flottante), alors $\alpha = 1/3$ devient $\alpha = 0.333333$ et il est inutile d'essayer d'inventer une méthode numérique pour calculer $y(10)$. En effet, la seule erreur sur la condition initiale provoque déjà une erreur inadmissible sur la solution. Nous sommes en présence ici d'un problème **numériquement mal posé**, appelé aussi **problème mal conditionné**.

4.4. Stabilité

Considérons le problème de CAUCHY (4.1) et supposons que l'on ait montré l'existence d'une solution y. Le principe des méthodes numériques est de subdiviser l'intervalle $I = [t_0, T]$ en N_h intervalles de longueur $h = (T - t_0)/N_h > 0$ où h est le pas de discrétisation. Alors, pour chaque nœud $t_n^{(h)} = t_0 + nh$ $(1 \le n \le N_h)$ on cherche la valeur inconnue $u_n^{(h)}$ qui approche $y(t_n^{(h)})$. L'ensemble des valeurs $\left\{u_0^{(h)} = y_0, u_1^{(h)}, \dots, u_{N_h}^{(h)}\right\}$ représente la solution numérique.

Deux questions naturelles se posent : que se passe-t-il lorsqu'on fait tendre le pas h vers 0 ? Que se passe-t-il lorsqu'on fixe le pas $h > 0$ mais on fait tendre T vers l'infini ? Dans les deux cas le nombre de nœuds tend vers l'infini mais dans le premier cas on s'intéresse à l'erreur en chaque point, dans le deuxième cas il s'agit du comportement asymptotique de la solution et de son approximation :

Zéro-stabilité : soit T fixé et considérons la limite $h \to 0$ (ainsi $N_h \to +\infty$). On note $e_n^{(h)} \equiv y(t_n^{(h)}) - u_n^{(h)} = y(t_0 + nh) - u_n^{(h)}$ l'erreur au point $t_0 + nh$. Il s'agit d'estimer le comportement de $e_n^{(h)}$ en tout point, *i.e.* pour tout $1 \leq n \leq N_h$. La méthode est zéro-stable si $e_n^{(h)} \xrightarrow[h\to 0]{} 0$ pour tout $n \in \mathbb{N}$.

Cette notion est très importante car le théorème de LAX-RICHTMYER (ou théorème d'équivalence) affirme que une méthode consistante et zéro-stable est convergente.

A-stabilité : on considère un problème de CAUCHY (4.1) dont la solution exacte vérifie $y(t) \xrightarrow[t\to+\infty]{} 0$. Soit h fixé et considérons la limite $T \to +\infty$ (ainsi $N_h \to +\infty$). On dit que la méthode est A-stable si $u_n^{(h)} \xrightarrow[n\to+\infty]{} 0$.

4.4.1. A-Stabilité

Dans la section précédente, on a considéré la résolution du problème de CAUCHY sur des intervalles bornés. Dans ce cadre, le nombre N_h de sous-intervalles ne tend vers l'infini que quand h tend vers zéro. Il existe cependant de nombreuses situations dans lesquelles le problème de CAUCHY doit être intégré sur des intervalles en temps très grands ou même infini. Dans ce cas, même pour h fixé, N_h tend vers l'infini. On s'intéresse donc à des méthodes capables d'approcher la solution pour des intervalles en temps arbitrairement grands, même pour des pas de temps h «assez grands».

Définition

Soit $\beta > 0$ un nombre réel positif et considérons le problème de CAUCHY

$$\begin{cases} y'(t) = -\beta y(t), & \text{pour } t > 0, \\ y(0) = y_0 \end{cases}$$

où $y_0 \neq 0$ est une valeur donnée. Sa solution est $y(t) = y_0 e^{-\beta t}$ et $\lim_{t\to+\infty} y(t) = 0$.
Soit $h > 0$ un pas de temps donné, $t_n = nh$ pour $n \in \mathbb{N}$ et notons $u_n \approx y(t_n)$ une approximation de la solution y au temps t_n.
Si, sous d'éventuelles conditions sur h, on a

$$\lim_{n\to+\infty} u_n = 0,$$

alors on dit que le schéma est A-stable.
On peut tirer des conclusions analogues quand β est un complexe ou une fonction positive de t. D'autre part, en général il n'y a aucune raison d'exiger qu'une méthode numérique soit absolument stable quand on l'applique à un autre problème. Cependant, on peut montrer que quand une méthode absolument stable sur le problème modèle est utilisée pour un problème modèle généralisé, l'erreur de perturbation (qui est la valeur absolue de la différence entre la solution perturbée et la solution non perturbée) est bornée uniformément (par rapport à h). En bref, on peut dire que les méthodes absolument stables permettent de contrôler les perturbations.

Étudions la A-stabilité des schémas classiques introduits ci-dessus.

⋆ Le ***schéma d'EULER progressif*** devient

$$u_{n+1} = (1 - \beta h)\, u_n, \qquad n = 0, 1, 2, \ldots$$

et par suite

$$u_n = (1 - \beta h)^n u_0, \qquad n = 0, 1, 2, \ldots$$

Par conséquente, $\lim_{n\to+\infty} u_n = 0$ si et seulement si

$$\left|1 - \beta h\right| < 1,$$

ce qui a pour effet de limiter h à

$$h \leq \frac{2}{\beta}.$$

Cette condition de A-stabilité limite le pas h d'avance en t lorsqu'on utilise le schéma d'EULER progressif. Notons que si $1 - \beta h > 1$ alors u_n tend vers $+\infty$ lorsque t tend vers l'infini et si $1 - \beta h < -1$ alors u_n tend vers l'infini en alternant de signe lorsque t tend vers l'infini. Nous dirons dans ces cas que le schéma d'EULER progressif est instable.

⋆ Le ***schéma d'EULER rétrograde*** devient dans le cadre de notre exemple

$$(1 + \beta h)\, u_{n+1} = u_n, \qquad n = 0, 1, 2, \ldots$$

et par suite

$$u_n = \frac{1}{(1+\beta h)^n} u_0, \qquad n = 0, 1, 2, \ldots$$

Dans ce cas nous voyons que pour tout $h > 0$ nous avons $\lim_{n\to\infty} u_n = 0$, le schéma d'EULER rétrograde est donc toujours stable, sans limitations sur h.

⋆ Le ***schéma de* CRANK-NICOLSON** appliqué à notre exemple s'écrit

$$\left(1 + \beta\frac{h}{2}\right) u_{n+1} = \left(1 - \beta\frac{h}{2}\right) u_n$$

et par suite

$$u_n = \left(\frac{2-\beta h}{2+\beta h}\right)^n u_0, \qquad n = 0, 1, 2, \ldots$$

Par conséquent, $\lim\limits_{n\to+\infty} u_n = 0$ si et seulement si

$$\left|\frac{2-\beta h}{2+\beta h}\right| < 1.$$

Notons x le produit $\beta h > 0$ et q la fonction $q(x) = \frac{2-x}{2+x} = 1 - 2\frac{x}{2+x}$. Nous avons $0 < \frac{x}{2+x} < 1$ pour tout $x \in \mathbb{R}_+^*$, donc $|q(x)| < 1$ pour tout $x \in \mathbb{R}_+^*$. La relation $\lim\limits_{n\to+\infty} u_n = 0$ est donc satisfaite pour tout $h > 0$: le schéma de CRANK-NICOLSON est donc toujours stable, sans limitations sur h.

⋆ Le ***schéma de* HEUN** pour notre exemple devient

$$u_{n+1} = \left(1 - \beta h + \frac{(\beta h)^2}{2}\right) u_n$$

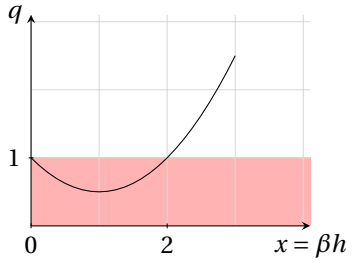
Par induction on obtient

$$u_n = \left(1 - \beta h + \frac{(\beta h)^2}{2}\right)^n u_0.$$

Par conséquent, $\lim\limits_{n\to+\infty} u_n = 0$ si et seulement si

$$\left|1 - \beta h + \beta^2 \frac{(h)^2}{2}\right| < 1.$$

Notons x le produit βh et q le polynôme $q(x) = \frac{1}{2}x^2 - x + 1$ dont le graphe est représenté en figure.



Nous avons $|q(x)| < 1$ si et seulement si $0 < x < 2$. La relation $\lim\limits_{n\to+\infty} u_n = 0$ est donc satisfaite si et seulement si

$$h < \frac{2}{\beta}.$$

Cette condition de stabilité limite le pas h d'avance en t lorsqu'on utilise le schéma de HEUN.

A première vue, il semble que le schéma d'EULER progressif et le schéma de HEUN soient préférable au schéma d'EULER rétrograde et de CRANK-NICOLSON puisque ces derniers ne sont pas explicites. Cependant, les méthodes d'EULER implicite et de CRANK-NICOLSON sont inconditionnellement A-stables. C'est aussi le cas de nombreuses autres méthodes implicites. Cette propriété rend les méthodes implicites attractives, bien qu'elles soient plus coûteuses que les méthodes explicites.

Codes Python

Voici les `function` python des méthodes illustrées dans ce chapitre

```
#!/usr/bin/python
#-*- coding: Utf-8 -*-

import math
import sys
import matplotlib.pyplot as plt

def euler_progressif(f,tt,N):
    yy = [y0]
    for i in range(N):
        yy.append(yy[i]+h*f(tt[i],yy[i]))
    return yy

def euler_modifie(f,tt,N):
    yy = [y0]
    for i in range(N):
        yy.append(yy[i]+h*f(tt[i]+h*0.5,yy[i]+h*0.5*f(tt[i],yy[i])))
    return yy

def heun(f,tt,N):
    yy = [y0]
    for i in range(N):
        yy.append(yy[i]+h*(f(tt[i],yy[i])+f(tt[i+1],yy[i]+h*f(tt[i],yy[i]))))
    return yy
```

et voici un exemple

```
# INITIALISATION
N = 3
exemple = 4

if exemple==1 :
    t0 = 0.
    y0 = 1.
    tfinal = 3.
    def f(t,y):
        return y
    def sol_exacte(t):
        return math.exp(t)
elif exemple==2 :
    t0 = 0.
    y0 = 1.
    tfinal = 3.
    def f(t,y):
        return t
    def sol_exacte(t):
        return 1.+0.5*t**2
elif exemple==3 :
    t0 = 0.
    y0 = 0.
    tfinal = 1.
    def f(t,y):
        return math.cos(2*y)
    def sol_exacte(t):
        return 0.5*math.asin((math.exp(4.*t)-1.)/(math.exp(4.*t)+1.))
else :
    print "Exemple non defini"
    sys.exit(0)

# CALCUL
h = (tfinal-t0)/N
tt = [ t0+i*h for i in range(N+1) ]
yy_exacte = [sol_exacte(t) for t in tt]
yy_euler_progressif = euler_progressif(f,tt,N)
```

```
62 yy_euler_modifie = euler_modifie(f,tt,N)
63 yy_heun = euler_modifie(f,tt,N)
64
65 # AFFICHAGE
66 plt.axis([t0, tfinal, min(yy_euler_progressif), max(yy_euler_progressif)])
67 plt.plot(tt,yy_exacte,'m',tt,yy_euler_progressif,'go',tt,yy_euler_modifie,'cs',tt,yy_heun,'r^')
68 plt.show()
```

Exercices

Exercice 4.1

Considérons le problème de CAUCHY : trouver $y\colon [t_0,T]\subset\mathbb{R}\to\mathbb{R}$ tel que

$$\begin{cases} y'(t)=\varphi(t,y(t)), & \forall\, t\in[t_0,T],\\ y(t_0)=y_0. \end{cases}$$

Supposons que l'on ait montré l'existence d'une unique solution y.
Le principe des méthodes numériques est de subdiviser l'intervalle $[t_0,T]$ en N intervalles de longueur $h=(T-t_0)/N=t_{n+1}-t_n$. Pour chaque nœud $t_n=t_0+nh$ $(1\le n\le N)$ on cherche la valeur inconnue u_n qui approche $y(t_n)$. L'ensemble des valeurs $\{u_0=y_0,u_1,\dots,u_N\}$ représente la solution numérique.

Dans cette exercice on va construire des nouveaux schémas numériques basés sur l'intégration de l'EDO $y'(t)=\varphi(t,y(t))$ entre t_n et t_{n+2} :

$$y(t_{n+2})=y(t_n)+\int_{t_n}^{t_{n+2}}\varphi(t,y(t))\,\mathrm{d}t.$$

1. En utilisant la formule de quadrature du point milieu pour approcher le membre de droite écrire un schéma numérique explicite permettant de calculer u_{n+2} à partir de u_{n+1} et u_n. Notons que ce schéma a besoin de deux valeurs initiales ; on posera alors $u_0=y_0$ et u_1 sera approché par une prédiction d'EULER progressive.
2. En utilisant la formule de quadrature de CAVALIERI-SIMPSON pour approcher le membre de droite écrire un schéma numérique implicite permettant de calculer u_{n+2} à partir de u_{n+1} et u_n. Notons que ce schéma a besoin de deux valeurs initiales ; on posera alors $u_0=y_0$ et u_1 sera approché par une prédiction d'EULER progressive.
3. Proposer une modification du schéma au point précédent pour qu'il devient explicite.

CORRECTION DE L'EXERCICE 4.1.

1. Si on utilise la formule de quadrature du point milieu sur l'intervalle $[t_n;t_{n+2}]$, *i.e.*

$$\int_{t_n}^{t_{n+2}}\varphi(t,y(t))\,\mathrm{d}t\approx 2h\varphi\big(t_{n+1},y(t_{n+1})\big)$$

on obtient

$$\begin{cases} u_0=y(t_0)=y_0,\\ u_1\approx y(t_1)\\ u_{n+2}=u_n+2h\varphi(t_{n+1},u_{n+1}) \quad n=1,2,\dots N-2 \end{cases}$$

où u_1 est une approximation de $y(t_1)$. Nous pouvons utiliser une prédiction d'EULER progressive pour approcher u_1. Nous avons construit ainsi un nouveau schéma explicite à trois pas :

$$\begin{cases} u_0=y(t_0)=y_0,\\ u_1=u_0+h\varphi(t_0,u_0),\\ u_{n+2}=u_n+2h\varphi(t_{n+1},u_{n+1}) \quad n=1,2,\dots N-2 \end{cases}$$

Il s'agit d'un schéma explicite car il permet d'expliciter u_{n+2} en fonction de u_n et de u_{n+1}.

2. Si on utilise la formule de quadrature de CAVALIERI-SIMPSON sur l'intervalle $[t_n;t_{n+2}]$, *i.e.*

$$\int_{t_n}^{t_{n+2}}\varphi(t,y(t))\,\mathrm{d}t\approx\frac{h}{3}\big(\varphi(t_n,y(t_n))+4\varphi(t_{n+1},y(t_{n+1}))+\varphi(t_{n+2},y(t_{n+2}))\big),$$

et une prédiction d'EULER progressive pour approcher u_1, nous obtenons un nouveau schéma implicite à trois pas :

$$\begin{cases} u_0=y(t_0)=y_0,\\ u_1=u_0+h\varphi(t_0,u_0),\\ u_{n+2}=u_n+\frac{h}{3}\big(\varphi(t_n,u_n)+4\varphi(t_{n+1},u_{n+1})+\varphi(t_{n+2},u_{n+2})\big) \quad n=1,2,\dots N-2 \end{cases}$$

3. Pour éviter le calcul implicite de u_{n+2}, nous pouvons utiliser une prédiction d'EULER progressive et remplacer le u_{n+2} dans le terme $\varphi(t_{n+2},u_{n+2})$ par $\tilde{u}_{n+2}=u_{n+1}+h\varphi(t_{n+1},u_{n+1})$. Nous avons construit ainsi un nouveau schéma explicite à trois pas :

$$\begin{cases} u_0=y(t_0)=y_0,\\ u_1=u_0+h\varphi(t_0,u_0),\\ u_{n+2}=u_n+\frac{h}{3}\big(\varphi(t_n,u_n)+4\varphi(t_{n+1},u_{n+1})+\varphi(t_{n+2},u_{n+1}+h\varphi(t_{n+1},u_{n+1}))\big) \quad n=1,2,\dots N-2 \end{cases}$$

Exercice 4.2 *Méthodes d'*ADAMS-BASHFORTH

Considérons le problème de CAUCHY suivant dont on suppose qu'il existe une et une seule solution :

$$\text{trouver } y\colon [t_0,T]\subset\mathbb{R}\to\mathbb{R} \text{ tel que}$$
$$\begin{cases} y'(t)=\varphi(t,y(t)), & \forall t\in[t_0,T],\\ y(t_0)=y_0. \end{cases}$$

Le principe des méthodes numériques pour approcher la fonction y est de subdiviser l'intervalle $[t_0,T]$ en N intervalles de longueur $h=(T-t_0)/N>0$. Pour chaque nœud $t_n=t_0+nh$ $(1\le n\le N)$ on cherche la valeur inconnue u_n qui approche $y(t_n)$. L'ensemble des valeurs $\{u_0=y_0,u_1,\dots,u_N\}$ représente la solution numérique.

Dans cette exercice on va construire des nouveaux schémas numériques basés sur l'intégration approchée de l'EDO $y'(t)=\varphi(t,y(t))$ entre t_n et t_{n+1} car l'on a

$$y(t_{n+1})=y(t_n)+\int_{t_n}^{t_{n+1}}\varphi(t,y(t))\,\mathrm{d}t.$$

Les schémas d'ADAM approchent l'intégrale précédent par l'intégrale d'un polynôme interpolant f en des points donnés.

1. Écrire le schéma explicite obtenu en choisissant comme unique point à interpoler le point t_n.
2. Écrire le schéma explicite obtenu en choisissant comme points à interpoler les points $\{t_{n-1},t_n\}$ en proposant une adéquate initialisation de la suite. (Attention : on intègre f sur l'intervalle $[t_n,t_{n+1}]$ mais on interpole f en t_n et t_{n-1})

CORRECTION DE L'EXERCICE 4.2.

1. On a

$$p(t)=\varphi(t_n,y(t_n))$$
$$\int_{t_n}^{t_{n+1}}p(t)\,\mathrm{d}t=h\varphi(t_n,y(t_n))$$

et on obtient le schéma

$$\begin{cases} u_0=y(t_0)=y_0,\\ u_{n+1}=u_n+h\varphi(t_n,u_n) \quad n=0,1,\dots N-1 \end{cases}$$

Il s'agit d'un schéma explicite appelé schéma d'ADAMS-BASHFORTH à un pas (qui coïncide avec la méthode d'EULER progressive).

2. On a

$$p(t)=\frac{\varphi(t_n,y(t_n))-\varphi(t_{n-1},y(t_{n-1}))}{h}(t-t_{n-1})+\varphi(t_{n-1},y(t_{n-1}))$$
$$\int_{t_n}^{t_{n+1}}p(t)\,\mathrm{d}t=\frac{h}{2}\left(3\varphi(t_n,y(t_n))-\varphi(t_{n-1},y(t_{n-1}))\right)$$

et on obtient le schéma

$$\begin{cases} u_0=y(t_0)=y_0,\\ u_1\approx y(t_1)\\ u_{n+1}=u_n+\frac{h}{2}\left(3\varphi(t_n,u_n)-\varphi(t_{n-1},u_{n-1})\right) \quad n=1,2,\dots N-1 \end{cases}$$

où u_1 est une approximation de $y(t_1)$. Nous pouvons utiliser une prédiction d'EULER progressive pour approcher u_1. Nous avons construit ainsi un nouveau schéma explicite à deux pas :

$$\begin{cases} u_0=y(t_0)=y_0,\\ u_1=u_0+h\varphi(t_0,u_0),\\ u_{n+1}=u_n+\frac{h}{2}\left(3\varphi(t_n,u_n)-\varphi(t_{n-1},u_{n-1})\right) \quad n=1,2,\dots N-1. \end{cases}$$

Il s'agit d'un schéma explicite appelé schéma d'ADAMS-BASHFORTH à deux pas.

Exercice 4.3 *Méthodes d'*ADAMS-MOULTON

Considérons le problème de CAUCHY suivant dont on suppose qu'il existe une et une seule solution :

$$\text{trouver } y\colon [t_0,T]\subset\mathbb{R}\to\mathbb{R} \text{ tel que}$$
$$\begin{cases} y'(t)=f(t,y(t)), & \forall t\in[t_0,T],\\ y(t_0)=y_0. \end{cases}$$

Le principe des méthodes numériques pour approcher la fonction y est de subdiviser l'intervalle $[t_0,T]$ en N intervalles de longueur $h=(T-t_0)/N>0$. Pour chaque nœud $t_n=t_0+nh$ $(1\le n\le N)$ on cherche la valeur inconnue u_n qui approche $y(t_n)$ à partir de $u_0=y_0$. L'ensemble des valeurs $\{u_0,u_1,\dots,u_N\}$ représente la solution numérique.

Dans cette exercice on va construire des schémas numériques basés sur l'intégration approchée de l'EDO $y'(t)=f(t,y(t))$ entre t_n et t_{n+1} à partir de la relation

$$y(t_{n+1})=y(t_n)+\int_{t_n}^{t_{n+1}} f(t,y(t))\,\mathrm{d}t.$$

Les schémas d'ADAM approchent l'intégrale précédente par l'intégrale d'un polynôme interpolant f en des points donnés.

1. Écrire le schéma implicite obtenu en choisissant comme points à interpoler le point $\{t_n\}$. Quel schéma reconnait-on ?
2. Écrire le schéma implicite obtenu en choisissant comme points à interpoler les points $\{t_n,t_{n+1}\}$. Quel schéma reconnait-on ?
3. Écrire le schéma implicite obtenu en choisissant comme points à interpoler les points $\{t_{n-1},t_n,t_{n+1}\}$ en proposant une adéquate initialisation de la suite. (Attention : on intègre f sur l'intervalle $[t_n,t_{n+1}]$ mais on interpole f en t_{n+1}, t_n et t_{n-1})

CORRECTION DE L'EXERCICE 4.3.

1. On a
$$p(t)=f(t_{n+1},y(t_{n+1}))$$
$$\int_{t_n}^{t_{n+1}} p(t)\,\mathrm{d}t=hf(t_{n+1},y(t_{n+1}))$$
et on obtient le schéma
$$\begin{cases} u_0=y(t_0)=y_0,\\ u_{n+1}=u_n+hf(t_{n+1},u_{n+1}) \quad n=0,1,\dots N-1 \end{cases}$$
Il s'agit d'un schéma implicite appelé schéma d'ADAMS-MOULTON à un pas (qui coïncide avec la méthode d'EULER régressive).
2. On a
$$p(t)=\frac{f(t_{n+1},y(t_{n+1}))-f(t_n,y(t_n))}{h}(t-t_n)+f(t_n,y(t_n))$$
$$\int_{t_n}^{t_{n+1}} p(t)\,\mathrm{d}t=\frac{h}{2}\big(f(t_n,y(t_n))+f(t_{n+1},y(t_{n+1}))\big)$$
et on obtient le schéma
$$\begin{cases} u_0=y(t_0)=y_0,\\ u_{n+1}=u_n+\frac{h}{2}\big(f(t_n,u_n)+f(t_{n+1},u_{n+1})\big) \quad n=1,2,\dots N-1 \end{cases}$$
Il s'agit d'un schéma implicite appelé schéma d'ADAMS-MOULTON à deux pas qui coïncide avec le schéma de CRANK-NICOLSON.
3. On a
$$p(t)=\frac{f(t_{n-1},y(t_{n-1}))}{2h^2}(t-t_n)(t-t_{n+1})-\frac{f(t_n,y(t_n))}{h^2}(t-t_{n-1})(t-t_{n+1})+\frac{f(t_{n+1},y(t_{n+1}))}{2h^2}(t-t_{n-1})(t-t_n)$$
$$\int_{t_n}^{t_{n+1}} p(t)\,\mathrm{d}t=\frac{h}{12}\big(5f(t_{n+1},y(t_{n+1}))+8f(t_n,y(t_n))-f(t_{n-1},y(t_{n-1}))\big)$$

et on obtient le schéma

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 = u_0 + hf(t_0, u_0) \approx y(t_1) \\ u_{n+1} = u_n + \frac{h}{12}\left(5f(t_{n+1}, u_{n+1}) + 8f(t_n, u_n) - f(t_{n-1}, u_{n-1}\right) \quad n = 1, 2, \dots N-1 \end{cases}$$

où u_1 est une approximation de $y(t_1)$ obtenue en utilisant une prédiction d'EULER progressive.

Exercice 4.4 *A-stabilité de la méthode d'*EULER *explicite en fonction du pas*

On considère le problème de CAUCHY

$$\begin{cases} y'(t) = -y(t), \\ y(0) = 1, \end{cases}$$

sur l'intervalle $[0;10]$.

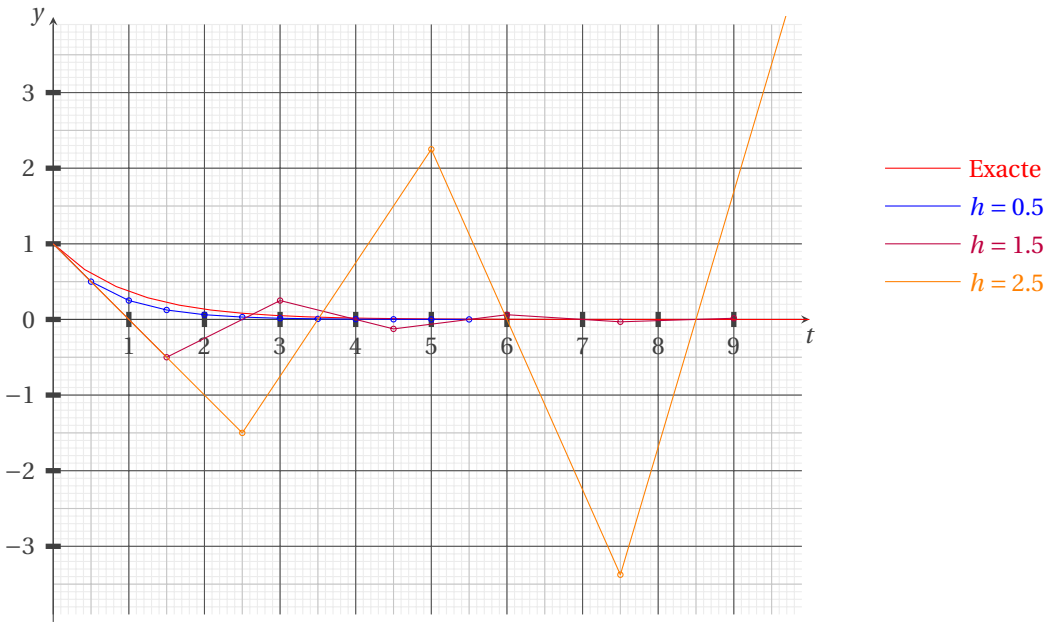
1. Calculer la solution exacte du problème de CAUCHY.
2. Soit h le pas temporel. Écrire la méthode d'EULER explicite pour cette équation différentielle ordinaire (EDO).
3. En déduire une forme du type
$$u_{n+1} = g(h, n)$$
avec $g(h, n)$ à préciser (autrement dit, l'itérée en t_n ne dépend que de h et n et ne dépend pas de u_n).
4. Utiliser la formulation ainsi obtenue pour tracer les solutions
   - ⋆ exacte,
   - ⋆ obtenue avec la méthode d'EULER avec $h = 2.5$,
   - ⋆ obtenue avec la méthode d'EULER avec $h = 1.5$,
   - ⋆ obtenue avec la méthode d'EULER avec $h = 0.5$.
5. Que peut-on en déduire sur la A-stabilité de la méthode ?

CORRECTION DE L'EXERCICE 4.4.

1. Il s'agit d'une EDO à variables séparables. L'unique solution constante de l'EDO est la fonction $y(t) \equiv 0$, toutes les autres solutions sont du type $y(t) = Ce^{-t}$. Donc l'unique solution du problème de CAUCHY est la fonction $y(t) = e^{-t}$ définie pour tout $t \in \mathbb{R}$.
2. La méthode d'EULER est une méthode d'intégration numérique d'EDO du premier ordre de la forme $y'(t) = F(t, y(t))$. C'est une méthode itérative : la valeur y à l'instant $t+h$ se déduisant de la valeur de y à l'instant t par l'approximation linéaire
$$y(t+h) \approx y(t) + y'(t)h = y(t) + F(t, y(t))h.$$
En choisissant un pas de discrétisation h, nous obtenons une suite de valeurs (t_n, u_n) qui peuvent être une excellente approximation de la fonction $y(t)$ avec
$$\begin{cases} t_n = t_0 + nh, \\ u_n = u_{n-1} + F(t_{n-1}, u_{n-1})h. \end{cases}$$
La méthode d'EULER explicite pour cette EDO s'écrit donc
$$u_{n+1} = (1-h)u_n.$$
3. En procédant par récurrence sur n, on obtient
$$u_{n+1} = (1-h)^{n+1}.$$
4. On a donc
   - ⋆ si $h = 2.5$ alors $u_n = \left(-\frac{3}{2}\right)^n$ tandis que $y(t_n) = e^{-5n/2}$,
   - ⋆ si $h = 1.5$ alors $u_n = \left(-\frac{1}{2}\right)^n$ tandis que $y(t_n) = e^{-3n/2}$,
   - ⋆ si $h = 0.5$ alors $u_n = \left(\frac{1}{2}\right)^n$ tandis que $y(t_n) = e^{-n/2}$.

	u_n		
$t_n = nh$	$h = 2.5$	$h = 1.5$	$h = 0.5$
0	1	1	1
0.5			0.5
1			0.25
1.5		−0.5	0.125
2			0.0625
2.5	−1.5		0.03125
3		0.25	0.015625
3.5			0.0078125
4			0.00390625
4.5		−0.125	0.001953125
5	2.25		0.0009765625
5.5			0.00048828125
6		0.0625	0.000244140625
6.5			0.0001220703125
7			$6.103515625 \times 10^{-5}$
7.5	−3.75	−0.03125	$3.0517578125 \times 10^{-5}$
8			$1.52587890625 \times 10^{-5}$
8.5			$7.62939453125 \times 10^{-6}$
9		0.015625	$3.814697265625 \times 10^{-6}$
9.5			$1.9073486328125 \times 10^{-6}$
10	5.0625		$9.5367431640625 \times 10^{-7}$

Ci-dessous sont tracées sur l'intervalle $[0;10]$, les courbes représentatives de la solution exacte et de la solution calculée par la méthode d'EULER explicite. En faisant varier le pas h nous pouvons constater que si $h = 2.5$ l'erreur commise entre la solution exacte et la solution calculée est amplifiée d'un pas à l'autre.



NB : les trois premières itérées ont la même pente (se rappeler de la construction géométrique de la méthode d'EULER).

5. De la formule $u_{n+1} = (1-h)^{n+1}$ on déduit que
   - ⋆ si $0 < h < 1$ alors la solution numérique est stable et convergente,
   - ⋆ si $1 < h < 2$ alors la solution numérique oscille mais est encore convergente,
   - ⋆ si $h > 2$ alors la solution numérique oscille et divergente.

   En effet, on sait que la méthode est A-stable si et seulement si $|1-h| < 1$.

   Remarque : la suite obtenue est une suite géométrique de raison $q = 1-h$. On sait qu'une telle suite
   - ⋆ diverge si $|q| > 1$ ou $q = -1$,
   - ⋆ est stationnaire si $q = 1$,
   - ⋆ converge vers 0 $|q| < 1$.

Exercice 4.5

L'évolution de la concentration de certaines réactions chimiques au cours du temps peut être décrite par l'équation différentielle

$$y'(t) = -\frac{1}{1+t^2}\,y(t).$$

Sachant qu'à l'instant $t = 0$ la concentration est $y(0) = 5$, déterminer la concentration à $t = 2$ à l'aide de la méthode d'**Euler implicite** avec un pas $h = 0.5$.

Correction de l'exercice 4.5. La méthode d'Euler implicite est une méthode d'intégration numérique d'EDO du premier ordre de la forme $y'(t) = F(t, y(t))$. C'est une méthode itérative : en choisissant un pas de discrétisation h, la valeur y à l'instant $t + h$ se déduit de la valeur de y à l'instant t par l'approximation linéaire

$$y(t+h) \approx y(t) + hy'(t+h) = y(t) + hF(t+h, y(t+h)).$$

On pose alors $t_n = t_0 + nh$, $n \in \mathbb{N}$. En résolvant l'équation non-linéaire

$$u_{n+1} = u_n + hF(t_{n+1}, u_{n+1}),$$

on obtient une suite $(u_n)_{n\in\mathbb{N}}$ qui approche les valeurs de la fonction y en t_n. Dans notre cas, l'équation non-linéaire s'écrit

$$u_{n+1} = u_n - \frac{h}{1+t_{n+1}^2}\,u_{n+1}.$$

Elle peut être résolue algébriquement et cela donne la suite

$$u_{n+1} = \frac{u_n}{1+\frac{h}{1+t_{n+1}^2}}.$$

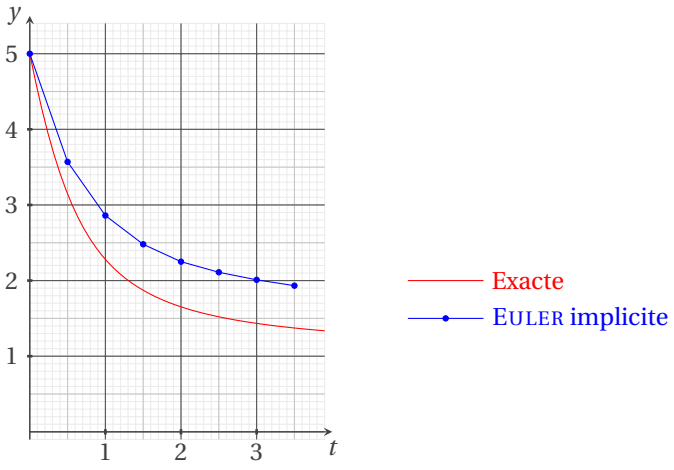
Si à l'instant $t = 0$ la concentration est $y(0) = 5$, et si $h = 1/2$, alors $t_n = n/2$ et

$$u_{n+1} = \frac{4+(n+1)^2}{6+(n+1)^2}\,u_n.$$

On obtient donc

n	t_n	u_n
0	0	5
1	0.5	$\frac{4+1^2}{6+1^2}5 = \frac{5}{7}5 = \frac{25}{7} \approx 3.57$
2	1.0	$\frac{4+2^2}{6+2^2}\frac{25}{7} = \frac{8}{10}\frac{25}{7} = \frac{20}{7} \approx 2.86$
3	1.5	$\frac{4+3^2}{6+3^2}\frac{20}{7} = \frac{13}{15}\frac{20}{7} = \frac{52}{21} \approx 2.48$
4	2.0	$\frac{4+4^2}{6+4^2}\frac{52}{21} = \frac{20}{22}\frac{52}{21} = \frac{520}{231} \approx 2.25$

La concentration à $t = 2$ est d'environ 2.25 qu'on peut comparer avec le calcul exact $y(2) = 5e^{-\arctan(2)} \approx 1.652499838$.



Exercice 4.6

Soit $\beta > 0$ un nombre réel positif et considérons le problème de CAUCHY

$$\begin{cases} y'(t) = -\beta y(t), & \text{pour } t > 0, \\ y(0) = y_0, \end{cases} \tag{4.2}$$

où y_0 est une valeur donnée. Soit $h > 0$ un pas de temps donné, $t_n = nh$ pour $n \in \mathbb{N}$ (ainsi $t_0 = 0$) et u_n une approximation de $y(t_n)$.

1. Écrire le schéma du trapèze (appelé aussi de CRANK-NICOLSON) permettant de calculer u_{n+1} à partir de u_n. Sous quelle condition sur h le schéma du trapèze est-il A-stable ? Autrement dit, pour quelles valeurs de h la relation $\lim_{n\to+\infty} u_n = 0$ a-t-elle lieu ?
2. À partir du schéma du trapèze, en déduire le schéma de HEUN. Sous quelle condition sur h le schéma de HEUN est-il A-stable ?

CORRECTION DE L'EXERCICE 4.6. Le problème (4.2) est un problème de CAUCHY de la forme (4.1) avec $\varphi(t, y) = -\beta y$. Le principe des méthodes d'approximation est de subdiviser l'intervalle I en sous-intervalles de longueur h et, pour chaque nœud $t_n = t_0 + nh$ (avec $n \in \mathbb{N}$), on cherche la valeur inconnue u_n qui approche $y(t_n)$. L'ensemble de valeurs $\{u_0, u_1, \dots\}$ représente la solution numérique. Les schémas numériques permettent de calculer u_{n+1} à partir de u_n et il est donc possible de calculer successivement u_1, u_2,... en partant de u_0.

1. Si nous intégrons l'EDO $y'(t) = \varphi(t, y(t))$ entre t_n et t_{n+1} nous obtenons

$$y(t_{n+1}) - y(t_n) = \int_{t_n}^{t_{n+1}} \varphi(t, y(t)) \, \mathrm{d}t.$$

Soit u_n une approximation de $u(t_n)$ et u_{n+1} une approximation de $y(t_{n+1})$. Si on utilise la formule du trapèze, *i.e.*

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t)) \, \mathrm{d}t \approx \frac{h}{2} \left(\varphi(t_n, y(t_n)) + \varphi(t_{n+1}, y(t_{n+1})) \right)$$

on obtient le *schéma du trapèze ou de* CRANK-NICOLSON

$$\begin{cases} u_0 = y(t_0), \\ u_{n+1} - \frac{h}{2}\varphi(t_{n+1}, u_{n+1}) = u_n + \frac{h}{2}\varphi(t_n, u_n), & \text{pour } n = 0, 1, 2, \dots \end{cases}$$

Il s'agit d'un schéma implicite car il ne permet pas d'écrire directement u_{n+1} en fonction de u_n lorsque la fonction φ n'est pas triviale. En appliquant le schéma du trapèze au problème (4.2) on obtient la suite définie par récurrence suivante

$$\begin{cases} u_0 = y(t_0), \\ \left(1 + \frac{h}{2}\beta\right) u_{n+1} = \left(1 - \frac{h}{2}\beta\right) u_n. \end{cases}$$

Par induction on obtient

$$u_n = \left(\frac{2 - \beta h}{2 + \beta h} \right)^n y_0.$$

Par conséquent, $\lim_{n\to+\infty} u_n = 0$ si et seulement si

$$\left| \frac{2 - \beta h}{2 + \beta h} \right| < 1.$$

Notons x le produit $\beta h > 0$ et q la fonction $q(x) = \frac{2-x}{2+x} = 1 - 2\frac{x}{2+x}$. Nous avons $0 < \frac{x}{2+x} < 1$ pour tout $x \in \mathbb{R}_+^*$, donc $|q(x)| < 1$ pour tout $x \in \mathbb{R}_+^*$. La relation $\lim_{n\to+\infty} u_n = 0$ est donc satisfaite pour tout $h > 0$.

2. Pour éviter le calcul implicite de u_{n+1} dans le schéma du trapèze, nous pouvons utiliser une prédiction d'EULER progressive et remplacer le u_{n+1} dans le terme $\varphi(t_{n+1}, u_{n+1})$ par $\tilde{u}_{n+1} = u_n + h\varphi(t_n, u_n)$. Nous avons construit ainsi le *schéma de* HEUN. Plus précisément, cette méthode s'écrit

$$\begin{cases} u_0 = y(t_0), \\ u_{n+1} = u_n + \frac{h}{2} \left(\varphi(t_n, u_n) + \varphi\left(t_{n+1}, u_n + h\varphi(t_n, u_n)\right) \right), & \text{pour } n = 0, 1, 2, \dots \end{cases}$$

En appliquant le schéma de HEUN au problème (4.2) on obtient la suite définie par récurrence suivante

$$\begin{cases} u_0 = y(t_0), \\ u_{n+1} = \left(1 - \beta h + \frac{(\beta h)^2}{2}\right) u_n. \end{cases}$$

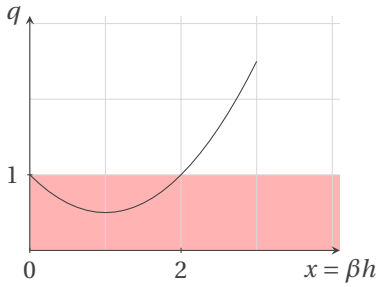
Par induction on obtient

$$u_n = \left(1 - \beta h + \frac{(\beta h)^2}{2}\right)^n y_0.$$

Par conséquent, $\lim_{n\to+\infty} u_n = 0$ si et seulement si

$$\left|1 - \beta h + \beta^2 \frac{(h)^2}{2}\right| < 1.$$

Notons x le produit βh et q le polynôme $q(x) = \frac{1}{2}x^2 - x + 1$ dont le graphe est représenté en figure.



Nous avons $|q(x)| < 1$ si et seulement si $0 < x < 2$. La relation $\lim_{n\to+\infty} u_n = 0$ est donc satisfaite si et seulement si

$$h < \frac{2}{\beta}.$$

Exercice 4.7

On considère le problème de CAUCHY

$$\begin{cases} y'(t) = -(y(t))^m + \cos(t), & \text{pour } t > 0, \\ y(0) = 0, \end{cases} \tag{4.3}$$

où m est un entier impair.

1. Montrer que le problème (4.3) possède une solution unique locale.
2. Soit $h > 0$ un pas de temps donné, soit $t_n = nh$ pour $n \in \mathbb{N}$ et u_n une approximation de $y(t_n)$. Écrire le schéma d'EULER rétrograde permettant de calculer u_{n+1} à partir de u_n.
3. À partir du schéma obtenu au point précédent, écrire un seul pas de la méthode de NEWTON pour calculer une nouvelle approximation de u_{n+1}. En déduire ainsi un nouveau schéma explicite.

CORRECTION DE L'EXERCICE 4.7.

1. Le problème (4.3) est un problème du type
   trouver $y\colon I \subset \mathbb{R}^+ \to \mathbb{R}$ tel que
   $$\begin{cases} y'(t) = \varphi(t, y(t)), & \forall t \in I, \\ y(t_0) = y_0, \end{cases}$$
   Si la fonction $\varphi(t, y)$ est de classe $\mathscr{C}^1$ par rapport à ses deux variables alors la solution $y = y(t)$ du problème de CAUCHY existe, est unique et appartient à $\mathscr{C}^1(I)$.
   Dans notre cas, $\varphi(t, y) = -(y^m) + \cos(t)$, donc $\partial_t\varphi(t, y) = -\sin(t)$ et $\partial_y\varphi(t, y) = -m(y^{m-1})$ qui sont de classe $\mathscr{C}^1$, donc le problème (4.3) possède une solution unique.
2. Le problème (4.2) est un problème du type
   trouver $y\colon I \subset \mathbb{R}^+ \to \mathbb{R}$ tel que
   $$\begin{cases} y'(t) = \varphi(t, y(t)), & \forall t \in I, \\ y(t_0) = y_0. \end{cases}$$
   Le principe des méthodes d'approximation est de subdiviser l'intervalle I en sous-intervalles de longueur h et, pour chaque nœud $t_n = t_0 + nh$ $(n \geq 0)$, on cherche la valeur inconnue u_n qui approche $y(t_n)$. L'ensemble de valeurs $\{u_0, u_1, \ldots\}$ représente la solution numérique. Le schéma de EULER rétrograde établit une relation entre u_n et u_{n+1} et il est donc possible de calculer successivement u_1, u_2,..., en partant de u_0.

Si nous intégrons l'EDO $y'(t) = \varphi(t, y(t))$ entre t_n et t_{n+1} nous obtenons

$$y(t_{n+1}) - y(t_n) = \int_{t_n}^{t_{n+1}} \varphi(t, y(t)) \, \mathrm{d}t.$$

Soit u_n une approximation de $y(t_n)$ et u_{n+1} une approximation de $y(t_{n+1})$. Si on utilise la formule du rectangle à droite, *i.e.*

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t)) \, \mathrm{d}t \approx h\varphi(t_{n+1}, y(t_{n+1}))$$

on obtient le *schéma d'*EULER *rétrograde* :

$$\begin{cases} y_0 = 0, \\ u_{n+1} = u_n + h\left(-(u_{n+1})^m + \cos(t_{n+1})\right). \end{cases}$$

Ce schéma est implicite car il ne permet pas de calculer u_{n+1} directement à partir de u_n.

3. Il s'agit de trouver u_{n+1} tel que

$$u_{n+1} = u_n + h\left(-u_{n+1}^m + \cos(t_{n+1})\right).$$

Pour déterminer u_{n+1} nous devons chercher le zéro de la fonction g_n définie par

$$g_n(x) = x + hx^m - (h\cos(t_{n+1}) + u_n).$$

La méthode de NEWTON pour approcher le zéro de g_n construit une suite $(x_k)_{k\in\mathbb{N}}$ qui converge vers u_{n+1} à partir d'un x_0 bien choisit selon la définition par récurrence suivante :

$$x_{k+1} = x_k - \frac{g_n(x_k)}{g_n'(x_k)} = x_k - \frac{x_k + hx_k^m - (h\cos(t_{n+1}) + u_n)}{1 + mhx_k^{m-1}}.$$

Le premier pas de la méthode de NEWTON s'écrit donc

$$x_1 = x_0 - \frac{x_0 + hx_0^m - (h\cos(t_{n+1}) + u_n)}{1 + mhx_0^{m-1}}.$$

Choisissons $x_0 = u_n$ comme valeur de départ (un autre choix pourrait être $x_0 = u_n + h\varphi(t_n, u_n)$). Nous pouvons utiliser x_1 comme approximation de u_{n+1} et on obtient le schéma

$$\begin{cases} u_0 = 0, \\ u_{n+1} = -\dfrac{u_n + hu_n^m - (h\cos(t_{n+1}) + u_n)}{1 + mhu_n^{m-1}}. \end{cases}$$

Exercice 4.8

Soit le problème de CAUCHY :

$$\begin{cases} y'(t) + 10y(t) = 0, \quad \forall t \in \mathbb{R}, \\ y(0) = y_0 > 0. \end{cases} \tag{4.4}$$

1. Montrer qu'il existe une unique solution globale $y \in \mathscr{C}^\infty(\mathbb{R}, \mathbb{R})$ que vous préciserez explicitement.
2. Soit le schéma numérique de CRANK-NICOLSON défini par la suite $(u_n)_{n\in\mathbb{N}}$ vérifiant

$$\frac{u_{n+1} - u_n}{h} + 5(u_{n+1} + u_n) = 0, \qquad \forall n \in \mathbb{N},$$

pour $h > 0$ fixé.
Montrer que la suite $(u_n)_{n\in\mathbb{N}}$ est une suite géométrique dont on précisera la raison.
3. Montrer que la raison r de la suite vérifie $|r| < 1$ pour tout $h > 0$. Ce schéma est-il inconditionnellement A-stable ?
4. Sous quelle condition sur $h > 0$ le schéma génère-t-il une suite positive ?
5. Donner l'expression de u_n en fonction de n.
6. Soit $T > 0$ fixé, soit n^* tel que $T - h < n^* h \leq T$ (donc n^* dépend de h). Montrer que

$$\lim_{h\to 0} u_{n^*} = y_0 e^{-10T}.$$

7. Soit $(v_n)_{n\in\mathbb{N}}$ la suite définissant le schéma d'EULER explicite pour l'équation différentielle (4.4). Montrer que

$$\lim_{h\to 0} v_{n^*} = y_0 e^{-10T}.$$

Montrer que u_{n^*} converge plus vite que v_{n^*} vers $y(t_{n^*}) = y_0 e^{-10T}$ lorsque $h \to 0$.

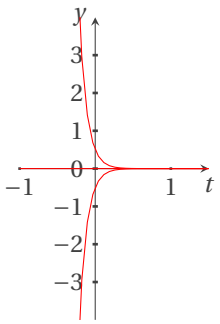
CORRECTION DE L'EXERCICE 4.8. C'est un problème de CAUCHY du type

$$\begin{cases} y'(t) = \varphi(t, y(t)), & \forall t \in \mathbb{R}, \\ y(0) = y_0 > 0, \end{cases} \tag{4.5}$$

avec $\varphi(t, y) = g(y) = -10y$.

1. On montre d'abord qu'il existe une et une seule solution locale (*i.e.* sur $[-T; T]$) de classe $\mathscr{C}^1([-T, T], \mathbb{R})$. On montre ensuite que cette solution est de classe $\mathscr{C}^\infty([-T, T], \mathbb{R})$. On montre enfin que la solution admet un prolongement sur $\mathbb{R}$.
   - ⋆ Comme $g \in \mathscr{C}^1(\mathbb{R}, \mathbb{R})$, d'après le théorème de CAUCHY-LIPSCHITZ il existe $T > 0$ et une unique solution $y \in \mathscr{C}^1([-T, T], \mathbb{R})$.
   - ⋆ Par récurrence, en exploitant l'EDO et la régularité de g, on grimpe en régularité sur y ainsi $y \in \mathscr{C}^\infty([-T, T], \mathbb{R})$.
   - ⋆ La fonctionne nulle est solution de l'EDO (mais non du problème de CAUCHY donné). Par l'unicité de la solution du problème de CAUCHY on en déduit que soit $y(t) > 0$ pour tout $t \in [-T, T]$ (*i.e.* lorsque $y_0 > 0$) soit $y(t) < 0$ pour tout $t \in [-T, T]$ (*i.e.* lorsque $y_0 > 0$). De plus, y est décroissante si $y_0 > 0$ et croissante si $y_0 < 0$. On en déduit par le théorème des extrémités que la solution u admet un prolongement sur $\mathbb{R}$ solution de l'EDO.

   Pour en calculer la solution, on remarque qu'il s'agit d'une EDO à variables séparables. L'unique solution constante est $y(t) \equiv 0$, toutes les autres solutions sont du type $y(t) = Ce^{-10t}$. En prenant en compte la condition initiale on conclut que l'unique solution du problème de CAUCHY est $y(t) = y_0 e^{-10t}$ définie pour tout $t \in \mathbb{R}$.



2. Soit le schéma numérique de CRANK-NICOLSON défini par la suite $(u_n)_{n\in\mathbb{N}}$ vérifiant

$$\frac{u_{n+1} - u_n}{h} + 5(u_{n+1} + u_n) = 0, \qquad \forall n \in \mathbb{N},$$

pour $h > 0$ fixé. On obtient une formule de récurrence rendue explicite par un calcul élémentaire :

$$u_{n+1} = -5hu_{n+1} - 5hu_n + u_n$$

d'où

$$u_{n+1} = \frac{1 - 5h}{1 + 5h} u_n.$$

Il s'agit d'une suite géométrique de raison

$$r = \frac{1 - 5h}{1 + 5h}.$$

3. Pour tout $h > 0$ on a

$$r = \frac{1 - 5h}{1 + 5h} = 1 - \frac{10h}{1 + 5h}$$

et

$$-1 < 1 - \frac{10h}{1 + 5h} < 1.$$

Ce schéma est donc inconditionnellement A-stable car $|u_{n+1}| = |r^{n+1} u_0| \leq |u_0|$.

4. Le schéma génère une suite positive ssi
$$1-\frac{10h}{1+5h}>0$$
i.e. ssi
$$h<\frac{1}{5}.$$

5. Par récurrence on obtient
$$u_n=\left(\frac{1-5h}{1+5h}\right)^n u_0.$$

6. Soit $T>0$ fixé et considérons n^* (dépendant de h) tel que $T-h<n^*h\leq T$. En se rappelant que
$$\lim_{x\to 0}\frac{\ln(1+\alpha x)}{\alpha x}=1$$
et en observant que
$$\begin{array}{ccccc}
\left(\frac{1-5h}{1+5h}\right)^{\frac{T}{h}-1} & \leq & \left(\frac{1-5h}{1+5h}\right)^{n^*} & \leq & \left(\frac{1-5h}{1+5h}\right)^{\frac{T}{h}} \\
\| & & & & \| \\
e^{(T-h)\frac{\ln(1-5h)-\ln(1+5h)}{h}} & & & & e^{T\frac{\ln(1-5h)-\ln(1+5h)}{h}} \\
\| & & & & \| \\
e^{(T-h)\frac{-5\ln(1-5h)-5\ln(1+5h)}{5h}} & & & & e^{T\frac{-5\ln(1-5h)-5\ln(1+5h)}{5h}} \\
\downarrow & & & & \downarrow \\
e^{-10T} & & & & e^{-10T}
\end{array}$$
on conclut que
$$\lim_{h\to 0}u_{n^*}=u_0\lim_{h\to 0}\left(\frac{1-5h}{1+5h}\right)^{n^*}=u_0e^{-10T}.$$

7. La suite définissant le schéma d'EULER explicite pour l'EDO assignée s'écrit
$$\frac{v_{n+1}-v_n}{h}=\varphi(t_n,v_n)\qquad\Longrightarrow\qquad v_{n+1}=v_n-10hv_n=(1-10h)v_n=(1-10h)^{n+1}v_0.$$
Il s'agit à nouveau d'une suite géométrique de raison
$$r_e=1-10h$$
qui converge si et seulement si $|r_e|<1$, *i.e.* si et seulement si $h<0{,}2$ (le schéma d'EULER pour cette EDO est conditionnellement stable).
Soit $T>0$ fixé et considérons n^* (dépendant de h) tel que $T-h<n^*h\leq T$. Alors
$$\begin{array}{ccccc}
(1-10h)^{\frac{T}{h}-1} & \leq & (1-10h)^{n^*} & \leq & (1-10h)^{\frac{T}{h}} \\
\| & & & & \| \\
e^{(T-h)\frac{\ln(1-10h)}{h}} & & & & e^{T\frac{\ln(1-10h)}{h}} \\
\| & & & & \| \\
e^{-10(T-h)\frac{\ln(1-10h)}{-10h}} & & & & e^{-10T\frac{\ln(1-10h)}{-10h}} \\
\downarrow & & & & \downarrow \\
e^{-10T} & & & & e^{-10T}
\end{array}$$
d'où
$$\lim_{h\to 0}v_{n^*}=u_0\lim_{h\to 0}(1-10h)^{\frac{T}{h}}=u_0e^{-10T}.$$
De plus, on sait (cf. cours) que la suite $(u_n)_{n\in\mathbb{N}}$ converge à l'ordre 2 tandis que la suite $(v_n)_{n\in\mathbb{N}}$ converge à l'ordre 1.

Exercice 4.9

Soit le problème de CAUCHY :
$$\begin{cases}y'(t)+\dfrac{\sqrt{y(t)}}{2}=0, & \forall t\in\mathbb{R}^+,\\ y(0)=u_0>0.\end{cases}$$

1. Soit le schéma numérique défini par la suite $(u_n)_{n\in\mathbb{N}}$ suivante
$$\frac{u_{n+1}-u_n}{h}+\frac{u_{n+1}}{2\sqrt{u_n}}=0, \qquad \forall n\in\mathbb{N},$$
pour $h>0$ fixé. Expliciter l'expression de u_{n+1} en fonction de u_n.
2. Montrer que la suite $(u_n)_{n\in\mathbb{N}}$ est une suite positive, décroissante et convergente vers 0.

CORRECTION DE L'EXERCICE 4.9.
C'est un problème de CAUCHY du type
$$\begin{cases} y'(t)=\varphi(t,y(t)), & \forall t\in\mathbb{R},\\ y(0)=y_0>0, \end{cases}$$
avec $\varphi(t,y)=g(y)=-\frac{\sqrt{y}}{2}$.

1. Pour $h>0$ fixé on obtient une formule de récurrence rendue explicite par un calcul élémentaire :
$$u_{n+1}=\frac{u_n}{1+\frac{h}{2\sqrt{u_n}}}=\frac{2(u_n)^{3/2}}{2\sqrt{u_n}+h}, \qquad \forall n\in\mathbb{N}.$$
2. On étudie la suite
$$\begin{cases} u_0>0,\\ u_{n+1}=\frac{2(u_n)^{3/2}}{2\sqrt{u_n}+h}, & \forall n\in\mathbb{N}, \end{cases}$$
pour $h>0$ fixé.
Par récurrence on montre que si $u_0>0$ alors $u_n>0$ pour tout $n\in\mathbb{N}$. De plus, $\frac{u_{n+1}}{u_n}=\frac{1}{1+\frac{h}{2\sqrt{u_n}}}<1$ pour tout $n\in\mathbb{N}$: la suite est monotone décroissante. On a alors une suite décroissante et bornée par zéro, donc elle converge. Soit ℓ la limite de cette suite, alors $\ell\geq 0$ et $\ell=\frac{2\ell^{3/2}}{2\sqrt{\ell}+h}$ donc $\ell=0$.

Exercice 4.10

Soit le problème de CAUCHY :
$$\begin{cases} y'(t)+y^5(t)=0, & \forall t\in\mathbb{R}^+,\\ y(0)=y_0>0. \end{cases} \tag{4.6}$$

1. Montrer qu'il existe une unique solution globale $y\in\mathscr{C}^\infty(\mathbb{R}^+,\mathbb{R}^+)$.
2. Soit le schéma numérique défini par la suite $(u_n)_{n\in\mathbb{N}}$ suivante
$$\frac{u_{n+1}-u_n}{h}+u_{n+1}u_n^4=0, \qquad \forall n\in\mathbb{N},$$
pour $h>0$ fixé. Expliciter l'expression de u_{n+1} en fonction de u_n.
3. Montrer que la suite $(u_n)_{n\in\mathbb{N}}$ est une suite décroissante et convergente vers 0 pour tout $h>0$ fixé.

CORRECTION DE L'EXERCICE 4.10.
C'est un problème de CAUCHY du type
$$\begin{cases} y'(t)=\varphi(t,y(t)), & \forall t\in\mathbb{R},\\ y(0)=y_0>0, \end{cases} \tag{4.7}$$
avec $\varphi(t,y)=g(y)=-y^5$.

1. On montre d'abord qu'il existe une et une seule solution locale (*i.e.* sur $[-T;T]$) de classe $\mathscr{C}^1([0,T],\mathbb{R})$. On montre ensuite que cette solution est de classe $\mathscr{C}^\infty([0,T],\mathbb{R})$. On montre enfin que la solution admet un prolongement sur $\mathbb{R}_+$.
   - ⋆ Comme $g\in\mathscr{C}^1(\mathbb{R}^+,\mathbb{R}^+)$, d'après le théorème de CAUCHY-LIPSCHITZ il existe $T>0$ et une unique solution $y\in\mathscr{C}^1([0,T],\mathbb{R})$.
   - ⋆ Par récurrence, en exploitant l'EDO et la régularité de g, on grimpe en régularité sur y ainsi $y\in\mathscr{C}^\infty([0,T],\mathbb{R})$.
   - ⋆ La fonctionne nulle est solution de l'EDO (mais non du problème de CAUCHY donné). Comme $y_0>0$, par l'unicité de la solution du problème de CAUCHY on a $y(t)>0$ pour tout $t\in[0,T]$ (car deux trajectoires ne peuvent pas se croiser). De plus, y est décroissante, ainsi la solution est bornée ($y(t)\in]0,y_0[$). On en déduit par le théorème des extrémités que la solution y admet un prolongement sur $\mathbb{R}^+$ solution de l'EDO. [1]

1. On peut montrer que l'unique solution du problème (4.6) est la fonction $y\colon[0,T]\to\mathbb{R}$ définie par $y(t)=y_0(4ty_0^4+1)^{-1/4}$.

2. Pour $h>0$ fixé on obtient une formule de récurrence rendue explicite par un calcul élémentaire :
$$u_{n+1}=\frac{u_n}{1+u_n^4 h},\qquad \forall n\in\mathbb{N}.$$
3. On étudie la suite
$$\begin{cases} u_0=y_0>0,\\ u_{n+1}=\frac{u_n}{1+u_n^4 h}, & \forall n\in\mathbb{N},\end{cases}$$
pour $h>0$ fixé.
Par récurrence on montre que si $u_0>0$ alors $u_n>0$ pour tout $n\in\mathbb{N}$. De plus, $\frac{u_{n+1}}{u_n}=\frac{1}{1+u_n^4 h}<1$ pour tout $n\in\mathbb{N}$: la suite est monotone décroissante. On a alors une suite décroissante et bornée par zéro, donc elle converge. Soit ℓ la limite de cette suite, alors $\ell\geq 0$ et $\ell=\frac{\ell}{1+\ell^4 h}$ donc $\ell=0$. Ce schéma est donc inconditionnellement A-stable.

Exercice 4.11

Soit le problème de CAUCHY :
$$\begin{cases} y'(t)+\sin(y(t))=0, & \forall t\in\mathbb{R},\\ y(0)=y_0>0.\end{cases} \tag{4.8}$$

1. Montrer qu'il existe une unique solution globale $y\in\mathscr{C}^\infty(\mathbb{R},\mathbb{R})$.
2. Écrire le schéma le schéma d'EULER explicite pour ce problème de CAUCHY en explicitant vos notations.
3. Montrer que la suite $(u_n)_{n\in\mathbb{N}}$ construite par ce schéma vérifie
$$|u_{n+1}|\leq|u_n|+h,\qquad \forall n\in\mathbb{N},$$
où $h>0$ est le pas de la suite.
4. En déduire que
$$|u_n|\leq|u_0|+nh,\qquad \forall n\in\mathbb{N}.$$

CORRECTION DE L'EXERCICE 4.11.
C'est un problème de CAUCHY du type
$$\begin{cases} y'(t)=\varphi(t,y(t)), & \forall t\in\mathbb{R},\\ y(0)=y_0>0,\end{cases} \tag{4.9}$$
avec $\varphi(t,y)=g(y)=-\sin(y)$.

1. Comme $g\in\mathscr{C}^1(\mathbb{R},\mathbb{R})$, d'après CAUCHY-LIPSCHITZ, il existe $T>0$ et une unique solution $y\in\mathscr{C}^1([-T,T],\mathbb{R})$. Par récurrence, en exploitant l'EDO et la régularité de g, on grimpe en régularité sur y et $y\in\mathscr{C}^\infty([-T,T],\mathbb{R})$.
Toutes les fonctions constante $y(t)=k\pi$ pour $k\in\mathbb{Z}$ sont solutions de l'équation différentielle car $g(k\pi)=0$. Pour y_0 donné, soit $\tilde{k}\in\mathbb{Z}$ tel que $y_0\in[\tilde{k}\pi;(\tilde{k}+1)\pi]$; par l'unicité de la solution du problème de CAUCHY on a $y(t)\in[\tilde{k}\pi;(\tilde{k}+1)\pi]$ pour tout $t\in[-T,T]$ (car deux trajectoires ne peuvent pas se croiser), *i.e.* la solution est bornée. On en déduit par le théorème des extrémités que la solution y admet un prolongement sur $\mathbb{R}$ solution de l'EDO.
2. Soit $h>0$ fixé et $t_n=nh$ pour tout $n\in\mathbb{Z}$. Le schéma d'EULER explicite pour l'EDO donnée construit la suite
$$u_{n+1}=u_n-h\sin(u_n),\qquad \forall n\in\mathbb{N}.$$
3. Comme $|a+b|\leq|a|+|b|$ et comme $|-\sin(x)|\leq 1$ pour tout $x\in\mathbb{R}$, on conclut que
$$|u_{n+1}|=|u_n-h\sin(u_n)|\leq|u_n|+|h\sin(u_n)|\leq|u_n|+h$$
pour $h>0$ fixé.
4. Par récurrence : $|u_{n+1}|\leq|u_n|+h\leq|u_{n-1}|+2h\leq\cdots\leq|u_0|+(n+1)h$.

Exercice 4.12 *Loi de* NEWTON

Considérons une tasse de café à la température de 75°C dans une salle à 25°C. On suppose que la température du café suit la loi de Newton, c'est-à-dire que la vitesse de refroidissement du café est proportionnelle à la différence des températures. En formule cela signifie qu'il existe une constante $K<0$ telle que la température vérifie l'équation différentielle

ordinaire (EDO) du premier ordre.

$$T'(t) = K(T(t) - 25).$$

La condition initiale (CI) est donc simplement

$$T(0) = 75.$$

Pour calculer la température à chaque instant on a besoin de connaître la constante K. Cette valeur peut être déduite en constatant qu'après 5 minutes le café est à 50°C, c'est-à-dire

$$T(5) = 50.$$

Calculer la solution exacte de ce problème de CAUCHY et la comparer avec la solution approchée obtenue par la méthode d'EULER explicite.

CORRECTION DE L'EXERCICE 4.12.

Solution exacte 1. On commence par calculer toutes les solutions de l'EDO. Étant une équation différentielle du premier ordre, la famille de solutions dépendra d'une constante qu'on fixera en utilisant la CI. Il s'agit d'une EDO à variables séparables donc formellement on a

$$T'(t) = K(T(t)-25) \implies \frac{T'(t)}{(T(t)-25)} = K \implies \frac{\mathrm{d}\,T}{(T-25)} = K\,\mathrm{d}\,t \implies$$
$$\int \frac{1}{(T-25)}\,\mathrm{d}\,T = K\int \mathrm{d}\,t \implies \ln(T-25) = Kt + c \implies T - 25 = De^{Kt} \implies T(t) = 25 + De^{Kt}$$

2. La valeur numérique de la constante d'intégration D est obtenue grâce à la CI :

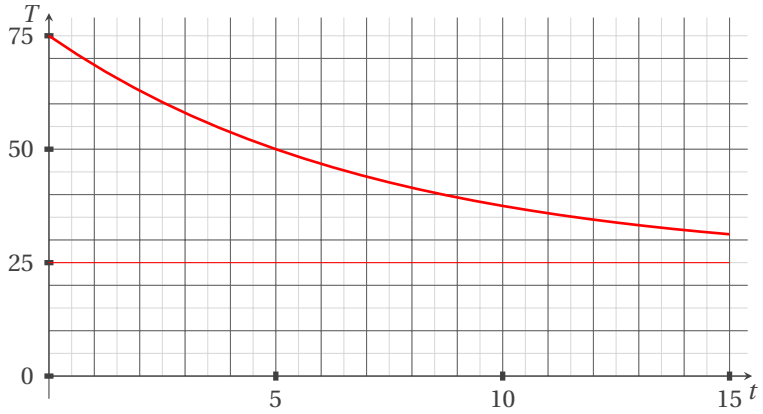
$$75 = T(0) = 25 + De^{K\cdot 0} \implies D = 50 \implies T(t) = 25 + 50e^{Kt}$$

3. Il ne reste qu'à établir la valeur numérique de la constante de refroidissement K grâce à l'«indice» :

$$50 = T(5) = 25 + 50e^{Kt} \implies K = -\frac{\ln(2)}{5} \approx -0.14 \implies T(t) = 25 + 50e^{-\frac{\ln(2)}{5}t}$$

On peut donc conclure que la température du café évolue selon la fonction

$$T(t) = 25 + 50e^{-\frac{\ln(2)}{5}t}.$$



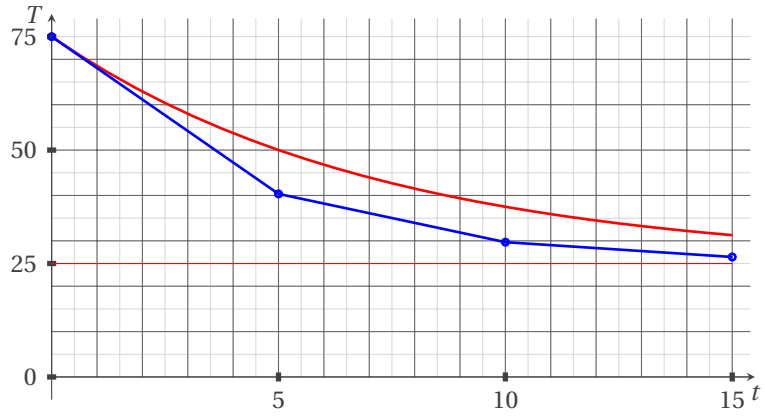
Solution approchée par la méthode d'Euler progressive Supposons de connaître K mais de ne pas vouloir/pouvoir calculer la fonction $T(t)$. Grâce à la méthode d'EULER on peut estimer la température à différentes instantes t_i en faisant une discrétisation temporelle du futur (i.e. on construit une suite de valeurs $\{t_i = 0 + i\Delta t\}_i$) et en construisant une suite de valeurs $\{T_i\}_i$ où chaque T_i est une approximation de $T(t_i)$. Si on utilise la méthode d'EULER, cette suite de température est ainsi construite :

$$\begin{cases} T_{i+1} = T_i - \frac{\ln(2)}{5}\Delta t\,(T_i - 25), \\ T_0 = 75, \end{cases}$$

qu'on peut réécrire comme

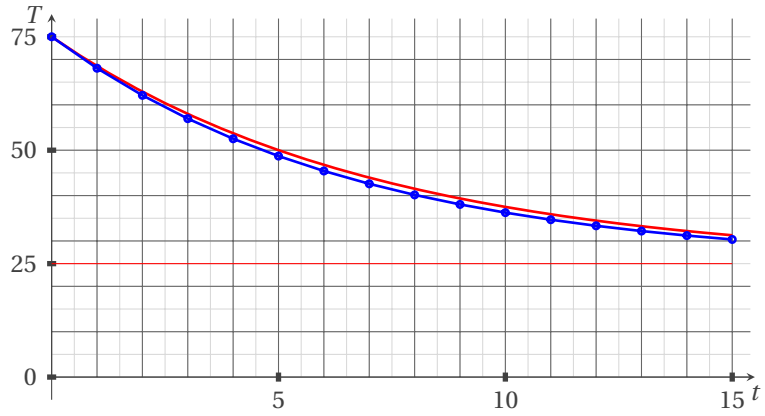
$$\begin{cases} T_{i+1} = (1 - \frac{\ln(2)}{5}\Delta t)\,T_i + 5\ln(2)\Delta t, \\ T_0 = 75. \end{cases}$$

1. Exemple avec $\Delta t = 5$:



t_i	$T(t_i)$	T_i	$T(t_i) - T_i$
0.000000	75.000000	75.000000	0.000000
5.000000	50.000000	40.342641	9.657359
10.000000	37.500000	29.707933	7.792067
15.000000	31.250000	26.444642	4.805358

2. Exemple avec $\Delta t = 1$:



t_i	$T(t_i)$	T_i	$T(t_i) - T_i$
0.000000	75.000000	75.000000	0.000000
1.000000	68.527528	68.068528	0.459000
2.000000	62.892914	62.097962	0.794952
3.000000	57.987698	56.955093	1.032605
4.000000	53.717459	52.525176	1.192283
5.000000	50.000000	48.709377	1.290623
6.000000	46.763764	45.422559	1.341205
7.000000	43.946457	42.591391	1.355066
8.000000	41.493849	40.152707	1.341142
9.000000	39.358729	38.052095	1.306634
10.000000	37.500000	36.242691	1.257309
11.000000	35.881882	34.684123	1.197759
12.000000	34.473229	33.341618	1.131610
13.000000	33.246924	32.185225	1.061700
14.000000	32.179365	31.189141	0.990224
15.000000	31.250000	30.331144	0.918856

Exercice 4.13 *«Les experts - Toulon»*

La loi de Newton affirme que la vitesse de refroidissement d'un corps est proportionnelle à la différence entre la température du corps et la température externe, autrement dit qu'il existe une constante $K < 0$ telle que la température du

corps suit l'équation différentielle

$$\begin{cases} T'(t) = K(T(t) - T_{\text{ext}}), \\ T(0) = T_0. \end{cases}$$

1. Soit Δt le pas temporel. Écrire le schéma d'EULER implicite pour approcher la solution de cette équation différentielle.
2. Soit $T_{\text{ext}} = 0°\text{C}$. En déduire une forme du type

$$T_{n+1} = g(\Delta t, n, T_0)$$

avec $g(\Delta t, n, T_0)$ à préciser (autrement dit, l'itéré en t_n ne dépend que de Δt, de n et de T_0). Que peut-on en déduire sur la convergence de la méthode ?
3. *Problème.* Un homicide a été commis. On veut établir l'heure du crime sachant que
   - ⋆ pour un corps humaine on peut approcher $K \approx -0.007438118376$ (l'échelle du temps est en minutes et la température en Celsius),
   - ⋆ le corps de la victime a été trouvé sur le lieu du crime à $2H20$ du matin,
   - ⋆ à l'heure du décès la température du corps était de 37°C,
   - ⋆ à l'heure de la découverte la température du corps est de 20°C,
   - ⋆ la température externe est $T_{\text{ext}} = 0°\text{C}$.

   Approcher l'heure de l'homicide en utilisant le schéma d'EULER implicite avec $\Delta t = 10$ minutes.
4. Pour cette équation différentielle, il est possible de calculer analytiquement ses solutions. Comparer alors la solution exacte avec la solution approchée obtenue au point précédent.

CORRECTION DE L'EXERCICE 4.13.

1. La méthode d'EULER implicite (ou régressive) est une méthode d'intégration numérique d'EDO du premier ordre de la forme $T'(t) = F(t, T(t))$. En choisissant un pas de discrétisation Δt, nous obtenons une suite de valeurs (t_n, T_n) qui peuvent être une excellente approximation de la fonction $T(t)$ avec

$$\begin{cases} t_n = t_0 + n\Delta t, \\ T_{n+1} = T_n + F(t_{n+1}, T_{n+1})\Delta t. \end{cases}$$

La méthode d'EULER implicite pour cette EDO s'écrit donc

$$T_{n+1} = T_n + K\Delta t(T_{n+1} - T_{\text{ext}}).$$

2. Si $T_{\text{ext}} = 0°\text{C}$, en procédant par récurrence sur n on obtient

$$T_{n+1} = g(\Delta t, n) = \frac{1}{1 - K\Delta t} T_n = \frac{1}{(1 - K\Delta t)^{n+1}} T_0,$$

autrement dit, l'itérée en t_n ne dépend que de Δt et de n mais ne dépend pas de T_n. Comme $0 < \frac{1}{1-K\Delta t} < 1$ pour tout $\Delta t > 0$, la suite est positive décroissante ce qui assure que la solution numérique est stable et convergente.

3. On cherche combien de minutes se sont écoulés entre le crime et la découverte du corps, autrement dit on cherche n tel que

$$20 = \frac{1}{(1 - K\Delta t)^{n+1}} 37 \implies (1 - K\Delta t)^{n+1} = \frac{37}{20} \implies n + 1 = \log_{(1-K\Delta t)}\left(\frac{37}{20}\right) = \frac{\ln\left(\frac{37}{20}\right)}{\ln(1 - K\Delta t)} \implies n \approx 8.$$

Comme $t_n = t_0 + n\Delta t$, si $t_n = 2H20$ alors $t_0 = t_n - n\Delta t = 2H20 - 1H20 = 01H00$.

4. Calcule analytique de toutes les solutions de l'équation différentielle :
   - ⋆ On cherche d'abord les solutions constantes, i.e. les solutions du type $T(t) \equiv c \in \mathbb{R}$ quelque soit t. On a

$$0 = K(c - T_{\text{ext}})$$

d'où l'unique solution constante $T(t) \equiv T_{\text{ext}}$.
   - ⋆ Soit $T(t) \neq T_{\text{ext}}$ quelque soit t. Puisqu'il s'agit d'une EDO à variables séparables on peut calculer la solution comme suit :

$$\begin{aligned} T'(t) = K(T(t) - T_{\text{ext}}) &\implies \frac{T'(t)}{T(t) - T_{\text{ext}}} = K \implies \frac{\mathrm{d}T}{T - T_{\text{ext}}} = K\mathrm{d}t \implies \\ \int \frac{1}{T - T_{\text{ext}}} \mathrm{d}T = K \int \mathrm{d}t &\implies \ln(T - T_{\text{ext}}) = Kt + c \implies T - T_{\text{ext}} = De^{Kt} \implies T(t) = T_{\text{ext}} + De^{Kt}. \end{aligned}$$

La valeur numérique de la constante d'intégration D est obtenue grâce à la CI :

$$T_0 = T(0) = De^{K\cdot 0} \qquad \Longrightarrow \qquad D = -T_0 \qquad \Longrightarrow \qquad T(t) = T_0 e^{Kt}$$

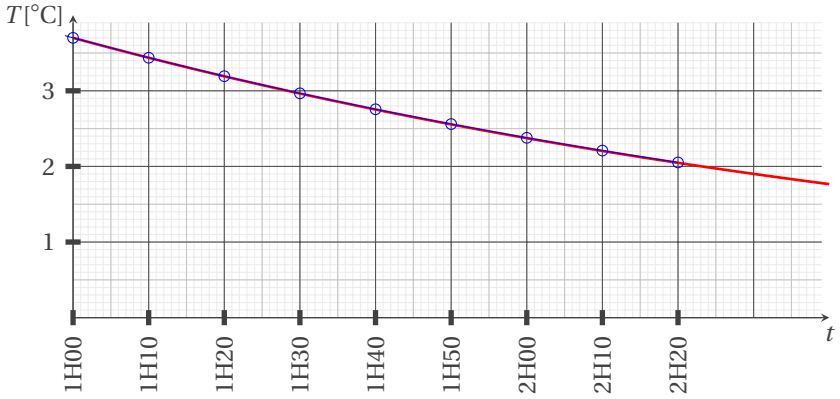
Ici $T_0 = 37°\text{C}$ donc la température du cadavre suit la loi

$$T(t) = 37e^{Kt}.$$

Pour déterminer l'heure du meurtre il faut alors résoudre l'équation

$$20 = 37e^{Kt}$$

d'où $t = \frac{1}{K}\ln\frac{20}{37} \approx 82{,}70715903$ minutes, c'est-à-dire 83 minutes avant 2H20 : le crime a été commit à 00H57.



Exercice 4.14

Un modèle pour la diffusion d'une épidémie se base sur l'hypothèse que sa vitesse de propagation est proportionnelle au nombre d'individus infectés et au nombre d'individus sains.
Si on note $I(t) \geq 0$ le nombre d'individus infectés à l'instant $t \geq 0$ et $A > 0$ le nombre d'individus total, il existe une constante $k \in \mathbb{R}^+$ telle que $I'(t) = kI(t)(A - I(t))$.

1. Montrer qu'il existe $T > 0$ et une unique solution $I \in \mathscr{C}^\infty([0,T])$ au problème de CAUCHY :

$$\begin{cases} I'(t) = kI(t)(A - I(t)), \\ I(0) = I_0 > 0. \end{cases}$$

2. Montrer que si $0 < I_0 < A$ alors $0 < I(t) < A$ pour tout $t > 0$.
3. Montrer que si $0 < I_0 < A$ alors $I(t)$ est croissante sur $\mathbb{R}^+$.
4. Soit $0 < I_0 < A$. On considère le schéma semi-implicite

$$\frac{I_{n+1} - I_n}{\Delta t} = kI_n(A - I_{n+1}).$$

Montrer que $I_n \to A$ lorsque $n \to +\infty$ indépendamment du pas $h > 0$ fixé.

CORRECTION DE L'EXERCICE 4.14. C'est un problème de CAUCHY du type

$$\begin{cases} I'(t) = \varphi(t, I(t)), \quad \forall\, t \in \mathbb{R}^+, \\ I(0) = I_0 > 0, \end{cases} \tag{4.10}$$

avec $\varphi(t, I(t)) = g(I(t)) = kI(t)(A - I(t))$.

1. Comme $g \in \mathscr{C}^1(\mathbb{R},\mathbb{R})$, d'après CAUCHY-LIPSCHITZ, il existe $T > 0$ et une unique $I \in \mathscr{C}^1([0,T],\mathbb{R})$ solution du problème de CAUCHY. Par récurrence, en exploitant l'EDO et la régularité de g, on grimpe en régularité sur I et $I \in \mathscr{C}^\infty([0,T],\mathbb{R})$.
2. Puisque la fonction nulle et la fonction constante $I(t) = A$ sont solutions de l'équation différentielle, si $0 < I_0 < A$ alors $0 < I(t) < A$ pour tout $t \in [0,T]$ (car, par l'unicité de la solution du problème de CAUCHY, deux trajectoires ne peuvent pas se croiser).
3. Puisque $I'(t) = kI(t)(A - I(t))$, si $0 < I_0 < A$ alors I est croissante pour tout $t \in [0,T]$. On en déduit par le théorème des extrémités que la solution I admet un prolongement sur $\mathbb{R}^+$ solution de l'EDO et que I est croissante pour tout $t \in \mathbb{R}^+$.

4. Soit $0 < I_0 < A$. On considère le schéma semi-implicite

$$\frac{I_{n+1} - I_n}{\Delta t} = kI_n(A - I_{n+1})$$

pour $\Delta t > 0$ fixé. On obtient une formule de récurrence rendue explicite par un calcul élémentaire :

$$I_{n+1} = \frac{1 + kA\Delta t}{1 + kI_n\Delta t} I_n.$$

Si $0 < I_0 < A$ alors
⋆ $I_n > 0$ quelque soit n ;
⋆ I_n est majorée par A car

$$I_{n+1} \leq A \iff (1 + kA\Delta t) I_n \leq (1 + kI_n\Delta t) A \iff I_n \leq A$$

donc par récurrence $I_{n+1} \leq A$ quelque soit n ;
⋆ si $I_n \to \ell$ alors $\ell = \frac{1+kA\Delta t}{1+k\ell\Delta t}\ell$ donc $\ell = 0$ ou $\ell = A$;
⋆ I_n est une suite monotone croissante (encore par récurrence on montre que $|I_{n+1}| \geq |I_n| \geq \cdots \geq |I_0|$) ;
donc $I_n \to A$ lorsque $n \to +\infty$ indépendamment du pas $h > 0$ choisi.

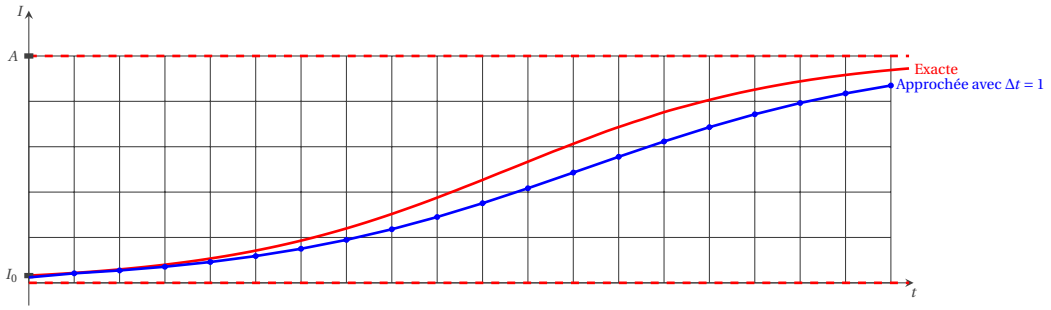
Calcul analytique de toutes les solutions :
On a déjà observé qu'il y a deux solutions constantes de l'EDO : la fonction $I(t) \equiv 0$ et la fonction $I(t) \equiv A$.
Pour chercher toutes les solutions non constantes on remarque qu'il s'agit d'une EDO à variables séparables donc on a

$$I(t) = \frac{A}{De^{-Akt} + 1}$$

La valeur numérique de la constante d'intégration D est obtenue grâce à la CI :

$$D = \frac{A - I_0}{I_0}$$

Exemple avec $A = 5000$, $I_0 = 160$, $k = \frac{\ln(363/38)}{35000}$ et $\Delta t = 1$:



Exercice 4.15

Considérons une population de bactéries. Soit $p(t)$ le nombre d'individus (≥ 0) à l'instant $t \geq 0$. Un modèle qui décrit l'évolution de cette population est l'«équation de la logistique» : soit k et h deux constantes positives, alors $p(t)$ vérifie l'équation différentielle ordinaire (EDO) du premier ordre

$$p'(t) = kp(t) - hp^2(t).$$

On veut calculer $p(t)$ à partir d'un nombre initiale d'individus donné

$$p(0) = p_0 \geq 0.$$

CORRECTION DE L'EXERCICE 4.15.

Solution exacte

1. On commence par calculer toutes les solutions de l'EDO. Étant une équation différentielle du premier ordre, la famille de solutions dépendra d'une constante qu'on fixera en utilisant la CI. Il s'agit d'une EDO à variables séparables.
   On cherche d'abord les solutions constantes, c'est-à-dire les solutions du type $p(t) \equiv c$ pour tout $t \in \mathbb{R}^+$:
   $$0 = kc - hc^2.$$
   On a donc deux solutions constantes :
   $$p(t) \equiv 0 \qquad \text{et} \qquad p(t) \equiv \frac{k}{h}.$$
   Étant donné que deux solutions d'une EDO ne s'intersectent jamais, dorénavant on supposera $p(t) \neq 0$ et $p(t) \neq \frac{k}{h}$ pour tout $t \in \mathbb{R}^+$, ainsi
   $$\frac{p'(t)}{kp(t) - hp^2(t)} = 1.$$
   Formellement on a
   $$\begin{aligned}
   \frac{\mathrm{d}p}{kp - hp^2} = 1\,\mathrm{d}t \quad &\Longrightarrow \quad \int \frac{1}{p(k-hp)}\,\mathrm{d}p = \int 1\,\mathrm{d}t \quad \Longrightarrow \\
   \frac{1}{k}\int \frac{1}{p}\,\mathrm{d}p - \frac{1}{k}\int \frac{-h}{k-hp}\,\mathrm{d}p = \int 1\,\mathrm{d}t \quad &\Longrightarrow \quad \frac{1}{k}\ln(p) - \frac{1}{k}\ln(k-hp) = t + c \quad \Longrightarrow \\
   \ln\left(\frac{p}{k-hp}\right) = kt + kc \quad &\Longrightarrow \quad \frac{p}{k-hp} = De^{kt} \quad \Longrightarrow \\
   p(t) = \frac{k}{\frac{1}{De^{kt}} + h}&.
   \end{aligned}$$

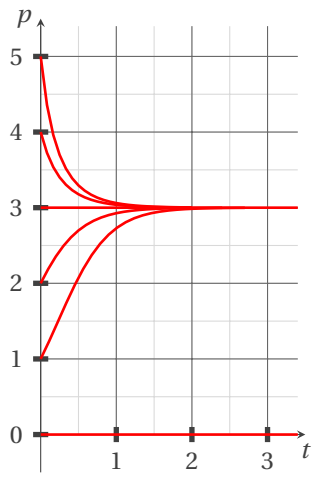
2. La valeur numérique de la constante d'intégration D est obtenue grâce à la CI :
   $$p_0 = p(0) = \frac{kD}{1 + hDe^{0k}} \qquad \Longrightarrow \qquad D = \frac{p_0}{k - hp_0}.$$

On peut donc conclure que la population évolue selon la fonction

$$p(t) = \begin{cases} 0 & \text{si } p_0 = 0, \\ \frac{k}{h} & \text{si } p_0 = \frac{k}{h}, \\ \frac{k}{\frac{k-hp_0}{p_0 e^{kt}} + h} & \text{sinon.} \end{cases}$$

Une simple étude de la fonction p montre que

- ⋆ si $p_0 \in]0; k/h[$ alors $p'(t) > 0$ et $\lim_{t\to+\infty} p(t) = k/h$,
- ⋆ si $p_0 \in]k/h; +\infty[$ alors $p'(t) < 0$ et $\lim_{t\to+\infty} p(t) = k/h$.



Exemple avec $k = 3$, $h = 1$ et différentes valeurs de p_0.

Solution approchée Supposons de ne pas vouloir/pouvoir calculer la fonction $p(t)$. Grâce à la méthode d'EULER on peut estimer le nombre d'indivus à différentes instantes t_i en faisant une discrétisation temporelle du futur (i.e. on construit une suite de valeurs $\{t_i = 0 + i\Delta t\}_i$) et en construisant une suite de valeurs $\{p_i\}_i$ où chaque p_i est une approximation de $p(t_i)$. Si on utilise la méthode d'EULER, cette suite est ainsi construite :

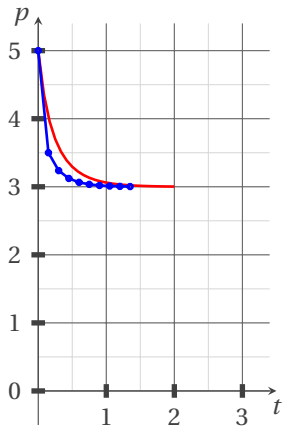
$$\begin{cases} p_{i+1} = p_i + \Delta t\ p_i(k - hp_i), \\ p_0 \text{ donné}, \end{cases}$$

qu'on peut réécrire comme

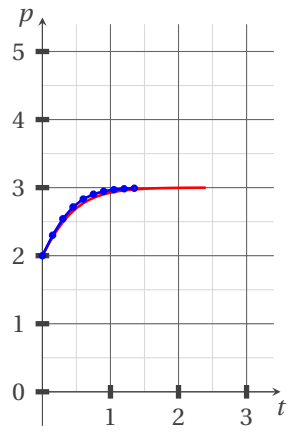
$$\begin{cases} p_{i+1} = (1 + k\Delta t - h\Delta t p_i)p_i, \\ p_0 \text{ donné}. \end{cases}$$

On veut appliquer cette méthode au cas de la figure précédente, i.e. avec $k = 3$, $h = 1$ et les valeurs initiales $p_0 = 5$ et $p_0 = 2$. Si on choisit comme pas temporelle $\Delta t = 0{,}15$, on obtient les figures suivantes :

t_i	$p(t_i)$	p_i	$p(t_i) - p_i$
0.00	5.000000	5.000000	0.000000
0.15	4.027123	3.500000	0.527123
0.30	3.582637	3.237500	0.345137
0.45	3.347079	3.122164	0.224915
0.60	3.212403	3.064952	0.147451
0.75	3.132046	3.035091	0.096956
0.90	3.082874	3.019115	0.063759
1.05	3.052319	3.010459	0.041861
1.20	3.033151	3.005736	0.027415
1.35	3.021054	3.003150	0.017904
1.50	3.013390	3.001731	0.011659
1.65	3.008524	3.000952	0.007573
1.80	3.005430	3.000523	0.004907



t_i	$p(t_i)$	p_i	$p(t_i) - p_i$
0.00	2.000000	2.000000	0.000000
0.15	2.274771	2.300000	−0.025229
0.30	2.493175	2.541500	−0.048325
0.45	2.655760	2.716292	−0.060532
0.60	2.770980	2.831887	−0.060907
0.75	2.849816	2.903298	−0.053483
0.90	2.902469	2.945411	−0.042942
1.05	2.937070	2.969529	−0.032459
1.20	2.959567	2.983102	−0.023535
1.35	2.974092	2.990663	−0.016571
1.50	2.983429	2.994852	−0.011423
1.65	2.989412	2.997164	−0.007752
1.80	2.993240	2.998439	−0.005199



Exercice 4.16 *Méthode de* TAYLOR

La méthode de TAYLOR est basé sur la relation

$$y(x+h) \simeq y(x) + y'(x)h + \frac{1}{2!}y''(x)h^2 + \frac{1}{3!}y'''(x)h^3 + \cdots + \frac{1}{m!}y^{(m)}(x)h^m$$

Cette relation prédit $y(x+h)$ à partir de $y(x)$, ainsi elle permet d'écrire une formule d'intégration numérique. Le dernier terme indique l'ordre de la méthode et l'erreur de troncature, due aux termes omis, est

$$E = \frac{1}{(m+1)!}y^{(m+1)}(\xi)h^{m+1} \qquad \text{pour } x < \xi < x+h,$$

que l'on peut approcher par

$$E \simeq \frac{h^m}{(m+1)!}\left(y^{(m)}(x+h) - y^{(m)}(x)\right).$$

Considérons le problème de CAUCHY

$$\begin{cases} y'(x) + 4y(x) = x^2, \\ y(0) = 1. \end{cases}$$

Estimer $y(0.1)$ par la méthode de TAYLOR d'ordre 4 avec un seul pas d'intégration.

CORRECTION DE L'EXERCICE 4.16. Le développement de TAYLOR en 0 jusqu'à l'ordre 4 est

$$y(h) \simeq y(0) + y'(0)h + \frac{1}{2!}y''(0)h^2 + \frac{1}{3!}y'''(0)h^3 + \frac{1}{4!}y^{IV}(0)h^4.$$

En dérivant l'EDO on trouve

$$\begin{aligned} & & y(0) &= 1, \\ y'(x) &= -4y(x) + x^2, & y'(0) &= -4, \\ y''(x) &= -4y'(x) + 2x, & y''(0) &= 16, \\ y'''(x) &= -4y''(x) + 2, & y'''(0) &= -62, \\ y^{IV}(x) &= -4y'''(x), & y^{IV}(0) &= 248. \end{aligned}$$

donc, pour $x = 0$ et $h = 0.1$, on obtient

$$y(0.1) \simeq 1 + \frac{-4}{10} + \frac{16}{200} + \frac{-62}{6000} + \frac{248}{240000} = 0.6707$$

et comme

$$y^{IV}(x+h) = -4y'''(x) = -4\left(-4y''(x)+2\right) = \left(-4\left(-4y'(x)+2x\right)+2\right) = \left(-4\left(-4\left(-4y(x)+x^2\right)+2x\right)+2\right)$$

alors $y^{IV}(0.1) \simeq \left(-4\left(-4\left(-4\times 0.6707 + (0.1)^2\right)+0.2\right)+2\right) = 166.259$ et on obtient l'estimation de l'erreur

$$E \simeq \frac{248}{960000}\left(y^{IV}(0.1) - y^{IV}(0)\right) = \frac{248}{960000}(166.259 - 248) = -0.000068.$$

5. Systèmes linéaires

Résoudre l'ensemble d'équations linéaires $\mathbb{A}\mathbf{x} = \mathbf{b}$

Définition *Définition : système linéaire*

Soit $n, p, \geq 1$ des entiers. Un SYSTÈME LINÉAIRE $n \times p$ est un ensemble de n équations linéaires à p inconnues de la forme

$$(S) \quad \begin{cases} a_{11}x_1 & + & \dots & + & a_{1p}x_p & = & b_1, \\ \vdots & & & & \vdots & & \vdots \\ a_{n1}x_1 & + & \dots & + & a_{np}x_p & = & b_n. \end{cases}$$

- Les COEFFICIENTS a_{ij} et les SECONDES MEMBRES b_i sont des éléments donnés de $\mathbb{R}$. Les INCONNUES $x_1, x_2, \dots, x_p$ sont à chercher dans $\mathbb{R}$.
- Le SYSTÈME HOMOGÈNE associé à (S) est le système obtenu en remplaçant les b_i par 0.
- Une SOLUTION de (S) est un p-uplet $(x_1, x_2, \dots, x_p)$ qui vérifie simultanément les n équations de (S). Résoudre (S) signifie chercher toutes les solutions.
- Un système est IMPOSSIBLE, ou incompatible, s'il n'admet pas de solution. Un système est POSSIBLE, ou compatible, s'il admet une ou plusieurs solutions.
- Deux systèmes sont ÉQUIVALENTS s'ils admettent les mêmes solutions.

Écriture matricielle

Si on note

$$\mathbf{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_p \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix} \qquad \mathbb{A} = \begin{pmatrix} a_{11} & \dots & a_{1p} \\ \vdots & & \vdots \\ a_{n1} & \dots & a_{np} \end{pmatrix}$$

le système (S) est équivalent à l'écriture matricielle $\mathbb{A}\mathbf{x} = \mathbf{b}$.

Dans ce chapitre, nous ne traiterons que des systèmes linéaires carrés d'ordre n à coefficients réels, autrement dit $\mathbb{A} = (a_{i,j}) \in \mathbb{R}^{n\times n}$ et $\mathbf{b} = (b_i) \in \mathbb{R}^n$. Dans ce cas, on est assuré de l'existence et de l'unicité de la solution si une des conditions équivalentes suivantes est remplie :

1. $\mathbb{A}$ est inversible (*i.e.* $\det(\mathbb{A}) \neq 0$) ;
2. le système homogène $\mathbb{A}\mathbf{x} = \mathbf{0}$ admet seulement la solution nulle.

La solution du système peut alors être calculée par la formule de CRAMER. Cependant cette formule est d'une utilité pratique limitée à cause du calcul des déterminants qui est très couteux. Pour cette raison, des méthodes numériques alternatives aux formules de CRAMER ont été développées. Elles sont dites *directes* si elles fournissent la solution du système en un nombre fini d'étapes, *itératives* si elles nécessitent (théoriquement) un nombre infini d'étapes. Notons dès à présent que le choix entre une méthode directe et une méthode itérative pour la résolution d'un système dépend non seulement de l'efficacité théorique des algorithmes, mais aussi du type de matrice, des capacités de stockage en mémoire et enfin de l'architecture de l'ordinateur.

Équilibrage de réactions chimiques

Du point de vue mathématique, équilibrer une réaction chimique signifie trouver des coefficients (dans $\mathbb{N}$ ou $\mathbb{Q}$), appelés coefficients stœchiométriques, qui satisfont certaines contraintes comme la conservation du nombre d'atomes, ou la conservation du nombre d'électrons (pour les réactions red-ox), ou la conservation de la charge (pour les réactions écrites sous forme ionique). Toutes ces contraintes dépendent linéairement des coefficients stœchiométriques, ce qui amène tout naturellement à l'écriture d'un système linéaire.

Par exemple, considérons la réaction

$$H_2 + O_2 \to H_2O.$$

Notons x_1, x_2 et x_3 les coefficients stœchiométriques

$$x_1 H_2 + x_2 O_2 \to x_3 H_2 O.$$

Les contraintes sont :
1. la conservation du nombre d'atomes d'hydrogène : $2x_1 = 2x_3$,
2. la conservation du nombre d'atomes d'oxygène : $2x_2 = x_3$.

On note qu'on a 3 inconnues mais seulement 2 équations linéairement indépendantes ; en effet, les coefficients stœchiométriques ne définissent pas des quantités absolues mais seulement les rapports entre les différents éléments. Par conséquent, si (x_1, x_2, x_3) équilibre la réaction, alors tous les multiples entiers de (x_1, x_2, x_3) équilibrent aussi la réaction. Pour résoudre le problème sans paramètres, fixons arbitrairement un des coefficients, par exemple $x_3 = 1$. On doit alors resoudre le système linéaire

$$\begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

On trouve alors $x_1 = 1$ et $x_2 = 1/2$. Si nous voulons des coefficients stœchiométriques entiers, il suffit de multiplier tous les coefficients par 2 et on a ainsi

$$2H_2 + 1O_2 \rightarrow 2H_2O.$$

5.1. Systèmes mal conditionnés

Considérons le système de deux équations à deux inconnues suivant :

$$\begin{cases} 3.218613x_1 + 6.327917x_2 = 10.546530, \\ 3.141592x_1 + 4.712390x_2 = 7.853982. \end{cases}$$

Ce système et non singulier et sa solution est donnée par $x_1 = x_2 = 1$. Considérons maintenant un système d'équations voisin (le carré indique un changement de décimale) :

$$\begin{cases} 3.21861\boxed{1}x_1 + 6.327917x_2 = 10.546530, \\ 3.14159\boxed{4}x_1 + 4.712390x_2 = 7.85398\boxed{0}. \end{cases}$$

Ce système et non singulier et sa solution est donnée par $x_1 = -5$, $x_2 = 5$.

On voit donc que, bien que ces deux systèmes soient voisins, leurs solutions sont très différentes. On parle dans ce cas de systèmes mal conditionnés. Résoudre un système mal conditionné avec un ordinateur peut être une affaire délicate si l'ordinateur calcule avec trop peu de chiffres significatifs. Dans l'exemple précédent nous observons que, si l'ordinateur ne retient que 6 chiffres significatifs, il est complètement inespéré d'obtenir une solution raisonnablement proche de la solution.

Définition *Conditionnement d'une matrice*

Le conditionnement d'une matrice $\mathbb{A} \in \mathbb{R}^{n\times n}$ non singulière est défini par

$$K(\mathbb{A}) = \|\mathbb{A}\|\|\mathbb{A}^{-1}\| (\geq 1),$$

où $\|\cdot\|$ est une norme matricielle subordonnée. En général, $K(\mathbb{A})$ dépend du choix de la norme ; ceci est signalé en introduisant un indice dans la notation. Par exemple, on a les deux normes matricielles suivantes :

$$\|\mathbb{A}\|_1 = \max_{j=1,\dots,n} \sum_{i=1}^{n} |a_{ij}|, \qquad \|\mathbb{A}\|_\infty = \max_{i=1,\dots,n} \sum_{j=1}^{n} |a_{ij}|.$$

Remarque *Cas particulier*

Si $\mathbb{A}$ est symétrique et définie positive [a],

$$K_2(\mathbb{A}) = \|\mathbb{A}\|_2\|\mathbb{A}^{-1}\|_2 = \frac{\lambda_{\max}}{\lambda_{\min}}$$

où $\lambda_{\max}$ (resp. $\lambda_{\min}$) est la plus grande (resp. petite) valeur propre de $\mathbb{A}$.

a. $\mathbb{A} \in \mathbb{R}^{n\times n}$ est
- ⋆ symétrique si $a_{ij} = a_{ji}$ pour tout $i, j = 1, \dots, n$,
- ⋆ définie positive si pour tout vecteurs $\mathbf{x} \in \mathbb{R}^n$ avec $\mathbf{x} \neq \mathbf{0}$, $\mathbf{x}^T\mathbb{A}\mathbf{x} > 0$.

Considérons un système non singulier $\mathbb{A}\mathbf{x} = \mathbf{b}$. Si $\delta\mathbf{b}$ est une perturbation de $\mathbf{b}$ et si on résout $\mathbb{A}\mathbf{y} = \mathbf{b} + \delta\mathbf{b}$, on obtient par linéarité $\mathbf{y} = \mathbf{x} + \delta\mathbf{x}$ avec $\mathbb{A}\delta\mathbf{x} = \delta\mathbf{b}$. La question est de savoir s'il est possible de majorer l'erreur relative $\|\delta\mathbf{x}\|/\|\mathbf{x}\|$ sur la solution du système en fonction de l'erreur relative $\|\delta\mathbf{b}\|/\|\mathbf{b}\|$ commise sur le second membre. Il est possible de démontrer que

$$\frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}\|} \leq K(\mathbb{A}) \frac{\|\delta\mathbf{b}\|}{\|\mathbf{b}\|}$$

où $K(\mathbb{A})$ est le nombre de conditionnement de la matrice $\mathbb{A}$. On voit alors que plus le conditionnement de la matrice est grand, plus la solution du système linéaire est sensible aux perturbations des données. Cependant, le fait qu'un système linéaire soit bien conditionné n'implique pas nécessairement que sa solution soit calculée avec précision. Il faut en plus utiliser des algorithmes stables. Inversement, le fait d'avoir une matrice avec un grand conditionnement n'empêche pas nécessairement le système global d'être bien conditionné pour des choix particuliers du second membre.

Si $\|\delta\mathbf{b}\|/\|\mathbf{b}\|$ est de l'ordre de la précision relative $\eta = 10^{-p}$ du calculateur, alors $\|\delta\mathbf{x}\|/\|\mathbf{x}\|$ pourrait, au pire, être égal à

$$K(\mathbb{A})\eta = 10^{\log_{10}(K(\mathbb{A}))} 10^{-p} = 10^{\log_{10}(K(\mathbb{A})-p)}.$$

Si on calcul la solution du système linéaire avec un ordinateur à p chiffres significatifs en valeur décimale, on ne pourra pas garantir à priori plus de

$$E(p - \log_{10}(K(\mathbb{A})))$$

chiffres significatifs sur la solution. Si on applique cette règle au système linéaire de l'exemple, il est facile de vérifier que $K(\mathbb{A}) \simeq 10^7$, par conséquent nous pouvons perdre jusqu'à 7 chiffres significatifs lors de sa résolution. Il faut donc un ordinateur calculant avec 10 chiffres significatifs pour être sûr d'obtenir les 3 premiers chiffres de la solution.

Exemple

Un exemple bien connu de matrice mal conditionnée est la matrice de HILBERT d'ordre n définie par $a_{ij} = 1/(i+j-1)$ pour $1 \leq i, j \leq n$.

Attention

Un système linéaire ne change pas de solution si on change l'ordre des équations. Cependant, l'ordre des équations peut changer totalement la solution donnée par une méthode numérique !

5.2. Méthode (directe) d'élimination de GAUSS et factorisation $\mathbb{LU}$

Définition *Matrices et systèmes triangulaires*

On dit qu'une matrice carrée $\mathbb{A} = (a_{ij})_{1\leq i,j\leq n}$ est TRIANGULAIRE SUPÉRIEURE (respectivement triangulaire INFÉRIEURE) si $i > j \implies a_{ij} = 0$ (resp. si $i < j \implies a_{ij} = 0$).

Si la matrice est triangulaire supérieure (resp. triangulaire inférieure), on dira que le système linéaire est un système triangulaire supérieur (resp. triangulaire inférieur).

Pour résoudre le système triangulaire $\mathbb{A}\mathbf{x} = \mathbf{b}$,

- si $\mathbb{A}$ est une matrice triangulaire inférieure, on a $x_1 = \frac{b_1}{a_{11}}$ et on déduit les inconnues $x_2, x_3, \dots x_n$ grâce à la relation $x_i = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j \right)$;
- si $\mathbb{A}$ est une matrice triangulaire supérieure on a $x_n = \frac{b_n}{a_{nn}}$ et on déduit les inconnues $x_{n-1}, x_{n-2}, \dots x_1$ grâce à la relation $x_i = \frac{1}{a_{ii}} \left(b_i - \sum_{j=i+1}^{n} a_{ij} x_j \right)$.

Propriété

Le déterminant d'une matrice triangulaire est égal au produit des éléments diagonaux.

La méthode du pivot de GAUSS transforme le système $\mathbb{A}\mathbf{x} = \mathbf{b}$ en un système équivalent (c'est-à-dire ayant la même solution) de la forme $\mathbb{U}\mathbf{x} = \mathbf{y}$, où $\mathbb{U}$ est une matrice triangulaire supérieure et $\mathbf{y}$ est un second membre convenablement modifié. Enfin on résout le système triangulaire $\mathbb{U}\mathbf{x} = \mathbf{y}$.

Définition *Méthode du pivot de GAUSS*

Soit $\mathbb{A} = (a_{ij})_{\substack{1\leq i\leq n \\ 1\leq j\leq p}}$ la matrice des coefficients du système $\mathbb{A}\mathbf{x} = \mathbf{b}$.

Étape k : en permutant éventuellement deux lignes du système, on peut supposer $a_{kk} \neq 0$ (appelé pivot de l'étape k). On

transforme toutes les lignes L_i avec $i > k$ comme suit :

$$L_i \leftarrow L_i - \underbrace{\boxed{\frac{a_{ik}}{a_{kk}}}}_{=\ell_{ik}} L_k.$$

En réitérant le procédé pour k de 1 à n, on aboutit à un système triangulaire supérieur.

Exemple

Soit le système linéaire

$$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 = 1, \\ 2x_1 + 3x_2 + 4x_3 + x_4 = 2, \\ 3x_1 + 4x_2 + x_3 + 2x_4 = 3, \\ 4x_1 + x_2 + 2x_3 + 3x_4 = 4. \end{cases}$$

1. Résolution par la méthode du pivot de GAUSS :

$$\begin{cases} x_1+2x_2+3x_3+4x_4=1 \\ 2x_1+3x_2+4x_3+x_4=2 \\ 3x_1+4x_2+x_3+2x_4=3 \\ 4x_1+x_2+2x_3+3x_4=4 \end{cases} \xrightarrow[L_4\leftarrow L_4-4L_1]{\substack{L_2\leftarrow L_2-2L_1 \\ L_3\leftarrow L_3-3L_1}} \begin{cases} x_1+2x_2+3x_3+4x_4=1 \\ -x_2-2x_3-7x_4=0 \\ -2x_2-8x_3-10x_4=0 \\ -7x_2-10x_3-13x_4=0 \end{cases}$$

$$\xrightarrow[L_4\leftarrow L_4-7L_2]{\substack{L_3\leftarrow L_3-2L_2 \\ }} \begin{cases} x_1+2x_2+3x_3+4x_4=1 \\ -x_2-2x_3-7x_4=0 \\ -4x_3+4x_4=0 \\ 4x_3+36x_4=0 \end{cases} \xrightarrow{L_4\leftarrow L_4+L_3} \begin{cases} x_1+2x_2+3x_3+4x_4=1 \\ -x_2-2x_3-7x_4=0 \\ -4x_3+4x_4=0 \\ 40x_4=0 \end{cases}$$

donc $x_4 = 0$, $x_3 = 0$, $x_2 = 0$ et $x_1 = 1$.

2. Résolution par la méthode du pivot de GAUSS en écriture matricielle :

$$[\mathbb{A}|\mathbf{b}] = \left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 1 \\ 2 & 3 & 4 & 1 & 2 \\ 3 & 4 & 1 & 2 & 3 \\ 4 & 1 & 2 & 3 & 4 \end{array}\right) \xrightarrow[L_4\leftarrow L_4-4L_1]{\substack{L_2\leftarrow L_2-2L_1 \\ L_3\leftarrow L_3-3L_1}} \left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 1 \\ 0 & -1 & -2 & -7 & 0 \\ 0 & -2 & -8 & -10 & 0 \\ 0 & -7 & -10 & -13 & 0 \end{array}\right)$$

$$\xrightarrow[L_4\leftarrow L_4-7L_2]{\substack{L_3\leftarrow L_3-2L_2 \\ }} \left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 1 \\ 0 & -1 & -2 & -7 & 0 \\ 0 & 0 & -4 & 4 & 0 \\ 0 & 0 & 4 & 36 & 0 \end{array}\right) \xrightarrow{L_4\leftarrow L_4+L_3} \left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 1 \\ 0 & -1 & -2 & -7 & 0 \\ 0 & 0 & -4 & 4 & 0 \\ 0 & 0 & 0 & 40 & 0 \end{array}\right)$$

donc $x_4 = 0$, $x_3 = 0$, $x_2 = 0$ et $x_1 = 1$.

Si on a plusieurs systèmes dont seul le second membre change, il peut être utile de factoriser une fois pour toute la matrice $\mathbb{A}$ et résoudre ensuite des systèmes triangulaires.

Algorithme de factorisation $\mathbb{LU}$ sans pivot

Soit le système linéaire $\mathbb{A}\mathbf{x} = \mathbf{b}$.

Factorisation On commence par factoriser la matrice $\mathbb{A} \in \mathbb{R}^{n\times n}$ sous la forme d'un produit de deux matrices $\mathbb{A} = \mathbb{LU}$. Les termes non nuls de $\mathbb{U}$ et les termes non nuls en-dessous de la diagonale principale de $\mathbb{L}$ sont mémorisés encore dans la matrice $\mathbb{A}$ et sont ainsi calculées :

for $k = 1$ to $n-1$ **do**
  for $i = k+1$ to n **do**
    $a_{ik} \leftarrow \dfrac{a_{ik}}{a_{kk}}$ {Il s'agit de ℓ_{ik} mémorisé dans a_{ik}}
    for $j = k+1$ to n **do**
      $a_{ij} \leftarrow a_{ij} - a_{ik}a_{kj}$ {Il s'agit de u_{ij} mémorisé dans a_{ij}}
    end for
  end for
end for

Résolution Résoudre le système linéaire revient maintenant à résoudre successivement

1. le **système triangulaire inférieur** $\mathbb{L}\mathbf{y} = \mathbf{b}$: les éléments non nuls de la matrice triangulaire inférieure $\mathbb{L}$ sont donné par $\ell_{ij} = a_{ij}$ pour $i = 1,\dots,n$ et $j = 1,\dots,i-1$ et $\ell_{ii} = 1$ pour tout $i = 1,\dots,n$, donc l'algorithme s'écrit

   $y_1 \leftarrow b_1$

for $i = 2$ to n **do**
  $s_i \leftarrow 0$
  for $j = 1$ to $i-1$ **do**
    $s_i \leftarrow s_i + a_{ij} y_j$
  end for
  $y_i \leftarrow b_i - s_i$
end for

2. le **système triangulaire supérieure** $\mathbb{U}\mathbf{x} = \mathbf{y}$: les éléments non nuls de la matrice triangulaire supérieure $\mathbb{U}$ sont donné par $u_{ij} = a_{ij}$ pour $i = 1,\dots,n$ et $j = i,\dots,n$, donc l'algorithme s'écrit

$x_n \leftarrow \dfrac{y_n}{a_{nn}}$
for $i = n-1$ to 1 by -1 **do**
  $s_i \leftarrow 0$
  for $j = 1$ to $i-1$ **do**
    $s_i \leftarrow s_i + a_{ij} y_j$
  end for
  $x_i \leftarrow \dfrac{y_i - s_i}{a_{ii}}$
end for

Attention

Pour une matrice quelconque $\mathbb{A} \in \mathbb{R}^{n\times n}$, la factorisation $\mathbb{LU}$ existe et est unique si et seulement si les sous-matrices principales $\mathbb{A}_i$ de $\mathbb{A}$ d'ordre $i = 1,\dots,n-1$ (celles que l'on obtient en restreignant $\mathbb{A}$ à ses i premières lignes et colonnes) ne sont pas singulières (autrement dit si les mineurs principaux, *i.e.* les déterminants des sous-matrices principales, sont non nuls).

On peut identifier des classes de matrices particulières pour lesquelles les hypothèses de cette proposition sont satisfaites :

Proposition

Si la matrice $\mathbb{A} \in \mathbb{R}^{n\times n}$ est symétrique et définie positive ou si est à diagonale dominante [a] alors la factorisation $\mathbb{LU}$ existe et est unique.

a. $\mathbb{A} \in \mathbb{R}^{n\times n}$ est
- ⋆ symétrique si $a_{ij} = a_{ji}$ pour tout $i, j = 1,\dots,n$,
- ⋆ définie positive si pour tout vecteurs $\mathbf{x} \in \mathbb{R}^n$ avec $\mathbf{x} \neq \mathbf{0}$, $\mathbf{x}^T \mathbb{A}\mathbf{x} > 0$,
- ⋆ à diagonale dominante par lignes si $|a_{ii}| \geq \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}|$, pour $i = 1,\dots,n$ (à diagonale dominante stricte par lignes si l'inégalité est stricte),
- ⋆ à diagonale dominante par colonnes si $|a_{ii}| \geq \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ji}|$, pour $i = 1,\dots,n$ (à diagonale dominante stricte par colonnes si l'inégalité est stricte),

Une technique qui permet d'effectuer la factorisation $\mathbb{LU}$ pour toute matrice $\mathbb{A}$ inversible, même quand les hypothèses de cette proposition ne sont pas vérifiées, est la méthode du pivot par ligne : il suffit d'effectuer une permutation convenable des lignes de la matrice originale $\mathbb{A}$ à chaque étape k où un terme diagonal a_{kk} s'annule.

Définition *Algorithme de* GAUSS *avec pivot*

Dans la méthode d'élimination de GAUSS les pivot $a_{kk}^{(k)}$ doivent être différents de zéro. Si la matrice est inversible mais un pivot est zéro (ou numériquement proche de zéro), on peut permuter deux lignes avant de poursuivre la factorisation. Concrètement, à chaque étape on cherche à avoir le pivot de valeur absolue la plus grande possible. L'algorithme modifié s'écrit alors

for $k = 1$ to $n-1$ **do**
  for $i = k+1$ to n **do**
    Chercher $\bar{r}$ tel que $|a_{\bar{r}k}^{(k)}| = \max_{r=k,\dots,n} |a_{rk}^{(k)}|$ et échanger la ligne k avec la ligne $\bar{r}$
    $\ell_{ik} \leftarrow \dfrac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$
    for $j = k+1$ to n **do**
      $a_{ij}^{(k+1)} \leftarrow a_{ij}^{(k)} - \ell_{ik}^{(k)} a_{kj}^{(k)}$
    end for
  end for
end for

Une fois calculées les matrices $\mathbb{L}$ et $\mathbb{U}$ et la matrice des permutations $\mathbb{P}$ (*i.e.* la matrice telle que $\mathbb{PA} = \mathbb{LU}$), résoudre le système linéaire consiste simplement à résoudre successivement le système triangulaire inférieur $\mathbb{L}\mathbf{y} = \mathbb{P}\mathbf{b}$ puis le système triangulaire supérieure $\mathbb{U}\mathbf{x} = \mathbf{y}$.

Propriété *Déterminant*

La factorisation $\mathbb{LU}$ permet de calculer le déterminant de $\mathbb{A}$ en $O(n^3)$ car $\det(\mathbb{A}) = \det(\mathbb{L})\det(\mathbb{U}) = \prod_{k=1}^{n} u_{kk}$.

Propriété *Inverse d'une matrice*

Le calcul explicite de l'inverse d'une matrice peut être effectué en utilisant la factorisation $\mathbb{LU}$ comme suit. En notant $\mathbb{X}$ l'inverse d'une matrice régulière $\mathbb{A} \in \mathbb{R}^{n\times n}$, les vecteurs colonnes de $\mathbb{X}$ sont les solutions des systèmes linéaires

$$\mathbb{A}\mathbf{x}_i = \mathbf{e}_i, \qquad \text{pour } i = 1, \dots, n.$$

En supposant que $\mathbb{PA} = \mathbb{LU}$, où $\mathbb{P}$ est la matrice de changement de pivot partiel, on doit résoudre $2n$ systèmes triangulaires de la forme

$$\mathbb{L}\mathbf{y}_i = \mathbb{P}\mathbf{e}_i, \qquad \mathbb{U}\mathbf{x}_i = \mathbf{y}_i, \qquad \text{pour } i = 1, \dots, n.$$

c'est-à-dire une suite de systèmes linéaires ayant la même matrice mais des seconds membres différents.

Exemple

Soit les systèmes linéaires

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \\ 3 & 4 & 1 & 2 \\ 4 & 1 & 2 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix} \quad \text{et} \quad \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \\ 3 & 4 & 1 & 2 \\ 4 & 1 & 2 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 10 \\ 10 \\ 10 \\ 10 \end{pmatrix}.$$

1. Résoudre les systèmes linéaires par la méthode du pivot de Gauss.
2. Factoriser la matrice $\mathbb{A}$ (sans utiliser la technique du pivot) et résoudre les systèmes linéaires.
3. Calculer le déterminant de $\mathbb{A}$.
4. Calculer $\mathbb{A}^{-1}$.

1. Résolution par la méthode du pivot de Gauss du premier système

$$[\mathbb{A}|\mathbf{b}] = \left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 1 \\ 2 & 3 & 4 & 1 & 2 \\ 3 & 4 & 1 & 2 & 3 \\ 4 & 1 & 2 & 3 & 4 \end{array}\right) \xrightarrow[L_4 \leftarrow L_4 - 4L_1]{\substack{L_2 \leftarrow L_2 - 2L_1 \\ L_3 \leftarrow L_3 - 3L_1}} \left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 1 \\ 0 & -1 & -2 & -7 & 0 \\ 0 & -2 & -8 & -10 & 0 \\ 0 & -7 & -10 & -13 & 0 \end{array}\right) \xrightarrow[L_4 \leftarrow L_4 - 7L_2]{L_3 \leftarrow L_3 - 2L_2} \left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 1 \\ 0 & -1 & -2 & -7 & 0 \\ 0 & 0 & -4 & 4 & 0 \\ 0 & 0 & 4 & 36 & 0 \end{array}\right)$$

$$\xrightarrow{L_4 \leftarrow L_4 + L_3} \left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 1 \\ 0 & -1 & -2 & -7 & 0 \\ 0 & 0 & -4 & 4 & 0 \\ 0 & 0 & 0 & 40 & 0 \end{array}\right)$$

donc

$$x_4 = 0, \quad x_3 = 0, \quad x_2 = 0, \quad x_1 = 1.$$

Résolution par la méthode du pivot de Gauss du second système

$$(\mathbb{A}|\mathbf{b}) = \left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 10 \\ 2 & 3 & 4 & 1 & 10 \\ 3 & 4 & 1 & 2 & 10 \\ 4 & 1 & 2 & 3 & 10 \end{array}\right) \xrightarrow[L_4 \leftarrow L_4 - 4L_1]{\substack{L_2 \leftarrow L_2 - 2L_1 \\ L_3 \leftarrow L_3 - 3L_1}} \left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 10 \\ 0 & -1 & -2 & -7 & -10 \\ 0 & -2 & -8 & -10 & -20 \\ 0 & -7 & -10 & -13 & -30 \end{array}\right) \xrightarrow[L_4 \leftarrow L_4 - 7L_2]{L_3 \leftarrow L_3 - 2L_2} \left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 10 \\ 0 & -1 & -2 & -7 & -10 \\ 0 & 0 & -4 & 4 & 0 \\ 0 & 0 & 4 & 36 & 40 \end{array}\right)$$

$$\xrightarrow{L_4 \leftarrow L_4 + L_3} \left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 10 \\ 0 & -1 & -2 & -7 & -10 \\ 0 & 0 & -4 & 4 & 0 \\ 0 & 0 & 0 & 40 & 40 \end{array}\right)$$

donc

$$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 = 10 \\ -x_2 - 2x_3 - 7x_4 = -10 \\ -4x_3 + 4x_4 = 0 \\ 40x_4 = 40 \end{cases} \implies x_4 = 1, \quad x_3 = 1, \quad x_2 = 1, \quad x_1 = 1.$$

2. Factorisation de la matrice $\mathbb{A}$:

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \\ 3 & 4 & 1 & 2 \\ 4 & 1 & 2 & 3 \end{pmatrix} \xrightarrow[L_4 \leftarrow L_4 - 4L_1]{\substack{L_2 \leftarrow L_2 - 2L_1 \\ L_3 \leftarrow L_3 - 3L_1}} \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & -1 & -2 & -7 \\ 3 & -2 & -8 & -10 \\ 4 & -7 & -10 & -13 \end{pmatrix} \xrightarrow[L_4 \leftarrow L_4 - 7L_2]{L_3 \leftarrow L_3 - 2L_2} \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & -1 & -2 & -7 \\ 3 & 2 & -4 & 4 \\ 4 & 7 & 4 & 36 \end{pmatrix} \xrightarrow{L_4 \leftarrow L_4 + L_3} \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & -1 & -2 & -7 \\ 3 & 2 & -4 & 4 \\ 4 & 7 & -1 & 40 \end{pmatrix}$$

donc

$$\mathbb{L} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 2 & 1 & 0 \\ 4 & 7 & -1 & 1 \end{pmatrix} \qquad \mathbb{U} = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & -1 & -2 & -7 \\ 0 & 0 & -4 & 4 \\ 0 & 0 & 0 & 40 \end{pmatrix}$$

Pour résoudre le premier système linéaire on résout les systèmes triangulaires $\mathbb{L}\mathbf{y} = \mathbf{b}$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 2 & 1 & 0 \\ 4 & 7 & -1 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix} \quad \Longrightarrow \quad y_1 = 1, \quad y_2 = 0, \quad y_3 = 0, \quad y_4 = 0$$

et $\mathbb{U}\mathbf{x} = \mathbf{y}$

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & -1 & -2 & -7 \\ 0 & 0 & -4 & 4 \\ 0 & 0 & 0 & 40 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad \Longrightarrow \quad x_4 = 0, \quad x_3 = 0, \quad x_2 = 0, \quad x_1 = 1.$$

Pour résoudre le second système linéaire on résout les systèmes triangulaires $\mathbb{L}\mathbf{y} = \mathbf{b}$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 2 & 1 & 0 \\ 4 & 7 & -1 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} = \begin{pmatrix} 10 \\ 10 \\ 10 \\ 10 \end{pmatrix} \quad \Longrightarrow \quad y_1 = 10, \quad y_2 = -10, \quad y_3 = 0, \quad y_4 = 40$$

et $\mathbb{U}\mathbf{x} = \mathbf{y}$

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & -1 & -2 & -7 \\ 0 & 0 & -4 & 4 \\ 0 & 0 & 0 & 40 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 10 \\ -10 \\ 0 \\ 40 \end{pmatrix} \quad \Longrightarrow \quad x_4 = 1, \quad x_3 = 1, \quad x_2 = 1, \quad x_1 = 1.$$

3. Le déterminant de $\mathbb{A}$ est $u_{11}u_{22}u_{33}u_{44} = 1 \times (-1) \times (-4) \times 40 = 160$.
4. Pour calculer $\mathbb{A}^{-1}$ on résout les quatre systèmes linéaires

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \\ 3 & 4 & 1 & 2 \\ 4 & 1 & 2 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \text{ i.e. } \begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 2 & 1 & 0 \\ 4 & 7 & -1 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \Longrightarrow \begin{pmatrix} 1 \\ -2 \\ 1 \\ 11 \end{pmatrix} \text{ puis } \begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & -1 & -2 & -7 \\ 0 & 0 & -4 & 4 \\ 0 & 0 & 0 & 40 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 1 \\ -2 \\ 1 \\ 11 \end{pmatrix} \Longrightarrow \begin{pmatrix} -9/40 \\ 1/40 \\ 1/40 \\ 11/40 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \\ 3 & 4 & 1 & 2 \\ 4 & 1 & 2 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \text{ i.e. } \begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 2 & 1 & 0 \\ 4 & 7 & -1 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \Longrightarrow \begin{pmatrix} 0 \\ 1 \\ -2 \\ -9 \end{pmatrix} \text{ puis } \begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & -1 & -2 & -7 \\ 0 & 0 & -4 & 4 \\ 0 & 0 & 0 & 40 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ -2 \\ -9 \end{pmatrix} \Longrightarrow \begin{pmatrix} 1/40 \\ 1/40 \\ 11/40 \\ -9/40 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \\ 3 & 4 & 1 & 2 \\ 4 & 1 & 2 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \text{ i.e. } \begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 2 & 1 & 0 \\ 4 & 7 & -1 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \Longrightarrow \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} \text{ puis } \begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & -1 & -2 & -7 \\ 0 & 0 & -4 & 4 \\ 0 & 0 & 0 & 40 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} \Longrightarrow \begin{pmatrix} 1/40 \\ 11/40 \\ -9/40 \\ 1/40 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \\ 3 & 4 & 1 & 2 \\ 4 & 1 & 2 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \text{ i.e. } \begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 2 & 1 & 0 \\ 4 & 7 & -1 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \Longrightarrow \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \text{ puis } \begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & -1 & -2 & -7 \\ 0 & 0 & -4 & 4 \\ 0 & 0 & 0 & 40 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \Longrightarrow \begin{pmatrix} 11/40 \\ -9/40 \\ 1/40 \\ 1/40 \end{pmatrix}$$

et finalement

$$\mathbb{A}^{-1} = \begin{pmatrix} -9/40 & 1/40 & 1/40 & 11/40 \\ 1/40 & 1/40 & 11/40 & -9/40 \\ 1/40 & 11/40 & -9/40 & 1/40 \\ 11/40 & -9/40 & 1/40 & 1/40 \end{pmatrix} = \frac{1}{40} \begin{pmatrix} -9 & 1 & 1 & 11 \\ 1 & 1 & 11 & -9 \\ 11 & 11 & -9 & 1 \\ 11 & -9 & 1 & 1 \end{pmatrix}.$$

5.3. Méthodes itératives

Une méthode itérative pour le calcul de la solution d'un système linéaire $\mathbb{A}\mathbf{x} = \mathbf{b}$ avec $\mathbb{A} \in \mathbb{R}^{n\times n}$ est une méthode qui construit une suite de vecteurs $\mathbf{x}^{(k)} = (x_1^{(k)}, x_2^{(k)}, \ldots, x_n^{(k)})^T \in \mathbb{R}^n$ convergent vers le vecteur solution exacte $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ pour tout vecteur initiale $\mathbf{x}^{(0)} = (x_1^{(0)}, x_2^{(0)}, \ldots, x_n^{(0)})^T \in \mathbb{R}^n$ lorsque k tend vers $+\infty$. Dans ces notes on ne verra que deux

méthodes itératives :

- la méthode de JACOBI,
- la méthode de GAUSS-SEIDEL.

Définition *Méthode de* JACOBI

Soit $\mathbf{x}^0 = (x_1^0, x_2^0, \dots, x_n^0)$ un vecteur donné. La méthode de JACOBI définit la composante x_i^{k+1} du vecteur $\mathbf{x}^{k+1}$ à partir des composantes x_j^k du vecteur $\mathbf{x}^k$ pour $j \neq i$ de la manière suivante :

$$x_i^{k+1} = \frac{b_i - \sum\limits_{\substack{j=1 \\ j\neq i}}^{n} a_{ij} x_j^k}{a_{ii}}, \qquad i = 1, \dots, n$$

$$\mathbf{x}^{(k)} = \begin{pmatrix} x_1^{(k)} \\ x_2^{(k)} \\ \vdots \\ x_{i-1}^{(k)} \\ x_i^{(k)} \\ x_{i+1}^{(k)} \\ \vdots \\ x_n^{(k)} \end{pmatrix} \qquad \mathbf{x}^{(k+1)} = \begin{pmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ \vdots \\ x_{i-1}^{(k+1)} \\ x_i^{(k+1)} \\ x_{i+1}^{(k+1)} \\ \vdots \\ x_n^{(k+1)} \end{pmatrix}$$

Proposition

Si la matrice $\mathbb{A}$ est à diagonale dominante stricte, la méthode de JACOBI converge.

La méthode de GAUSS-SIDEL est une amélioration de la méthode de JACOBI dans laquelle les valeurs calculées sont utilisées au fur et à mesure du calcul et non à l'issue d'une itération comme dans la méthode de JACOBI.

Définition *Méthode de* GAUSS-SIDEL

Soit $\mathbf{x}^0 = (x_1^0, x_2^0, \dots, x_n^0)$ un vecteur donné. La méthode de GAUSS-SIDEL définit la composante x_i^{k+1} du vecteur $\mathbf{x}^{k+1}$ à partir des composantes x_j^{k+1} du vecteur $\mathbf{x}^{k+1}$ pour $j < i$ et des composantes x_j^k du vecteur $\mathbf{x}^k$ pour $j \geq i$ de la manière suivante :

$$x_i^{k+1} = \frac{b_i - \sum\limits_{j=1}^{i-1} a_{ij} x_j^{k+1} - \sum\limits_{j=i+1}^{n} a_{ij} x_j^k}{a_{ii}}, \qquad i = 1, \dots, n$$

$$\mathbf{x}^{(k)} = \begin{pmatrix} x_1^{(k)} \\ x_2^{(k)} \\ \vdots \\ x_{i-1}^{(k)} \\ x_i^{(k)} \\ x_{i+1}^{(k)} \\ \vdots \\ x_n^{(k)} \end{pmatrix} \qquad \mathbf{x}^{(k+1)} = \begin{pmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ \vdots \\ x_{i-1}^{(k+1)} \\ x_i^{(k+1)} \\ x_{i+1}^{(k+1)} \\ \vdots \\ x_n^{(k+1)} \end{pmatrix}$$

Proposition

Si la matrice $\mathbb{A}$ est à diagonale dominante stricte ou si elle est symétrique et définie positive, la méthode de GAUSS-SEIDEL converge.

Algorithmes

Ces algorithmes tentent de résoudre le système d'équations linéaires $\mathbb{A}\mathbf{x} = \mathbf{b}$ d'inconnue $\mathbf{x}$. La matrice $\mathbb{A}$, de taille $n \times n$, doit être inversible et le second membre $\mathbf{b}$ doit être de longueur n. Les itérations s'arrêtent quand le rapport entre la norme du k-ème residu est inférieure ou égale à TOLL, le nombre d'itérations effectuées est alors renvoyé dans iter. MaxITER est le nombre maximum d'itérations.

JACOBI

Require: $\mathbb{A} = (a_{ij})_{1\leq i,j\leq n}$, $\mathbf{b} = (b_i)_{1\leq i\leq n}$, $\mathbf{x}$, MaxITER, TOLL
  iter $\leftarrow 0$
  $r \leftarrow \|\mathbf{b} - \mathbb{A}\mathbf{x}\|$
  while (r >TOLL & iter<MaxITER) **do**
    iter $\leftarrow$ iter +1
    $\mathbf{y} \leftarrow \mathbf{x}$
    for i from 1 to n **do**
      $s \leftarrow 0$
      for j from 1 to $i-1$ **do**
        $s \leftarrow s + a_{ij} y_j$
      end for
      for j from $i+1$ to n **do**
        $s \leftarrow s + a_{ij} y_j$
      end for
      $x_i \leftarrow (b_i - s)/a_{ii}$
    end for
    $r \leftarrow \|\mathbf{b} - \mathbb{A}\mathbf{x}\|$
  end while

GAUSS-SEIDEL

Require: $\mathbb{A} = (a_{ij})_{1\leq i,j\leq n}$, $\mathbf{b} = (b_i)_{1\leq i\leq n}$, $\mathbf{x}$, MaxITER, TOLL
  iter $\leftarrow 0$
  $r \leftarrow \|\mathbf{b} - \mathbb{A}\mathbf{x}\|$
  while (r >TOLL & iter<MaxITER) **do**
    iter $\leftarrow$ iter +1
    $\mathbf{y} \leftarrow \mathbf{x}$
    for i from 1 to n **do**
      $s \leftarrow 0$
      for j from 1 to $i-1$ **do**
        $s \leftarrow s + a_{ij} x_j$
      end for
      for j from $i+1$ to n **do**
        $s \leftarrow s + a_{ij} y_j$
      end for
      $x_i \leftarrow (b_i - s)/a_{ii}$
    end for
    $r \leftarrow \|\mathbf{b} - \mathbb{A}\mathbf{x}\|$
  end while

Il n'y a pas de résultat général établissant que la méthode de GAUSS-SEIDEL converge toujours plus vite que celle de JACOBI. On peut cependant l'affirmer dans certains cas, comme le montre la proposition suivante

Proposition

Soit $\mathbb{A}$ une matrice tridiagonale de taille $n \times n$ inversible dont les coefficients diagonaux sont tous non nuls. Alors les méthodes de JACOBI et de GAUSS-SEIDEL sont soit toutes les deux convergentes soit toutes les deux divergentes. En cas de convergence, la méthode de GAUSS-SEIDEL est plus rapide que celle de JACOBI.

Exemple

Considérons le système linéaire

$$\begin{pmatrix} 4 & 2 & 1 \\ -1 & 2 & 0 \\ 2 & 1 & 4 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 4 \\ 2 \\ 9 \end{pmatrix}$$

mis sous la forme

$$\begin{cases} x = 1 - \frac{y}{2} - \frac{z}{4}, \\ y = 1 + \frac{x}{2}, \\ z = \frac{9}{4} - \frac{x}{2} - \frac{y}{4}. \end{cases}$$

Soit $\mathbf{x}^{(0)} = (0,0,0)$ le vecteur initial.

⋆ En calculant les itérées avec la méthode de JACOBI on trouve

$$\mathbf{x}^{(1)} = \begin{pmatrix} 1 - \frac{0}{2} - \frac{0}{4} \\ 1 + \frac{0}{2} \\ \frac{9}{4} - \frac{0}{2} - \frac{0}{4} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 9/4 \end{pmatrix}, \quad \mathbf{x}^{(2)} = \begin{pmatrix} 1 - \frac{1}{2} - \frac{9/4}{4} \\ 1 + \frac{1}{2} \\ \frac{9}{4} - \frac{1}{2} - \frac{1}{4} \end{pmatrix} = \begin{pmatrix} -1/16 \\ 3/2 \\ 3/2 \end{pmatrix}, \quad \mathbf{x}^{(3)} = \begin{pmatrix} 1 - \frac{3/2}{2} - \frac{3/2}{4} \\ 1 + \frac{-1/16}{2} \\ \frac{9}{4} - \frac{-1/16}{2} - \frac{3/2}{4} \end{pmatrix} = \begin{pmatrix} -1/8 \\ -1/32 \\ 61/32 \end{pmatrix}, \quad \mathbf{x}^{(4)} = \begin{pmatrix} 1 - \frac{-1/32}{2} - \frac{61/32}{4} \\ 1 + \frac{-1/8}{2} \\ \frac{9}{4} - \frac{-1/8}{2} - \frac{-1/32}{4} \end{pmatrix} = \begin{pmatrix} 5/128 \\ 15/16 \\ 265/128 \end{pmatrix}.$$

La suite $\mathbf{x}^{(k)}$ converge vers $(0,1,2)$ la solution du système.

⋆ En calculant les itérées avec la méthode de GAUSS-SEIDEL on trouve

$$\mathbf{x}^{(1)} = \begin{pmatrix} 1 - \frac{0}{2} - \frac{0}{4} \\ 1 + \frac{1}{2} \\ \frac{9}{4} - \frac{1}{2} - \frac{3/2}{4} \end{pmatrix} = \begin{pmatrix} 1 \\ 3/2 \\ 11/8 \end{pmatrix}, \quad \mathbf{x}^{(2)} = \begin{pmatrix} 1 - \frac{3/2}{2} - \frac{11/8}{4} \\ 1 + \frac{-3/32}{2} \\ \frac{9}{4} - \frac{-3/32}{2} - \frac{61/64}{4} \end{pmatrix} = \begin{pmatrix} -3/32 \\ 61/64 \\ 527/256 \end{pmatrix}, \quad \mathbf{x}^{(3)} = \begin{pmatrix} 1 - \frac{-3/32}{2} - \frac{61/64}{4} \\ 1 + \frac{9/1024}{2} \\ \frac{9}{4} - \frac{9/1024}{2} - \frac{2047/2048}{4} \end{pmatrix} = \begin{pmatrix} 9/1024 \\ 2047/2048 \\ 16349/8192 \end{pmatrix},$$

La suite $\mathbf{x}^{(k)}$ converge vers $(0,1,2)$ la solution du système.

Exercices

Exercice 5.1

Soit le système linéaire

$$\begin{pmatrix} 6 & 1 & 1 \\ 2 & 4 & 0 \\ 1 & 2 & 6 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 12 \\ 0 \\ 6 \end{pmatrix}.$$

1. Approcher la solution avec la méthode de JACOBI avec 3 itérations à partir de $\mathbf{x}^{(0)} = (2,2,2)$.
2. Approcher la solution avec la méthode de GAUSS-SEIDEL avec 3 itérations à partir de $\mathbf{x}^{(0)} = (2,2,2)$.
3. Résoudre les systèmes linéaires par la méthode d'élimination de GAUSS.
4. Factoriser la matrice $\mathbb{A}$ (sans utiliser la technique du pivot) et résoudre les systèmes linéaires.

CORRECTION DE L'EXERCICE 5.1.

1. Méthode de JACOBI :

$$\mathbf{x}^{(0)} = \begin{pmatrix} 2 \\ 2 \\ 2 \end{pmatrix}, \quad \mathbf{x}^{(1)} = \begin{pmatrix} \frac{12-(1\times 2+1\times 2)}{6} \\ \frac{0-(2\times 2+0\times 2)}{4} \\ \frac{6-(1\times 2+2\times 2)}{6} \end{pmatrix} = \begin{pmatrix} 4/3 \\ -1 \\ 0 \end{pmatrix}, \quad \mathbf{x}^{(2)} = \begin{pmatrix} \frac{12-(1\times(-1)+1\times 0)}{6} \\ \frac{0-(2\times\frac{4}{3}+0\times 0)}{4} \\ \frac{6-(1\times\frac{4}{3}+2\times(-1))}{6} \end{pmatrix} = \begin{pmatrix} 13/6 \\ -2/3 \\ 10/9 \end{pmatrix}, \quad \mathbf{x}^{(3)} = \begin{pmatrix} \frac{12-(1\times\frac{-2}{3}+1\times\frac{10}{9})}{6} \\ \frac{0-(2\times\frac{13}{6}+0\times\frac{10}{9})}{4} \\ \frac{6-(1\times\frac{13}{6}+2\times\frac{-2}{3})}{6} \end{pmatrix} = \begin{pmatrix} 52/27 \\ -13/12 \\ 31/36 \end{pmatrix}$$

ainsi

$$\mathbf{x} \approx \begin{pmatrix} 1.926 \\ -1.083 \\ 0.861 \end{pmatrix}.$$

2. Méthode de GAUSS-SEIDEL :

$$\mathbf{x}^{(0)} = \begin{pmatrix} 2 \\ 2 \\ 2 \end{pmatrix}, \quad \mathbf{x}^{(1)} = \begin{pmatrix} \frac{12-(1\times 2+1\times 2)}{6} \\ \frac{0-(2\times\frac{4}{3}+0\times 2)}{4} \\ \frac{6-(1\times\frac{4}{3}+2\times\frac{-2}{3})}{6} \end{pmatrix} = \begin{pmatrix} 4/3 \\ -\frac{2}{3} \\ 1 \end{pmatrix}, \quad \mathbf{x}^{(2)} = \begin{pmatrix} \frac{12-(1\times\frac{-2}{3}+1\times 1)}{6} \\ \frac{0-(2\times\frac{35}{18}+0\times 1)}{4} \\ \frac{6-(1\times\frac{35}{18}+2\times\frac{-35}{36})}{6} \end{pmatrix} = \begin{pmatrix} 35/18 \\ -35/36 \\ 1 \end{pmatrix}, \quad \mathbf{x}^{(3)} = \begin{pmatrix} \frac{12-(1\times\frac{35}{18}+1\times\frac{-35}{36})}{6} \\ \frac{0-(2\times\frac{431}{216}+0\times 1)}{4} \\ \frac{6-(1\times\frac{431}{216}+2\times\frac{-431}{432})}{6} \end{pmatrix} = \begin{pmatrix} 431/216 \\ -431/432 \\ 1 \end{pmatrix}$$

ainsi

$$\mathbf{x} \approx \begin{pmatrix} 1.995 \\ -0.995 \\ 1 \end{pmatrix}.$$

3. Méthode d'élimination de GAUSS :

$$(\mathbb{A}|\mathbf{b}) = \left(\begin{array}{ccc|c} 6 & 1 & 1 & 12 \\ 2 & 4 & 0 & 0 \\ 1 & 2 & 6 & 6 \end{array}\right) \xrightarrow[L_3 \leftarrow L_3 - \frac{1}{6}L_1]{L_2 \leftarrow L_2 - \frac{2}{6}L_1} \left(\begin{array}{ccc|c} 6 & 1 & 1 & 12 \\ 0 & \frac{11}{3} & -\frac{1}{3} & -4 \\ 0 & \frac{11}{6} & \frac{35}{6} & 4 \end{array}\right) \xrightarrow{L_3 \leftarrow L_3 - \frac{\frac{11}{6}}{\frac{11}{3}}L_2} \left(\begin{array}{ccc|c} 6 & 1 & 1 & 12 \\ 0 & \frac{11}{3} & -\frac{1}{3} & -4 \\ 0 & 0 & 6 & 6 \end{array}\right)$$

donc

$$\begin{cases} 6x_1 + x_2 + x_3 = 12, \\ \frac{11}{3}x_2 - \frac{1}{3}x_3 = -4 \\ 6x_3 = 6 \end{cases} \implies x_3 = 1, \quad x_2 = -1, \quad x_1 = 2.$$

4. Factorisation de la matrice $\mathbb{A}$:

$$\begin{pmatrix} 6 & 1 & 1 \\ 2 & 4 & 0 \\ 1 & 2 & 6 \end{pmatrix} \xrightarrow[L_3 \leftarrow L_3 - \frac{1}{6}L_1]{L_2 \leftarrow L_2 - \frac{2}{6}L_1} \begin{pmatrix} 6 & 1 & 1 \\ \frac{2}{6} & \frac{11}{3} & -\frac{1}{3} \\ \frac{1}{6} & \frac{11}{6} & \frac{35}{6} \end{pmatrix} \xrightarrow{L_3 \leftarrow L_3 - \frac{\frac{11}{6}}{\frac{11}{3}}L_2} \begin{pmatrix} 6 & 1 & 1 \\ \frac{2}{6} & \frac{11}{3} & -\frac{1}{3} \\ \frac{1}{6} & \frac{\frac{11}{6}}{\frac{11}{3}} & 6 \end{pmatrix}$$

donc

$$\mathbb{L} = \begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{3} & 1 & 0 \\ \frac{1}{6} & \frac{1}{2} & 1 \end{pmatrix} \qquad \mathbb{U} = \begin{pmatrix} 6 & 1 & 1 \\ 0 & \frac{11}{3} & -\frac{1}{3} \\ 0 & 0 & 6 \end{pmatrix}$$

Pour résoudre le système linéaire on résout les systèmes triangulaires $\mathbb{L}\mathbf{y} = \mathbf{b}$

$$\begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{3} & 1 & 0 \\ \frac{1}{6} & \frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 12 \\ 0 \\ 6 \end{pmatrix} \implies y_1 = 12, \quad y_2 = -4, \quad y_3 = 6$$

et $\mathbb{U}\mathbf{x} = \mathbf{y}$

$$\begin{pmatrix} 6 & 1 & 1 \\ 0 & \frac{11}{3} & -\frac{1}{3} \\ 0 & 0 & 6 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 1 \\ -4 \\ 6 \end{pmatrix} \qquad \Longrightarrow \qquad x_3 = 1, \quad x_2 = -1, \quad x_1 = 2.$$

Exercice 5.2

Soit $\mathbb{A}$ une matrice, $\mathbb{A} \in \mathscr{M}_{n,n}(\mathbb{R})$.

1. Rappeler les conditions nécessaires et suffisantes pour l'existence d'une factorisation $\mathbb{LU}$ de la matrice $\mathbb{A}$ et préciser les définitions de $\mathbb{L}$ et $\mathbb{U}$.
2. On suppose $\mathbb{L}$ et $\mathbb{U}$ construites (*i.e.* on dispose de tous les coefficients $\ell_{i,j}$ et $u_{i,j}$ de $\mathbb{L}$ et $\mathbb{U}$), écrire l'algorithme de résolution de $\mathbb{A}\mathbf{x} = \mathbf{b}$, avec $\mathbf{b} \in \mathscr{M}_{n,1}(\mathbb{R})$ donné.
3. Soit la matrice $\mathbb{A}$ suivante :
$$\begin{pmatrix} 3 & -1 & -1 \\ -1 & 3 & -1 \\ -1 & -1 & 3 \end{pmatrix}.$$
Construire à la main les matrices $\mathbb{L}$ et $\mathbb{U}$ de la factorisation $\mathbb{LU}$.

CORRECTION DE L'EXERCICE 5.2.

1. Pour une matrice quelconque $\mathbb{A} \in \mathscr{M}_{n,n}(\mathbb{R})$, la factorisation $\mathbb{LU}$ (sans pivot) existe et est unique ssi les sous-matrices principales $\mathbb{A}_i$ de $\mathbb{A}$ d'ordre $i = 1, \dots, n-1$ (celles que l'on obtient en restreignant $\mathbb{A}$ à ses i premières lignes et colonnes) ne sont pas singulières (autrement dit si les mineurs principaux, *i.e.* les déterminants des sous-matrices principales, sont non nuls). On peut identifier des classes de matrices particulières pour lesquelles les hypothèses de cette proposition sont satisfaites. Mentionnons par exemple : les matrices à diagonale strictement dominante, les matrices réelles symétriques définies positives. Une technique qui permet d'effectuer la factorisation $\mathbb{LU}$ pour toute matrice $\mathbb{A}$ inversible, même quand les hypothèses de cette proposition ne sont pas vérifiées, est la méthode du pivot par ligne : il suffit d'effectuer une permutation convenable des lignes de la matrice originale $\mathbb{A}$ à chaque étape k où un terme diagonal a_{kk} s'annule.
2. Une fois calculées les matrices $\mathbb{L}$ et $\mathbb{U}$, résoudre le système linéaire $\mathbb{A}\mathbf{x} = \mathbf{b}$, avec $\mathbf{b} \in \mathscr{M}_{n,1}(\mathbb{R})$ donné consiste simplement à résoudre successivement
   2.1. le système triangulaire inférieur $\mathbb{L}\mathbf{y} = \mathbf{b}$ par l'algorithme
   $$y_1 = b_1, \qquad y_i = b_i - \sum_{j=1}^{i-1} \ell_{ij} y_j, \qquad i = 2, \dots, n$$
   2.2. le système triangulaire supérieure $\mathbb{U}\mathbf{x} = \mathbf{y}$ par l'algorithme
   $$x_n = \frac{y_n}{u_{nn}}, \qquad x_i = \frac{1}{u_{ii}} \left(y_i - \sum_{j=i+1}^{n} u_{ij} x_j \right), \qquad j = n-1, \dots, 1$$
3. Factorisation :
$$\begin{pmatrix} 3 & -1 & -1 \\ -1 & 3 & -1 \\ -1 & -1 & 3 \end{pmatrix} \xrightarrow[L_3 \leftarrow L_3 - \frac{-1}{3} L_1]{L_2 \leftarrow L_2 - \frac{-1}{3} L_1} \begin{pmatrix} 3 & -1 & -1 \\ 0 & 8/3 & -4/3 \\ 0 & -4/3 & 8/3 \end{pmatrix} \xrightarrow{L_3 \leftarrow L_3 - \frac{-4/3}{8/3} L_2} \begin{pmatrix} 3 & -1 & -1 \\ 0 & 8/3 & -4/3 \\ 0 & 0 & 2 \end{pmatrix}.$$
Par conséquent
$$\mathbb{L} = \begin{pmatrix} 1 & 0 & 0 \\ -1/3 & 1 & 0 \\ -1/3 & -1/2 & 1 \end{pmatrix} \quad \text{et} \quad \mathbb{U} = \begin{pmatrix} 3 & -1 & -1 \\ 0 & 8/3 & -4/3 \\ 0 & 0 & 2 \end{pmatrix}.$$

Exercice 5.3

Calculer, lorsqu'il est possible, la factorisation $\mathbb{LU}$ des matrices suivantes :

$$\mathbb{A} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 7 & 8 & 9 \end{pmatrix}, \qquad \mathbb{B} = \begin{pmatrix} 1 & 2 & 3 \\ 7 & 8 & 9 \\ 2 & 4 & 5 \end{pmatrix}.$$

Comment peut-on modifier l'algorithme de factorisation pour pouvoir toujours aboutir à une factorisation $\mathbb{LU}$ lorsque la matrice est inversible ?

CORRECTION DE L'EXERCICE 5.3. Pour une matrice quelconque $\mathbb{A} \in \mathscr{M}_{n,n}(\mathbb{R})$, la factorisation $\mathbb{LU}$ (sans pivot) existe et est unique ssi les sous-matrices principales $\mathbb{A}_i$ de $\mathbb{A}$ d'ordre $i = 1, \dots, n-1$ (celles que l'on obtient en restreignant $\mathbb{A}$ à ses i premières lignes et colonnes) ne sont pas singulières (autrement dit si les mineurs principaux, *i.e.* les déterminants des sous-matrices principales, sont non nuls).

Matrice $\mathbb{A}$: comme $\det(\mathbb{A}) \neq 0$, la matrice $\mathbb{A}$ est bien inversible. Puisque $\det(\mathbb{A}_1) = a_{11} = 1 \neq 0$ mais $\det(\mathbb{A}_2) = a_{11}a_{22} - a12a_{21} = 0$, on ne peut pas factoriser $\mathbb{A}$ sans utiliser la technique du pivot. En effet,

$$\mathbb{A} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 7 & 8 & 9 \end{pmatrix} \xrightarrow[L_3 \leftarrow L_3 - \frac{7}{1} L_1]{L_2 \leftarrow L_2 - \frac{2}{1} L_1} \begin{pmatrix} 1 & 2 & 3 \\ 0 & 0 & -1 \\ 0 & -6 & -12 \end{pmatrix}$$

La factorisation $\mathbb{LU}$ ne peut pas être calculée car à la prochaine étape il faudrait effectuer le changement $L_3 \leftarrow L_3 - \frac{-6}{0} L_2$.

Matrice $\mathbb{B}$:

$$\mathbb{A}_2 = \begin{pmatrix} 1 & 2 & 3 \\ 7 & 8 & 9 \\ 2 & 4 & 5 \end{pmatrix} \xrightarrow[L_3 \leftarrow L_3 - \frac{2}{1} L_1]{L_2 \leftarrow L_2 - \frac{7}{1} L_1} \begin{pmatrix} 1 & 2 & 3 \\ 0 & -6 & -12 \\ 0 & 0 & -1 \end{pmatrix}$$

La factorisation $\mathbb{LU}$ de la matrice $\mathbb{B}$ est donc

$$\mathbb{L} = \begin{pmatrix} 1 & 0 & 0 \\ 7 & 1 & 0 \\ 2 & 0 & 1 \end{pmatrix}, \qquad \mathbb{U} = \begin{pmatrix} 1 & 2 & 3 \\ 0 & -6 & -12 \\ 0 & 0 & -1 \end{pmatrix}.$$

Lorsqu'un pivot est nul, la méthode de GAUSS pour calculer la factorisation $\mathbb{LU}$ de la matrice $\mathbb{A}$ n'est plus applicable. De plus, si le pivot n'est pas nul mais très petit, l'algorithme conduit à des erreurs d'arrondi importantes. C'est pourquoi des algorithmes qui échangent les éléments de façon à avoir le pivot le plus grand possible ont été développés. Les programmes optimisés intervertissent les lignes à chaque étape de façon à placer en pivot le terme de coefficient le plus élevé : c'est la méthode du pivot partiel. Pour la matrice $\mathbb{A}$ cela aurait donné

$$\mathbb{A} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 7 & 8 & 9 \end{pmatrix} \xrightarrow{L_2 \leftrightarrow L_3} \begin{pmatrix} 1 & 2 & 3 \\ 7 & 8 & 9 \\ 2 & 4 & 5 \end{pmatrix} \xrightarrow[L_3 \leftarrow L_3 - \frac{2}{1} L_1]{L_2 \leftarrow L_2 - \frac{7}{1} L_1} \begin{pmatrix} 1 & 2 & 3 \\ 0 & -6 & -12 \\ 0 & 0 & -1 \end{pmatrix}.$$

Bien évidemment, il faut garder trace de cet échange de lignes pour qu'il puisse être répercuté sur le terme source et sur l'inconnue lors de la résolution du système linéaire ; ceci est réalisé en introduisant une nouvelle matrice $\mathbb{P}$, dite matrice pivotale, telle que $\mathbb{PA} = \mathbb{LU}$: la résolution du système linéaire $\mathbb{A}\mathbf{x} = \mathbf{b}$ est donc ramené à la résolution des deux systèmes triangulaires $\mathbb{L}\mathbf{y} = \mathbb{P}\mathbf{b}$ et $\mathbb{U}\mathbf{x} = \mathbf{y}$. Dans notre exemple cela donne

$$\mathbb{P} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

Exercice 5.4

Soit α un paramètre réel et soient les matrices $\mathbb{A}_\alpha$, $\mathbb{P}$ et le vecteur $\mathbf{b}$ définis par

$$\mathbb{A}_\alpha = \begin{pmatrix} 2 & 4 & 1 \\ \alpha & -2 & -1 \\ 2 & 3 & 2 \end{pmatrix}, \qquad \mathbb{P} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \qquad \mathbf{b} = \begin{pmatrix} 0 \\ -3/2 \\ -1 \end{pmatrix}.$$

1. À quelle condition sur α, la matrice $\mathbb{A}_\alpha$ est inversible ?
2. À quelle condition sur α, la matrice $\mathbb{A}_\alpha$ admet-elle une décomposition $\mathbb{LU}$ (sans pivot) ?
3. Soit $\alpha = -1$. Calculer, si elle existe, la décomposition $\mathbb{LU}$ de la matrice $\mathbb{M} = \mathbb{P}\mathbb{A}_\alpha$.
4. Soit $\alpha = -1$. Résoudre le système linéaire $\mathbb{A}\mathbf{x} = \mathbf{b}$ en résolvant le système linéaire $\mathbb{M}\mathbf{x} = \mathbb{P}\mathbf{b}$.

CORRECTION DE L'EXERCICE 5.4.

1. La matrice $\mathbb{A}_\alpha$ est inversible si et seulement si $\det(\mathbb{A}) \neq 0$. Comme

$$\det(\mathbb{A}) = \det \begin{pmatrix} 2 & 4 & 1 \\ \alpha & -2 & -1 \\ 2 & 3 & 2 \end{pmatrix}$$

$$= (2\times(-2)\times 2) + (4\times(-1)\times 2) + (1\times\alpha\times 3) - (2\times(-1)\times 3) - (4\times\alpha\times 2) - (1\times(-2)\times 2)$$
$$= (-8) + (-8) + (3\alpha) - (-6) - (8\alpha) - (-4)$$
$$= -6 - 5\alpha,$$

la matrice $\mathbb{A}_\alpha$ est inversible si et seulement si $\alpha \neq -\frac{6}{5}$.

2. Pour une matrice $\mathbb{A}$ carrée d'ordre n quelconque, la factorisation de GAUSS existe et est unique si et seulement si les sous-matrices principales $\mathbb{A}_i$ de $\mathbb{A}$ d'ordre $i = 1,\dots,n-1$ (celles que l'on obtient en restreignant $\mathbb{A}$ à ses i premières lignes et colonnes) ne sont pas singulières (autrement dit si les mineurs principaux, *i.e.* les déterminants des sous-matrices principales, sont non nuls).
Pour la matrice $\mathbb{A}_\alpha$ on a les sous-matrices principales suivantes :

$$\mathbb{A}_1 = \begin{pmatrix}2\end{pmatrix}, \qquad \det(\mathbb{A}_1) = 2;$$
$$\mathbb{A}_2 = \begin{pmatrix}2 & 4\\ \alpha & -2\end{pmatrix}, \qquad \det(\mathbb{A}_2) = -4(1+\alpha).$$

Par conséquent, la matrice $\mathbb{A}_\alpha$ admet une décomposition $\mathbb{LU}$ (sans pivot) si et seulement si $\alpha \neq -1$.

3. Si $\alpha = -1$ la matrice $\mathbb{A}_\alpha$ n'admet pas de décomposition $\mathbb{LU}$ sans pivot. La matrice $\mathbb{P}$ échange les lignes 2 et 3 de la matrice $\mathbb{A}$ et on obtient la matrice

$$\mathbb{PA}_{-1} = \begin{pmatrix}1&0&0\\0&0&1\\0&1&0\end{pmatrix}\begin{pmatrix}2&4&1\\-1&-2&-1\\2&3&2\end{pmatrix} = \begin{pmatrix}2&4&1\\2&3&2\\-1&-2&-1\end{pmatrix}.$$

La matrice $\mathbb{M}$ admet une décomposition $\mathbb{LU}$ (sans pivot) et l'on a

$$\begin{pmatrix}2&4&1\\2&3&2\\-1&-2&-1\end{pmatrix} \xrightarrow[L_3\leftarrow L_3-\frac{-1}{2}L_1]{L_2\leftarrow L_2-L_1} \begin{pmatrix}2&4&1\\0&-1&1\\0&0&-\frac{1}{2}\end{pmatrix}$$

Par conséquent, on obtient la décomposition $\mathbb{LU}$ suivante de la matrice $\mathbb{M}$:

$$\mathbb{L} = \begin{pmatrix}1&0&0\\1&1&0\\-\frac{1}{2}&0&1\end{pmatrix}, \qquad \mathbb{U} = \begin{pmatrix}2&4&1\\0&-1&1\\0&0&-\frac{1}{2}\end{pmatrix}.$$

4. Pour résoudre le système linéaire $\mathbb{M}\mathbf{x} = \mathbb{P}\mathbf{b}$ il suffit de résoudre les deux systèmes triangulaires suivantes :
⋆ $\mathbb{L}\mathbf{y} = \mathbb{P}\mathbf{b}$:

$$y_1 = 0, \qquad y_2 = -1 - y_1 = -1, \qquad y_3 = -\frac{3}{2} + \frac{1}{2}y_1 = -\frac{3}{2};$$

⋆ $\mathbb{U}\mathbf{x} = \mathbf{y}$:

$$x_3 = \frac{-3}{2}(-2) = 3, \qquad x_2 = (-1 - x_3)/(-1) = 4, \qquad x_1 = (0 - x_2 - 4x_3)/2 = -\frac{19}{2}.$$

Exercice 5.5

Considérons les deux matrices carrées d'ordre $n > 3$:

$$\mathbb{A} = \begin{pmatrix}\alpha&0&0&0&&\dots&\beta\\0&\alpha&0&0&0&\dots&\beta\\0&0&\alpha&0&\ddots&&\vdots\\&0&\ddots&\ddots&&\dots&\beta\\\vdots&\vdots&&\ddots&&0&\beta\\0&0&&&0&\alpha&\beta\\\beta&\beta&\dots&&\beta&\beta&\alpha\end{pmatrix} \qquad \mathbb{B} = \begin{pmatrix}\beta&0&\dots&\dots&0&0&\alpha\\\beta&&0&0&0&\alpha&0\\\vdots&&&0&\ddots&&0\\&&&\ddots&&\dots&\vdots\\\vdots&0&\alpha&0&&0&0\\\beta&\alpha&0&&0&\alpha&0\\\alpha&\beta&\beta&\dots&&\beta&\beta\end{pmatrix}$$

avec α et β réels non nuls.

1. Vérifier que la factorisation $\mathbb{LU}$ de la matrice $\mathbb{B}$ ne peut pas être calculée sans utiliser la technique du pivot.
2. Calculer analytiquement le nombre d'opérations nécessaires pour calculer la factorisation $\mathbb{LU}$ de la matrice $\mathbb{A}$.

3. Exprimer le déterminant de la matrice $\mathbb{A}$ sous forme récursive en fonction des coefficients de la matrice et de sa dimension n.
4. Sous quelles conditions sur α et β la matrice $\mathbb{A}$ est définie positive ? Dans ce cas, exprimer le conditionnement de la matrice en fonction des coefficients et de la dimension n.

CORRECTION DE L'EXERCICE 5.5.

1. La factorisation $\mathbb{LU}$ de la matrice $\mathbb{B}$ ne peut pas être calculée sans utiliser la technique du pivot car l'élément pivotale au deuxième pas est nul. Par exemple, si $n=4$, on obtient :

$$\mathbb{B}^{(1)} = \begin{pmatrix} \beta & 0 & 0 & \alpha \\ \beta & 0 & \alpha & 0 \\ \beta & \alpha & 0 & 0 \\ \alpha & \beta & \beta & \beta \end{pmatrix} \xrightarrow[L_4 \leftarrow L_4 - \frac{\alpha}{\beta} L_1]{\substack{L_2 \leftarrow L_2 - L_1 \\ L_3 \leftarrow L_3 - L_1}} \mathbb{B}^{(2)} = \begin{pmatrix} \beta & 0 & 0 & \alpha \\ 0 & \boxed{0} & \alpha & -\alpha \\ 0 & \alpha & 0 & -\alpha \\ 0 & \beta & \beta & \beta - \frac{\alpha^2}{\beta} \end{pmatrix}.$$

2. La matrice $\mathbb{A}$ est une matrice «en flèche» : pour en calculer la factorisation $\mathbb{LU}$ il suffit de transformer la dernière ligne, ce qui requiert le calcul de l'unique multiplicateur $\ell_{nk} = \beta/\alpha$ et l'exécution de $n-1$ produits et sommes. Le coût globale est donc de l'ordre de n.
3. Le déterminant δ_n de la matrice $\mathbb{A}$ de dimension n coïncide avec le déterminant de la matrice $\mathbb{U}$. Comme $u_{ii} = \alpha$ pour tout $i < n$ et $u_{nn} = \alpha - (n-1)\beta^2/\alpha$, on conclut que

$$\delta_n = \prod_{i=1}^{n} u_{ii} = u_{nn} \cdot \prod_{i=1}^{n-1} u_{ii} = \left(\alpha - (n-1)\frac{\beta^2}{\alpha}\right)\alpha^{n-1} = \alpha^n - (n-1)\alpha^{n-2}\beta^2.$$

4. Les valeurs propres de la matrice $\mathbb{A}$ sont les racines du déterminant de la matrice $\mathbb{A} - \lambda\mathbb{I}$. Suivant le même raisonnement du point précédant, ce déterminant s'écrit

$$(\alpha-\lambda)^n - (n-1)(\alpha-\lambda)^{n-2}\beta^2$$

dont les racines sont

$$\lambda_{1,2} = \alpha \pm \sqrt{(n-1)\beta}, \qquad \lambda_3 = \cdots = \lambda_n = \alpha.$$

Par conséquent, pour que la matrice $\mathbb{A}$ soit définie positive il faut que les valeurs propres soient tous positifs, ce qui impose

$$\alpha > 0, \qquad |\beta| < \frac{\alpha}{\sqrt{n-1}}.$$

Dans ce cas, le conditionnement de la matrice en norme 2 est

$$K_2(\mathbb{A}) = \begin{cases} \frac{\alpha+\beta\sqrt{n-1}}{\alpha-\beta\sqrt{n-1}} & \text{si } \beta \geq 0, \\ \frac{\alpha-\beta\sqrt{n-1}}{\alpha+\beta\sqrt{n-1}} & \text{sinon.} \end{cases}$$

Exercice 5.6

Donner une condition suffisante sur le coefficient α pour avoir convergence des méthodes de JACOBI et GAUSS-SEIDEL pour la résolution d'un système linéaire associé à la matrice

$$\mathbb{A} = \begin{pmatrix} \alpha & 0 & 1 \\ 0 & \alpha & 0 \\ 1 & 0 & \alpha \end{pmatrix}$$

CORRECTION DE L'EXERCICE 5.6. Une condition suffisante pour la convergence des méthodes de JACOBI et de GAUSS-SEIDEL est que $\mathbb{A}$ est à diagonale strictement dominante, *i.e.* $\sum_{\substack{i=1 \\ i\neq j}}^{3} |a_{ij}| < |a_{ii}|$ pour $j = 1,2,3$. La matrice $\mathbb{A}$ vérifie cette condition si et seulement si $|\alpha| > 1$.

Exercice 5.7

Considérons le système linéaire $\mathbb{A}\mathbf{x} = \mathbf{b}$ avec

$$\mathbb{A} = \begin{pmatrix} \alpha & 0 & \gamma \\ 0 & \alpha & \beta \\ 0 & \delta & \alpha \end{pmatrix}$$

avec α, β, γ et δ des paramètres réels. Donner des conditions suffisantes sur les coefficients pour avoir

1. convergence de la méthode de JACOBI
2. convergence de la méthode de GAUSS-SEIDEL.

CORRECTION DE L'EXERCICE 5.7.

1. Une condition suffisante pour que la méthode de JACOBI converge est que la matrice soit à dominance diagonale stricte, ce qui équivaut à imposer
$$\begin{cases} |\alpha| > |\gamma|, \\ |\alpha| > |\beta|, \\ |\alpha| > |\delta|, \end{cases}$$
c'est-à-dire $|\alpha| > \max\{|\beta|, |\gamma|, |\delta|\}$.
2. La condition précédente est aussi suffisante pour la convergence de la méthode de GAUSS-SEIDEL. Une autre condition suffisante pour la convergence de cette méthode est que la matrice soit symétrique définie positive. Pour la symétrie il faut que
$$\begin{cases} \gamma = 0, \\ \beta = \delta, \end{cases}$$
on obtient ainsi la matrice
$$\mathbb{A} = \begin{pmatrix} \alpha & 0 & 0 \\ 0 & \alpha & \beta \\ 0 & \beta & \alpha \end{pmatrix}.$$
Elle est définie positive si ses valeurs propres sont positifs. On a
$$\lambda_1 = \alpha, \qquad \lambda_2 = \alpha - \beta, \qquad \lambda_3 = \alpha + \beta,$$
donc il faut que $\alpha > |\beta|$.

On note que dans ce cas, lorsque $\mathbb{A}$ est symétrique définie positive alors elle est aussi à dominance diagonale stricte.

Exercice 5.8

Écrire les formules de la méthode d'élimination de GAUSS pour une matrice de la forme

$$\mathbb{A} = \begin{pmatrix} a_{1,1} & a_{1,2} & 0 & \dots & & 0 \\ a_{2,1} & a_{2,2} & a_{2,3} & 0 & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & \ddots & 0 \\ \vdots & & & & a_{n-1,n-1} & a_{n-1,n} \\ a_{n,1} & a_{n,2} & \dots & & a_{n,n-1} & a_{n,n} \end{pmatrix}.$$

Quelle est la forme finale de la matrice $\mathbb{U} = \mathbb{A}^{(n)}$? Étant donné la forme particulière de la matrice $\mathbb{A}$, indiquer le nombre minimal d'opérations nécessaire pour calculer $\mathbb{U}$ ainsi que celui pour la résolution des systèmes triangulaires finaux.

CORRECTION DE L'EXERCICE 5.8. Comme la matrice a une seule sur-diagonale non nulle, les formules de la méthode d'élimination de GAUSS deviennent

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} + \ell_{ik} a_{k,j}^{(k)}, \qquad i, j = k+1,$$
$$\ell_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}, \qquad i = k+1.$$

La coût est donc de l'ordre de n et la matrice $\mathbb{U}$ est bidiagonale supérieure.

Exercice 5.9

Soit $\alpha \in \mathbb{R}^*$ et considérons les matrices carrées de dimension n

$$\mathbb{A} = \begin{pmatrix} \alpha & 0 & \dots & -\alpha \\ 0 & \ddots & & \vdots \\ \vdots & & \alpha & -\alpha \\ -\alpha & \dots & -\alpha & -\alpha \end{pmatrix}, \qquad \mathbb{B} = \begin{pmatrix} \frac{\beta}{\alpha} & -\frac{\gamma}{\alpha} & \dots & -\frac{\gamma}{\alpha} \\ -\frac{\gamma}{\alpha} & \ddots & & \vdots \\ \vdots & & \frac{\beta}{\alpha} & -\frac{\gamma}{\alpha} \\ -\frac{\gamma}{\alpha} & \dots & -\frac{\gamma}{\alpha} & \frac{\gamma}{\alpha} \end{pmatrix}.$$

1. Calculer γ et β pour que $\mathbb{B}$ soit l'inverse de $\mathbb{A}$.
2. Calculer le conditionnement $K_\infty(\mathbb{A})$ en fonction de n et en calculer la limite pour n qui tend vers l'infini.

Correction de l'exercice 5.9.

1. Par définition, $\mathbb{B}$ est la matrice inverse de $\mathbb{A}$ si $\mathbb{AB} = \mathbb{BA} = \mathbb{I}$. Comme

$$\mathbb{AB} = \begin{pmatrix} \beta+\gamma & 0 & \dots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & \beta+\gamma & 0 \\ -\beta+(n-3)\gamma & \dots & -\beta+(n-3)\gamma & (n-2)\gamma \end{pmatrix},$$

il faut que

$$\begin{cases} \beta+\gamma = 1 \\ -\beta+(n-3)\gamma = 0 \\ (n-2)\gamma = 1 \end{cases}$$

ce qui donne

$$\beta = \frac{n-3}{n-2}, \qquad \gamma = \frac{1}{n-2}.$$

2. On trouve immédiatement $\|\mathbb{A}\|_\infty = n|\alpha|$ tandis que

$$\|\mathbb{A}^{-1}\|_\infty = \frac{1}{|\alpha|} \max\left\{ n, \frac{n}{n-2} \right\} = \frac{2}{|\alpha|}.$$

On conclut que le conditionnement $K_\infty(\mathbb{A})$ en fonction de n est

$$K_\infty(\mathbb{A}) = n|\alpha| \frac{2}{|\alpha|} = 2n.$$

La matrice est donc mal conditionnée pour n grand.

Exercice 5.10

On suppose que le nombre réel $\varepsilon > 0$ est assez petit pour que l'ordinateur arrondisse $1+\varepsilon$ en 1 et $1+(1/\varepsilon)$ en $1/\varepsilon$ (ε est plus petit que l'erreur machine (relative), par exemple, $\varepsilon = 2^{-30}$ en format 32 *bits*). Simuler la résolution par l'ordinateur des deux systèmes suivants :

$$\begin{cases} \varepsilon a + b = 1 \\ 2a + b = 0 \end{cases} \quad \text{et} \quad \begin{cases} 2a + b = 0 \\ \varepsilon a + b = 1 \end{cases}$$

On appliquera pour cela la méthode du pivot de Gauss et on donnera les décompositions $\mathbb{LU}$ des deux matrices associées à ces systèmes. On fournira également la solution exacte de ces systèmes. Commenter.

Correction de l'exercice 5.10.

Premier système :

$$\begin{pmatrix} \varepsilon & 1 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Factorisation $\mathbb{LU}$:

$$\begin{pmatrix} \varepsilon & 1 \\ 2 & 1 \end{pmatrix} \xrightarrow{L_2 \leftarrow L_2 - \frac{2}{\varepsilon} L_1} \begin{pmatrix} \varepsilon & 1 \\ 0 & 1-\frac{2}{\varepsilon} \end{pmatrix} \qquad \text{donc} \qquad \mathbb{L} = \begin{pmatrix} 1 & 0 \\ \frac{2}{\varepsilon} & 1 \end{pmatrix}, \qquad \mathbb{U} = \begin{pmatrix} \varepsilon & 1 \\ 0 & 1-\frac{2}{\varepsilon} \end{pmatrix}$$

Pour résoudre le système linéaire on résout les systèmes triangulaires $\mathbb{L}\mathbf{y}=\mathbf{b}$ et $\mathbb{U}\mathbf{x}=\mathbf{y}$:

$$\begin{pmatrix}1 & 0\\ \frac{2}{\varepsilon} & 1\end{pmatrix}\begin{pmatrix}y_1\\ y_2\end{pmatrix}=\begin{pmatrix}1\\ 0\end{pmatrix} \quad\Longrightarrow\quad y_1=1,\quad y_2=-\frac{2}{\varepsilon};$$

$$\begin{pmatrix}\varepsilon & 1\\ 0 & 1-\frac{2}{\varepsilon}\end{pmatrix}\begin{pmatrix}a\\ b\end{pmatrix}=\begin{pmatrix}1\\ -\frac{2}{\varepsilon}\end{pmatrix} \quad\Longrightarrow\quad b=-\frac{2}{\varepsilon\left(1-\frac{2}{\varepsilon}\right)},\quad a=\frac{1+\frac{2}{\varepsilon\left(1-\frac{2}{\varepsilon}\right)}}{\varepsilon}.$$

Mais avec l'ordinateur, comme $1+\varepsilon\approx 1$ et $1+(1/\varepsilon)\approx 1/\varepsilon$, on obtient

$$\widetilde{\mathbb{L}}=\begin{pmatrix}1 & 0\\ \frac{2}{\varepsilon} & 1\end{pmatrix} \qquad \widetilde{\mathbb{U}}=\begin{pmatrix}\varepsilon & 1\\ 0 & -\frac{2}{\varepsilon}\end{pmatrix}$$

Pour résoudre ce système linéaire approché on résout les systèmes triangulaires $\widetilde{\mathbb{L}}\mathbf{y}=\mathbf{b}$ et $\widetilde{\mathbb{U}}\mathbf{x}=\mathbf{y}$:

$$\begin{pmatrix}1 & 0\\ \frac{2}{\varepsilon} & 1\end{pmatrix}\begin{pmatrix}y_1\\ y_2\end{pmatrix}=\begin{pmatrix}1\\ 0\end{pmatrix} \quad\Longrightarrow\quad y_1=1,\quad y_2=-\frac{2}{\varepsilon};$$

$$\begin{pmatrix}\varepsilon & 1\\ 0 & -\frac{2}{\varepsilon}\end{pmatrix}\begin{pmatrix}a\\ b\end{pmatrix}=\begin{pmatrix}1\\ -\frac{2}{\varepsilon}\end{pmatrix} \quad\Longrightarrow\quad b=1,\quad a=0.$$

Second système :

$$\begin{pmatrix}2 & 1\\ \varepsilon & 1\end{pmatrix}\begin{pmatrix}a\\ b\end{pmatrix}=\begin{pmatrix}0\\ 1\end{pmatrix}.$$

Factorisation $\mathbb{L}\mathbb{U}$:

$$\begin{pmatrix}2 & 1\\ \varepsilon & 1\end{pmatrix}\xrightarrow{L_2\leftarrow L_2-\frac{\varepsilon}{2}L_1}\begin{pmatrix}2 & 1\\ 0 & 1-\frac{\varepsilon}{2}\end{pmatrix} \quad\text{donc}\quad \mathbb{L}=\begin{pmatrix}1 & 0\\ \frac{\varepsilon}{2} & 1\end{pmatrix},\quad \mathbb{U}=\begin{pmatrix}2 & 1\\ 0 & 1-\frac{2}{\varepsilon}\end{pmatrix}$$

Pour résoudre le système linéaire on résout les systèmes triangulaires $\mathbb{L}\mathbf{y}=\mathbf{b}$ et $\mathbb{U}\mathbf{x}=\mathbf{y}$:

$$\begin{pmatrix}1 & 0\\ \frac{\varepsilon}{2} & 1\end{pmatrix}\begin{pmatrix}y_1\\ y_2\end{pmatrix}=\begin{pmatrix}0\\ 1\end{pmatrix} \quad\Longrightarrow\quad y_1=0,\quad y_2=1;$$

$$\begin{pmatrix}2 & 1\\ 0 & 1-\frac{2}{\varepsilon}\end{pmatrix}\begin{pmatrix}a\\ b\end{pmatrix}=\begin{pmatrix}0\\ 1\end{pmatrix} \quad\Longrightarrow\quad b=-\frac{2}{\varepsilon\left(1-\frac{2}{\varepsilon}\right)},\quad a=\frac{1+\frac{2}{\varepsilon\left(1-\frac{2}{\varepsilon}\right)}}{\varepsilon}.$$

Mais avec l'ordinateur, comme $1+\varepsilon\approx 1$ et $1+(1/\varepsilon)\approx 1/\varepsilon$, on obtient

$$\widetilde{\mathbb{L}}=\begin{pmatrix}1 & 0\\ \frac{\varepsilon}{2} & 1\end{pmatrix} \qquad \widetilde{\mathbb{U}}=\begin{pmatrix}2 & 1\\ 0 & -\frac{2}{\varepsilon}\end{pmatrix}$$

Pour résoudre ce système linéaire approché on résout les systèmes triangulaires $\widetilde{\mathbb{L}}\mathbf{y}=\mathbf{b}$ et $\widetilde{\mathbb{U}}\mathbf{x}=\mathbf{y}$:

$$\begin{pmatrix}1 & 0\\ \frac{\varepsilon}{2} & 1\end{pmatrix}\begin{pmatrix}y_1\\ y_2\end{pmatrix}=\begin{pmatrix}0\\ 1\end{pmatrix} \quad\Longrightarrow\quad y_1=0,\quad y_2=1;$$

$$\begin{pmatrix}2 & 1\\ 0 & -\frac{2}{\varepsilon}\end{pmatrix}\begin{pmatrix}a\\ b\end{pmatrix}=\begin{pmatrix}0\\ 1\end{pmatrix} \quad\Longrightarrow\quad b=-\frac{\varepsilon}{2},\quad a=\frac{\varepsilon}{4}.$$

Exercice 5.11

Rappeler l'algorithme vu en cours pour calculer la décomposition $\mathbb{L}\mathbb{U}$ d'une matrice $\mathbb{A}$ et la solution du système $\mathbb{A}\mathbf{x}=\mathbf{b}$ où le vecteur colonne $\mathbf{b}$ est donné. On appliquera ces algorithmes pour les cas suivants :

$$\begin{pmatrix}1 & 1 & 1\\ 2 & 1 & 3\\ -3 & 2 & 4\end{pmatrix}\begin{pmatrix}x_1\\ x_2\\ x_3\end{pmatrix}=\begin{pmatrix}1\\ 1\\ 1\end{pmatrix} \quad\text{et}\quad \begin{pmatrix}1 & 2 & 3 & 4\\ 2 & -5 & 7 & 1\\ 3 & 1 & 1 & 5\\ 2 & 2 & 0 & 3\end{pmatrix}\begin{pmatrix}x_1\\ x_2\\ x_3\\ x_4\end{pmatrix}=\begin{pmatrix}1\\ 1\\ 1\\ 1\end{pmatrix} \quad\text{et}\quad \begin{pmatrix}1 & 1 & 1 & 1\\ 1 & -2 & 3 & 4\\ 1 & 4 & 6 & 8\\ 1 & 0 & 0 & 0\end{pmatrix}\begin{pmatrix}x_1\\ x_2\\ x_3\\ x_4\end{pmatrix}=\begin{pmatrix}1\\ 1\\ 1\\ 1\end{pmatrix}$$

CORRECTION DE L'EXERCICE 5.11.

Premier système :

$$\left(\begin{array}{ccc|c} 1 & 1 & 1 & 1 \\ 2 & 1 & 3 & 1 \\ -3 & 2 & 4 & 1 \end{array}\right) \xrightarrow[L_3 \leftarrow L_3 - \frac{-3}{1} L_1]{L_2 \leftarrow L_2 - \frac{2}{1} L_1} \left(\begin{array}{ccc|c} 1 & 1 & 1 & 1 \\ 0 & -1 & 1 & -1 \\ 0 & 5 & 7 & 4 \end{array}\right) \xrightarrow{L_3 \leftarrow L_3 - \frac{5}{-1} L_2} \left(\begin{array}{ccc|c} 1 & 1 & 1 & 1 \\ 0 & -1 & 1 & -1 \\ 0 & 0 & 12 & -1 \end{array}\right)$$

donc

$$\mathbb{L} = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -3 & -5 & 1 \end{pmatrix} \qquad \mathbb{U} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & -1 & 1 \\ 0 & 0 & 12 \end{pmatrix}$$

Il ne reste à résoudre que le système triangulaire

$$\begin{cases} x_1 + x_2 + x_3 = 1 \\ -x_2 + x_3 = -1 \\ 12 x_3 = -1 \end{cases} \implies x_3 = -\frac{1}{12}, \quad x_2 = \frac{11}{12}, \quad x_1 = \frac{1}{6}.$$

Deuxième système :

$$\left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 1 \\ 2 & -5 & 7 & 1 & 1 \\ 3 & 1 & 1 & 5 & 1 \\ 2 & 2 & 0 & 3 & 1 \end{array}\right) \xrightarrow[L_4 \leftarrow L_4 - \frac{2}{1} L_1]{\substack{L_2 \leftarrow L_2 - \frac{2}{1} L_1 \\ L_3 \leftarrow L_3 - \frac{3}{1} L_1}} \left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 1 \\ 0 & -9 & 1 & -7 & -1 \\ 0 & -5 & -8 & -7 & -2 \\ 0 & -2 & -6 & -5 & -1 \end{array}\right)$$

$$\xrightarrow[L_4 \leftarrow L_4 - \frac{-2}{-9} L_2]{L_3 \leftarrow L_3 - \frac{-5}{-9} L_2} \left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 1 \\ 0 & -9 & 1 & -7 & -1 \\ 0 & 0 & -\frac{77}{9} & -\frac{28}{9} & -\frac{13}{9} \\ 0 & 0 & -\frac{56}{9} & -\frac{31}{9} & -\frac{7}{9} \end{array}\right) \xrightarrow{L_4 \leftarrow L_4 - \frac{56/9}{77/9} L_2} \left(\begin{array}{cccc|c} 1 & 2 & 3 & 4 & 1 \\ 0 & -9 & 1 & -7 & -1 \\ 0 & 0 & -\frac{77}{9} & -\frac{28}{9} & -\frac{13}{9} \\ 0 & 0 & 0 & -\frac{13}{11} & \frac{3}{11} \end{array}\right)$$

donc

$$\mathbb{L} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & \frac{5}{9} & 1 & 0 \\ 2 & \frac{2}{9} & \frac{56}{77} & 1 \end{pmatrix} \qquad \mathbb{U} = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & -9 & 1 & -7 \\ 0 & 0 & -\frac{77}{9} & -\frac{28}{9} \\ 0 & 0 & 0 & -\frac{13}{11} \end{pmatrix}$$

Il ne reste à résoudre que le système triangulaire

$$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 = 1 \\ -9x_2 + x_3 - 7x_4 = -1 \\ -\frac{77}{9} x_3 - \frac{28}{9} x_4 = -\frac{13}{9} \\ -\frac{13}{11} x_4 = \frac{3}{11} \end{cases} \implies x_4 = -\frac{3}{13}, \quad x_3 = \frac{23}{91}, \quad x_2 = \frac{29}{91}, \quad x_1 = \frac{48}{91}.$$

Troisième système :

$$\left(\begin{array}{cccc|c} 1 & 1 & 1 & 1 & 1 \\ 1 & -2 & 3 & 4 & 1 \\ 1 & 4 & 6 & 8 & 1 \\ 1 & 0 & 0 & 0 & 1 \end{array}\right) \xrightarrow[L_4 \leftarrow L_4 - L_1]{\substack{L_2 \leftarrow L_2 - L_1 \\ L_3 \leftarrow L_3 - L_1}} \left(\begin{array}{cccc|c} 1 & 1 & 1 & 1 & 1 \\ 0 & -3 & 2 & 3 & -0 \\ 0 & 3 & 5 & 7 & 0 \\ 0 & -1 & -1 & -1 & 0 \end{array}\right)$$

$$\xrightarrow[L_4 \leftarrow L_4 - \frac{-1}{-3} L_2]{L_3 \leftarrow L_3 - (-1) L_2} \left(\begin{array}{cccc|c} 1 & 1 & 1 & 1 & 1 \\ 0 & -3 & 2 & 3 & -0 \\ 0 & 0 & 7 & 10 & 0 \\ 0 & 0 & -\frac{5}{3} & -2 & 0 \end{array}\right) \xrightarrow{L_4 \leftarrow L_4 - \frac{-5/3}{7} L_2} \left(\begin{array}{cccc|c} 1 & 1 & 1 & 1 & 1 \\ 0 & -3 & 2 & 3 & -0 \\ 0 & 0 & 7 & 10 & 0 \\ 0 & 0 & 0 & \frac{8}{21} & 0 \end{array}\right)$$

donc

$$\mathbb{L} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & -1 & 1 & 0 \\ 1 & \frac{1}{3} & \frac{-5}{21} & 1 \end{pmatrix} \qquad \mathbb{U} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & -3 & 2 & 3 \\ 0 & 0 & 7 & 10 \\ 0 & 0 & 0 & \frac{8}{21} \end{pmatrix}$$

Il ne reste à résoudre que le système triangulaire

$$\begin{cases} x_1 + x_2 + x_3 + x_4 = 1 \\ -3x_2 + 2x_3 + 3x_4 = 0 \\ 7x_3 + 10x_4 = 0 \\ \frac{8}{21} x_4 = 0 \end{cases} \implies x_4 = 0, \quad x_3 = 0, \quad x_2 = 0, \quad x_1 = 1.$$

Exercice 5.12

Écrire les méthodes itératives de GAUSS, JACOBI et GAUSS-SEIDEL pour les systèmes suivants :

$$\begin{cases} 10a+b=11 \\ 2a+10b=12 \end{cases} \quad \text{et} \quad \begin{cases} 2a+10b=12 \\ 10a+b=11. \end{cases}$$

Pour chacun de ces méthodes et systèmes, on illustrera les résultats théoriques de convergence/non-convergence en calculant les 3 premières itérés en prenant comme point de départ le vecteur $(a,b)=(0,0)$.

CORRECTION DE L'EXERCICE 5.12.

Gauss ⋆ Premier système :

$$\left(\begin{array}{cc|c} 10 & 1 & 11 \\ 2 & 10 & 12 \end{array}\right) \xrightarrow{L_2 \leftarrow L_2 - \frac{2}{10}L_1} \left(\begin{array}{cc|c} 10 & 1 & 11 \\ 0 & \frac{49}{5} & \frac{49}{5} \end{array}\right) \Longrightarrow \begin{cases} 10a+b=11 \\ \frac{49}{5}b=\frac{49}{5} \end{cases} \Longrightarrow \begin{cases} a=1 \\ b=1. \end{cases}$$

⋆ Second système :

$$\left(\begin{array}{cc|c} 2 & 10 & 12 \\ 10 & 1 & 11 \end{array}\right) \xrightarrow{L_2 \leftarrow L_2 - \frac{10}{2}L_1} \left(\begin{array}{cc|c} 2 & 10 & 12 \\ 0 & -49 & -49 \end{array}\right) \Longrightarrow \begin{cases} 2a+10b=12 \\ -49b=-49 \end{cases} \Longrightarrow \begin{cases} a=1 \\ b=1. \end{cases}$$

Jacobi ⋆ Premier système :

$$\begin{cases} 10a+b=11 \\ 2a+10b=12 \end{cases} \Longleftrightarrow \begin{cases} a=\frac{11-b}{10} \\ b=\frac{12-2a}{10} \end{cases}$$

La matrice étant à diagonale dominante stricte, la méthode converge et on a

$$\mathbf{x}^{(0)}=\begin{pmatrix}0\\0\end{pmatrix}, \quad \mathbf{x}^{(1)}=\begin{pmatrix}\frac{11-0}{10}\\ \frac{12-0}{10}\end{pmatrix}=\begin{pmatrix}{}^{11}/_{10}\\ {}^{12}/_{10}\end{pmatrix}, \quad \mathbf{x}^{(2)}=\begin{pmatrix}\frac{11-\frac{12}{10}}{10}\\ \frac{12-2\frac{11}{10}}{10}\end{pmatrix}=\begin{pmatrix}{}^{49}/_{50}\\ {}^{49}/_{50}\end{pmatrix}, \quad \mathbf{x}^{(3)}=\begin{pmatrix}\frac{11-\frac{49}{50}}{10}\\ \frac{12-2\frac{49}{50}}{10}\end{pmatrix}=\begin{pmatrix}{}^{501}/_{500}\\ {}^{502}/_{500}\end{pmatrix}.$$

⋆ Second système :

$$\begin{cases} 2a+10b=12 \\ 10a+b=11 \end{cases} \Longleftrightarrow \begin{cases} a=\frac{12-10b}{2} \\ b=11-10a \end{cases}$$

La méthode ne converge pas, en effet on a

$$\mathbf{x}^{(0)}=\begin{pmatrix}0\\0\end{pmatrix}, \quad \mathbf{x}^{(1)}=\begin{pmatrix}\frac{12-0}{2}\\ 11-0\end{pmatrix}=\begin{pmatrix}6\\11\end{pmatrix}, \quad \mathbf{x}^{(2)}=\begin{pmatrix}\frac{12-10\times 11}{2}\\ 11-10\times 6\end{pmatrix}=\begin{pmatrix}-49\\-49\end{pmatrix}, \quad \mathbf{x}^{(3)}=\begin{pmatrix}\frac{12-10\times(-49)}{2}\\ 11-10\times(-49)\end{pmatrix}=\begin{pmatrix}251\\501\end{pmatrix}.$$

Gauss-Seidel ⋆ Premier système :

$$\begin{cases} 10a+b=11 \\ 2a+10b=12 \end{cases} \Longleftrightarrow \begin{cases} a=\frac{11-b}{10} \\ b=\frac{12-2a}{10} \end{cases}$$

La matrice étant à diagonale dominante stricte, la méthode converge et on a

$$\mathbf{x}^{(0)}=\begin{pmatrix}0\\0\end{pmatrix}, \quad \mathbf{x}^{(1)}=\begin{pmatrix}\frac{11-0}{10}\\ \frac{12-2\frac{11}{10}}{10}\end{pmatrix}=\begin{pmatrix}{}^{11}/_{10}\\ {}^{49}/_{50}\end{pmatrix}, \quad \mathbf{x}^{(2)}=\begin{pmatrix}\frac{11-\frac{49}{50}}{10}\\ \frac{12-2\frac{501}{500}}{10}\end{pmatrix}=\begin{pmatrix}{}^{501}/_{500}\\ {}^{2499}/_{2500}\end{pmatrix}, \quad \mathbf{x}^{(3)}=\begin{pmatrix}\frac{11-\frac{2499}{2500}}{10}\\ \frac{12-2\frac{25001}{25000}}{10}\end{pmatrix}=\begin{pmatrix}{}^{25001}/_{25000}\\ {}^{12499}/_{125000}\end{pmatrix}.$$

⋆ Second système :

$$\begin{cases} 2a+10b=12 \\ 10a+b=11 \end{cases} \Longleftrightarrow \begin{cases} a=\frac{12-10b}{2} \\ b=11-10a \end{cases}$$

La méthode ne converge pas, en effet on a

$$\mathbf{x}^{(0)}=\begin{pmatrix}0\\0\end{pmatrix}, \quad \mathbf{x}^{(1)}=\begin{pmatrix}\frac{12-0}{2}\\ 11-10\times 6\end{pmatrix}=\begin{pmatrix}6\\-49\end{pmatrix}, \quad \mathbf{x}^{(2)}=\begin{pmatrix}\frac{12-10\times(-49)}{2}\\ 11-10\times 251\end{pmatrix}=\begin{pmatrix}251\\-2499\end{pmatrix}, \quad \mathbf{x}^{(3)}=\begin{pmatrix}\frac{12-10\times(-2499)}{2}\\ 11-10\times(12501)\end{pmatrix}=\begin{pmatrix}12501\\-124999\end{pmatrix}.$$

Exercice 5.13

Résoudre les systèmes linéaires suivants :

$$\begin{cases} x-5y-7z=3 \\ 2x-13y-18z=3 \\ 3x-27y-36z=3 \end{cases} \quad \text{et} \quad \begin{cases} x-5y-7z=6 \\ 2x-13y-18z=0 \\ 3x-27y-36z=-3 \end{cases} \quad \text{et} \quad \begin{cases} x-5y-7z=0 \\ 2x-13y-18z=3 \\ 3x-27y-36z=6. \end{cases}$$

CORRECTION DE L'EXERCICE 5.13. Le trois systèmes s'écrivent sous forme matricielle

$$\begin{pmatrix} 1 & -5 & -7 \\ 2 & -13 & -18 \\ 3 & -27 & -36 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 3 \\ 3 \\ 3 \end{pmatrix} \quad \text{et} \quad \begin{pmatrix} 1 & -5 & -7 \\ 2 & -13 & -18 \\ 3 & -27 & -36 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 6 \\ 0 \\ -3 \end{pmatrix} \quad \text{et} \quad \begin{pmatrix} 1 & -5 & -7 \\ 2 & -13 & -18 \\ 3 & -27 & -36 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 0 \\ 3 \\ 6 \end{pmatrix}$$

On remarque que seul le terme source change. On calcul d'abord la décomposition $\mathbb{L}\mathbb{U}$ de la matrice $\mathbb{A}$:

$$\begin{pmatrix} 1 & -5 & -7 \\ 2 & -13 & -18 \\ 3 & -27 & -36 \end{pmatrix} \xrightarrow[L_3 \leftarrow L_3 - 3L_1]{L_2 \leftarrow L_2 - 2L_1} \begin{pmatrix} 1 & -5 & -7 \\ 0 & -3 & -4 \\ 0 & -12 & -15 \end{pmatrix} \xrightarrow{L_3 \leftarrow L_3 - 4L_2} \begin{pmatrix} 1 & -5 & -7 \\ 0 & -3 & -4 \\ 0 & 0 & 1 \end{pmatrix}$$

donc

$$\mathbb{L} = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 4 & 1 \end{pmatrix} \qquad \mathbb{U} = \begin{pmatrix} 1 & -5 & -7 \\ 0 & -3 & -4 \\ 0 & 0 & 1 \end{pmatrix}$$

Pour résoudre chaque système linéaire on résout les systèmes triangulaires $\mathbb{L}\mathbf{y} = \mathbf{b}$ et $\mathbb{U}\mathbf{x} = \mathbf{y}$.

1. Pour le premier système on a

$$\begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 4 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 3 \\ 3 \\ 3 \end{pmatrix} \implies y_1 = 3, \quad y_2 = -3, \quad y_3 = 6;$$

$$\begin{pmatrix} 1 & -5 & -7 \\ 0 & -3 & -4 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 3 \\ -3 \\ 6 \end{pmatrix} \implies x_3 = 6, \quad x_2 = -7, \quad x_1 = 10.$$

2. Pour le seconde système on a

$$\begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 4 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 6 \\ 0 \\ -3 \end{pmatrix} \implies y_1 = 6, \quad y_2 = -12, \quad y_3 = 27;$$

$$\begin{pmatrix} 1 & -5 & -7 \\ 0 & -3 & -4 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 6 \\ -12 \\ 27 \end{pmatrix} \implies x_3 = 27, \quad x_2 = -32, \quad x_1 = 35.$$

3. Pour le dernier système on a

$$\begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 4 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 3 \\ 6 \end{pmatrix} \implies y_1 = 0, \quad y_2 = 3, \quad y_3 = -6;$$

$$\begin{pmatrix} 1 & -5 & -7 \\ 0 & -3 & -4 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 6 \\ -12 \\ 27 \end{pmatrix} \implies x_3 = -6, \quad x_2 = 7, \quad x_1 = -7.$$

Exercice 5.14

Soit $\mathbb{A}$ une matrice, $\mathbb{A} \in \mathscr{M}_{n,n}(\mathbb{R})$.

1. Rappeler la méthode de JACOBI pour la résolution du système $\mathbb{A}\mathbf{x} = \mathbf{b}$, avec $\mathbf{b} \in \mathscr{M}_{n,1}(\mathbb{R})$ donné.

2. Soit la matrice $\mathbb{A}$ suivante :
$$\begin{pmatrix} 4 & -1 & -1 \\ -1 & 3 & -1 \\ -1 & -1 & 4 \end{pmatrix}.$$
La méthode de JACOBI est-elle convergente pour cette matrice ?
3. Construire à la main les matrices $\mathbb{L}$ et $\mathbb{U}$ de la factorisation $\mathbb{LU}$ pour la matrice ci-dessus.

CORRECTION DE L'EXERCICE 5.14.

1. La méthode de JACOBI est une méthode itérative pour le calcul de la solution d'un système linéaire qui construit une suite de vecteurs $\mathbf{x}^{(k)} \in \mathbb{R}^n$ convergent vers la solution exacte $\mathbf{x}$ pour tout vecteur initiale $\mathbf{x}^{(0)} \in \mathbb{R}^n$:
$$x_i^{k+1} = \frac{b_i - \sum\limits_{\substack{j=1 \\ j\neq i}}^{n} a_{ij} x_j^k}{a_{ii}}, \qquad i = 1, \dots, n.$$
2. Comme $|4| > |-1| + |-1|$, $|3| > |-1| + |-1|$ et $|4| > |-1| + |-1|$, la matrice $\mathbb{A}$ est à diagonale dominante stricte donc la méthode de JACOBI converge
3. Factorisation :
$$\begin{pmatrix} 4 & -1 & -1 \\ -1 & 3 & -1 \\ -1 & -1 & 4 \end{pmatrix} \xrightarrow[L_3 \leftarrow L_3 - \frac{-1}{4} L_1]{L_2 \leftarrow L_2 - \frac{-1}{4} L_1} \begin{pmatrix} 4 & -1 & -1 \\ 0 & {}^{11}/_{4} & -{}^{5}/_{4} \\ 0 & -{}^{5}/_{4} & {}^{15}/_{4} \end{pmatrix} \xrightarrow{L_3 \leftarrow L_3 - \frac{-5/4}{11/4} L_2} \begin{pmatrix} 4 & -1 & -1 \\ 0 & {}^{11}/_{4} & -{}^{5}/_{4} \\ 0 & 0 & {}^{35}/_{11} \end{pmatrix}.$$
Par conséquent
$$\mathbb{L} = \begin{pmatrix} 1 & 0 & 0 \\ -{}^{1}/_{4} & 1 & 0 \\ -{}^{1}/_{4} & -{}^{5}/_{11} & 1 \end{pmatrix} \quad \text{et} \quad \mathbb{U} = \begin{pmatrix} 4 & -1 & -1 \\ 0 & {}^{11}/_{4} & -{}^{5}/_{4} \\ 0 & 0 & {}^{35}/_{11} \end{pmatrix}.$$

Exercice 5.15

Soit la matrice $\mathbb{A} \in \mathbb{R}^{n\times n}$, $n \geq 3$, dont les éléments vérifient
- $a_{ij} = 1$ si $i = j$ ou $i = n$,
- $a_{ij} = -1$ si $i < j$,
- $a_{ij} = 0$ sinon.

Calculer la factorisation $\mathbb{LU}$ de $\mathbb{A}$.

CORRECTION DE L'EXERCICE 5.15. Factorisation $\mathbb{LU}$ de la matrice $\mathbb{A}$:
$$\begin{pmatrix} 1 & -1 & \dots & \dots & \dots & -1 \\ 0 & 1 & \ddots & & & \vdots \\ \vdots & \ddots & 1 & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & 0 & 1 & -1 \\ 1 & 1 & 1 & \dots & 1 & 1 \end{pmatrix} \xrightarrow{L_n \leftarrow L_n - \frac{1}{1} L_1} \begin{pmatrix} 1 & -1 & \dots & \dots & \dots & -1 \\ 0 & 1 & \ddots & & & \vdots \\ \vdots & \ddots & 1 & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & 0 & 1 & -1 \\ 0 & 2 & 2 & \dots & 2 & 2 \end{pmatrix} \xrightarrow{L_n \leftarrow L_n - \frac{2}{1} L_2} \begin{pmatrix} 1 & -1 & \dots & \dots & \dots & -1 \\ 0 & 1 & \ddots & & & \vdots \\ \vdots & \ddots & 1 & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & 0 & 1 & -1 \\ 0 & 0 & 4 & \dots & 4 & 4 \end{pmatrix}$$
$$[\dots] \xrightarrow{L_n \leftarrow L_n - \frac{2^{n-2}}{1} L_{n-1}} \begin{pmatrix} 1 & -1 & \dots & \dots & \dots & -1 \\ 0 & 1 & \ddots & & & \vdots \\ \vdots & \ddots & 1 & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & 0 & 1 & -1 \\ 0 & 0 & 0 & \dots & 0 & 2^{n-1} \end{pmatrix}.$$
On obtient les matrices
$$\mathbb{L} = \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & 0 \\ 0 & 1 & \ddots & & & \vdots \\ 0 & 0 & 1 & \ddots & & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 1 & 0 \\ 1 & 2 & 4 & \dots & 2^{n-2} & 1 \end{pmatrix} \quad \text{et} \quad \mathbb{U} = \begin{pmatrix} 1 & -1 & \dots & \dots & \dots & -1 \\ 0 & 1 & \ddots & & & \vdots \\ 0 & 0 & 1 & \ddots & & \vdots \\ \vdots & \vdots & \ddots & \ddots & -1 & \vdots \\ 0 & 0 & \dots & 0 & 1 & -1 \\ 0 & 0 & 0 & \dots & 0 & 2^{n-1} \end{pmatrix}.$$

c'est-à-dire
- ⋆ $\ell_{ii} = 1$ pour $i = 1,\dots,n$,
- ⋆ $\ell_{ij} = 0$ si $i < n$ et $i \neq j$,
- ⋆ $\ell_{nj} = 2^{j-1}$ si $j < n$;
- ⋆ $u_{ij} = a_{ij}$ pour i=1,. . .,n-1, j=1,. . .,n,
- ⋆ $u_{nj} = 0$ si $j < n$,
- ⋆ $u_{nn} = 2^{n-1}$.

Exercice 5.16

Considérons une matrice $\mathbb{A} \in \mathbb{R}^{n\times n}$ (avec $n \geq 3$) dont les éléments vérifient
- ⋆ $a_{ij} = 1$ si $i = j$ ou $j = n$,
- ⋆ $a_{ij} = -1$ si $i > j$,
- ⋆ $a_{ij} = 0$ sinon.

Calculer la factorisation $\mathbb{LU}$ de $\mathbb{A}$.

CORRECTION DE L'EXERCICE 5.16. Factorisation $\mathbb{LU}$ de la matrice $\mathbb{A}$:

$$\begin{pmatrix} 1 & 0 & \dots & \dots & 0 & 1 \\ -1 & 1 & \ddots & & \vdots & \vdots \\ \vdots & \ddots & 1 & \ddots & \vdots & \vdots \\ \vdots & & \ddots & \ddots & 0 & \vdots \\ \vdots & & & \ddots & 1 & 1 \\ -1 & \dots & \dots & \dots & -1 & 1 \end{pmatrix} \xrightarrow[L_n \leftarrow L_n + L_1]{L_2 \leftarrow L_2 + L_1} \begin{pmatrix} 1 & 0 & \dots & \dots & 0 & 1 \\ 0 & 1 & \ddots & & \vdots & 2 \\ \vdots & -1 & 1 & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 & \vdots \\ \vdots & \vdots & & \ddots & 1 & 2 \\ 0 & -1 & \dots & \dots & -1 & 2 \end{pmatrix} \xrightarrow[L_n \leftarrow L_n + L_2]{L_3 \leftarrow L_3 + L_2} \begin{pmatrix} 1 & 0 & \dots & \dots & 0 & 1 \\ 0 & 1 & \ddots & & \vdots & 2 \\ \vdots & 0 & 1 & \ddots & \vdots & 4 \\ \vdots & \vdots & -1 & \ddots & 0 & \vdots \\ \vdots & \vdots & \vdots & \ddots & 1 & 4 \\ 0 & 0 & -1 & \dots & -1 & 4 \end{pmatrix}$$

$$[\dots] \xrightarrow{L_n \leftarrow L_n + L_{n-1}} \begin{pmatrix} 1 & 0 & \dots & \dots & 0 & 2^0 \\ 0 & 1 & \ddots & & \vdots & 2^1 \\ \vdots & \ddots & 1 & \ddots & \vdots & 2^2 \\ \vdots & & \ddots & \ddots & 0 & \vdots \\ \vdots & & & \ddots & 1 & 2^{n-2} \\ 0 & \dots & \dots & \dots & 0 & 2^{n-1} \end{pmatrix}$$

On obtient les matrices

$$\mathbb{L} = \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & 0 \\ -1 & 1 & \ddots & & & \vdots \\ \vdots & \ddots & 1 & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & 1 & 0 \\ -1 & \dots & \dots & \dots & -1 & 1 \end{pmatrix} \quad \text{et} \quad \mathbb{U} = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 & 2^0 \\ 0 & 1 & \ddots & & \vdots & 2^1 \\ \vdots & \ddots & 1 & \ddots & \vdots & 2^2 \\ \vdots & & \ddots & \ddots & 0 & \vdots \\ \vdots & & & \ddots & 1 & 2^{n-2} \\ 0 & \dots & \dots & \dots & 0 & 2^{n-1} \end{pmatrix}.$$

i.e.

- ⋆ $\ell_{ii} = 1$ pour $i = 1,\dots,n,$,
- ⋆ $\ell_{ij} = -1$ si $i > j$
- ⋆ $\ell_{ij} = 0$ sinon ;
- ⋆ $u_{ii} = 1$ pour $i = 1,\dots,n-1$,
- ⋆ $u_{in} = 2^{i-1}$ pour $i = 1,\dots,n$,
- ⋆ $u_{ij} = 0$ sinon.

Exercice 5.17

On considère la matrice tridiagonale inversible $\mathbb{A} \in \mathbb{R}^{n\times n}$

$$\mathbb{A} = \begin{pmatrix} a_1 & c_1 & 0 & \dots & \dots & 0 \\ b_2 & a_2 & c_2 & \ddots & & \vdots \\ 0 & b_3 & a_3 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & b_{n-1} & a_{n-1} & c_{n-1} \\ 0 & \dots & \dots & 0 & b_n & a_n \end{pmatrix}$$

1. Montrer que les matrices $\mathbb{L}$ et $\mathbb{U}$ de la factorisation $\mathbb{LU}$ de $\mathbb{A}$ sont bidiagonales, *i.e.* si $a_{ij}=0$ pour $|i-j|>1$ alors $\ell_{ij}=0$ pour $i>1+j$ (et pour $i<j$ car triangulaire inférieure) et $u_{ij}=0$ pour $i<j-1$ (et pour $i>j$ car triangulaire supérieure).
   Soit $\mathbb{A}^{(k)}$, $k=0,\dots,n-1$ la matrice obtenue à l'étape k de la méthode de GAUSS, avec $\mathbb{A}^{(0)}=\mathbb{A}$ et $\mathbb{A}^{(n-1)}=\mathbb{U}$. On montrera par récurrence sur k que $\mathbb{A}^{(k)}$ est tridiagonale pour tout $k=0,\dots,n-1$, *i.e.* $a_{ij}^{(k)}=0$ pour $|i-j|>1$.

   Initialisation : pour $k=0$, $\mathbb{A}^{(0)}=\mathbb{A}$ est une matrice tridiagonale.

   Hérédité : soit $\mathbb{A}^{(k)}$ une matrice tridiagonale (*i.e.* $a_{ij}^{(k)}=0$ pour $|i-j|>1$) et montrons que $\mathbb{A}^{(k+1)}$ l'est aussi.
   - ⋆ Si $i\leq k$, que valent-ils les coefficients $a_{ij}^{(k+1)}$?
   - ⋆ Si $i>k$ alors on va considérer séparément les cas suivants :
     - ⋆ si $j\leq k$, que valent-ils les coefficients $a_{ij}^{(k+1)}$?
     - ⋆ si $j>k$ et $j<i-1$, que valent-ils les coefficients $a_{ik}^{(k)}$ et $\ell_{ik}^{(k)}$? Que peut-on déduire sur les coefficients $a_{ij}^{(k+1)}$?
     - ⋆ si $j>k$ et $j>i+1$, que valent-ils les coefficients $a_{kj}^{(k)}$ et $\ell_{ik}^{(k)}$? Que peut-on déduire sur les coefficients $a_{ij}^{(k+1)}$?

   NB : Justifier succinctement chaque réponse !
2. On a montré au point précédent que les matrices $\mathbb{L}$ et $\mathbb{U}$ de la factorisation $\mathbb{LU}$ de $\mathbb{A}$ sont bidiagonales, écrivons-les sous la forme
$$\mathbb{L}=\begin{pmatrix} 1 & 0 & \dots & \dots & \dots & 0\\ \beta_2 & 1 & \ddots & & & \vdots\\ 0 & \beta_3 & 1 & \ddots & & \vdots\\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots\\ \vdots & & \ddots & \beta_{n-1} & 1 & 0\\ 0 & \dots & \dots & 0 & \beta_n & 1 \end{pmatrix},\qquad \mathbb{U}=\begin{pmatrix} \alpha_1 & \gamma_1 & 0 & \dots & \dots & 0\\ 0 & \alpha_2 & \gamma_2 & \ddots & & \vdots\\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots\\ \vdots & & \ddots & \ddots & \ddots & 0\\ \vdots & & & \ddots & \alpha_{n-1} & \gamma_{n-1}\\ 0 & \dots & \dots & \dots & 0 & \alpha_n \end{pmatrix}.$$
   Calculer $(\alpha_1,\alpha_2,\dots,\alpha_n)$, $(\beta_2,\beta_3,\dots,\beta_n)$ et $(\gamma_1,\gamma_2,\dots,\gamma_{n-1})$ en fonction de $(a_1,a_2,\dots,a_n)$, $(b_2,b_3,\dots,b_n)$ et $(c_1,c_2,\dots,c_{n-1})$. En déduire un algorithme de factorisation.
3. À l'aide des formules trouvées au point précédent, écrire l'algorithme pour résoudre le système linéaire $\mathbb{A}\mathbf{x}=\mathbf{f}$ où $\mathbf{f}=(f_1,f_2,\dots,f_n)^T\in\mathbb{R}^n$.

CORRECTION DE L'EXERCICE 5.17.

1. Soit $\mathbb{A}^{(k)}$, $k=0,\dots,n-1$ la matrice obtenue à l'étape k de la méthode de GAUSS, avec $\mathbb{A}^{(0)}=\mathbb{A}$ et $\mathbb{A}^{(n-1)}=\mathbb{U}$. On montrera par récurrence sur k que $\mathbb{A}^{(k)}$ est tridiagonale, *i.e.* $a_{ij}^{(k)}=0$ pour $|i-j|>1$.

   Initialisation : pour $k=0$, $\mathbb{A}^{(0)}=\mathbb{A}$ qui est une matrice tridiagonale.

   Hérédité : soit $\mathbb{A}^{(k)}$ une matrice tridiagonale (*i.e.* $a_{ij}^{(k)}=0$ pour $|i-j|>1$) et montrons que $\mathbb{A}^{(k+1)}$ l'est aussi.
   - ⋆ Si $i\leq k$ alors $a_{ij}^{(k+1)}=a_{ij}^{(k)}=0$ (les lignes $L_1,\dots,L_k$ de la matrice $\mathbb{A}^{(k)}$ ne sont pas modifiées à l'étape k).
   - ⋆ Soit $i>k$, alors les lignes $L_{k+1},\dots,L_n$ de la matrice $\mathbb{A}^{(k)}$ vont être modifiées selon la relation)
$$a_{ij}^{(k+1)}=a_{ij}^{(k)}-\frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}a_{kj}^{(k)}.$$
     Pour chaque ligne $i>k$, considérons séparément les colonnes $j\leq k$ et les colonnes $j>k$:
     - ⋆ si $j\leq k$, $a_{ij}^{(k+1)}=0$ (zéros qu'on fait apparaitre avec la méthode de GAUSS pour une matrice quelconque),
     - ⋆ soit $j>k$:
       - ⋆ si $j<i-1$, comme $i,j>k$ alors $a_{ij}^{(k)}=0$ et $i>j+1>k+1$, c'est-à-dire $i-k>1$ et donc $a_{ik}^{(k)}=0$ et $\ell_{ik}^{(k)}=0$. Donc $a_{ij}^{(k+1)}=0$.
       - ⋆ si $j>i+1$, comme $i,j>k$ alors $a_{ij}^{(k)}=0$ et $j>i+1>k+1$, c'est-à-dire $j-k>1$ et donc $a_{kj}^{(k)}=0$. Donc $a_{ij}^{(k+1)}=0$.
2. Les coefficients $(\alpha_1,\alpha_2,\dots,\alpha_n)$, $(\beta_2,\beta_3,\dots,\beta_n)$ et $(\gamma_1,\gamma_2,\dots,\gamma_{n-1})$ se calculent en imposant l'égalité $\mathbb{LU}=\mathbb{A}$. L'algorithme se déduit en parcourant les étapes de la méthode de GAUSS :

$$\mathbb{A}^{(0)} = \begin{pmatrix} a_1 & c_1 & 0 & \dots & \dots & 0 \\ b_2 & a_2 & c_2 & \ddots & & \vdots \\ 0 & b_3 & a_3 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & b_{n-1} & a_{n-1} & c_{n-1} \\ 0 & \dots & \dots & 0 & b_n & a_n \end{pmatrix} \xrightarrow[\beta_2 = \frac{b_2}{a_1}]{L_2 \leftarrow L_2 - \beta_2 L_1} \mathbb{A}^{(1)} = \begin{pmatrix} \alpha_1 = a_1 & \gamma_1 = c_1 & 0 & \dots & \dots & 0 \\ 0 & \alpha_2 = a_2 - \beta_2 c_1 & \gamma_2 = c_2 & \ddots & & \vdots \\ 0 & b_3 & a_3 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & b_{n-1} & a_{n-1} & c_{n-1} \\ 0 & \dots & \dots & 0 & b_n & a_n \end{pmatrix}$$

$$\xrightarrow[\beta_3 = \frac{b_3}{\alpha_2}]{L_3 \leftarrow L_3 - \beta_3 L_2} \mathbb{A}^{(2)} = \begin{pmatrix} \alpha_1 = a_1 & \gamma_1 = c_1 & 0 & \dots & \dots & 0 \\ 0 & \alpha_2 = a_2 - \beta_2 c_1 & \gamma_2 = c_2 & \ddots & & \vdots \\ 0 & 0 & \alpha_3 = a_3 - \beta_3 c_2 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & b_{n-1} & a_{n-1} & c_{n-1} \\ 0 & \dots & \dots & 0 & b_n & a_n \end{pmatrix} \xrightarrow[\beta_4 = \frac{b_4}{\alpha_3}]{L_4 \leftarrow L_4 - \beta_4 L_3} [\cdots]$$

$$[\cdots] \xrightarrow[\beta_n = \frac{b_n}{\alpha_n}]{L_n \leftarrow L_n - \beta_n L_{n-1}} \mathbb{A}^{(n-1)} = \begin{pmatrix} \alpha_1 = a_1 & \gamma_1 = c_1 & 0 & \dots & & \dots & 0 \\ 0 & \alpha_2 = a_2 - \beta_2 c_1 & \gamma_2 = c_2 & \ddots & & & \vdots \\ 0 & 0 & \alpha_3 = a_3 - \beta_3 c_2 & \ddots & & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & & \ddots & 0 \\ \vdots & & \ddots & 0 & \alpha_{n-1} = a_{n-1} - \beta_{n-1} c_{n-2} & & \gamma_{n-1} = c_{n-1} \\ 0 & \dots & \dots & 0 & 0 & & \alpha_n = a_n - \beta_n c_{n-1} \end{pmatrix}$$

Donc $\gamma_i = c_i$ pour $i = 1, \dots, n$, $\alpha_1 = a_1$ et on définie par récurrence

$$\begin{cases} \beta_i &= \frac{b_i}{\alpha_{i-1}} \\ \alpha_i &= a_i - \beta_i c_{i-1} \end{cases} \qquad \text{pour } i = 2, \dots, n.$$

3. La résolution du système linéaire $\mathbb{A}\mathbf{x} = \mathbf{f}$ se ramène à la résolution des deux systèmes linéaires $\mathbb{L}\mathbf{y} = \mathbf{f}$ et $\mathbb{U}\mathbf{x} = \mathbf{y}$, pour lesquels on obtient les formules suivantes :

$$\begin{cases} y_1 = f_1, \\ y_i = f_i - \beta_i y_{i-1}, \quad \text{pour } i = 2, \dots, n, \end{cases} \qquad \textit{i.e.} \qquad \begin{cases} y_1 = f_1, \\ y_i = f_i - \frac{b_i}{a_i - \beta_i c_{i-1}} y_{i-1}, \quad \text{pour } i = 2, \dots, n; \end{cases}$$

$$\begin{cases} x_n = \frac{y_n}{\alpha_n}, \\ x_i = \frac{y_i - \gamma_i x_{i+1}}{\alpha_i}, \quad \text{pour } i = n-1, \dots, 1, \end{cases} \qquad \textit{i.e.} \qquad \begin{cases} x_n = \frac{y_n}{\alpha_n}, \\ x_i = \frac{y_i - c_i x_{i+1}}{a_i - \beta_i c_{i-1}}, \quad \text{pour } i = n-1, \dots, 2, \\ x_1 = \frac{y_i - c_i x_2}{a_1}. \end{cases}$$

Exercice 5.18

Soit les systèmes linéaires

$$\begin{cases} 4x_1 + 3x_2 + 3x_3 = 10 \\ 3x_1 + 4x_2 + 3x_3 = 10 \\ 3x_1 + 3x_2 + 4x_3 = 10 \end{cases} \tag{5.1}$$

$$\begin{cases} 4x_1 + x_2 + x_3 = 6 \\ x_1 + 4x_2 + x_3 = 6 \\ x_1 + x_2 + 4x_3 = 6 \end{cases} \tag{5.2}$$

1. Rappeler une condition suffisante de convergence pour les méthodes de JACOBI et de GAUSS-SEIDEL. Rappeler une autre condition suffisante de convergence pour la méthode de GAUSS-SEIDEL (mais non pour la méthode de JACOBI). Les systèmes (5.1) et (5.2) vérifient-ils ces conditions ?
2. Écrire les méthodes de JACOBI et de GAUSS-SEIDEL pour ces deux systèmes linéaires.
3. On illustrera les résultats théoriques de convergence/non-convergence de ces deux schémas en prenant comme point de départ le vecteur $(x_1, x_2, x_3) = (0, 0, 0)$ et en calculant les 3 premiers itérés :
   3.1. avec la méthode de JACOBI pour le système (5.1),

3.2. avec la méthode de GAUSS-SEIDEL pour le système (5.1),

3.3. avec la méthode de JACOBI pour le système (5.2),

3.4. avec la méthode de GAUSS-SEIDEL pour le système (5.2).

4. On comparera le résultat obtenu avec la solution exacte (qu'on calculera à l'aide de la méthode d'élimination de GAUSS).

CORRECTION DE L'EXERCICE 5.18. Écrivons les deux systèmes sous forme matricielle $\mathbb{A}\mathbf{x} = \mathbf{b}$:

$$\underbrace{\begin{pmatrix} 4 & 3 & 3 \\ 3 & 4 & 3 \\ 3 & 3 & 4 \end{pmatrix}}_{\mathbb{A}_1} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 10 \\ 10 \\ 10 \end{pmatrix} \quad \text{et} \quad \underbrace{\begin{pmatrix} 4 & 1 & 1 \\ 1 & 4 & 1 \\ 1 & 1 & 4 \end{pmatrix}}_{\mathbb{A}_2} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 6 \\ 6 \\ 6 \end{pmatrix}$$

1. Rappelons deux propriétés de convergence :
   - ⋆ Si la matrice $\mathbb{A}$ est à diagonale dominante stricte, les méthodes de JACOBI et de GAUSS-SEIDEL convergent.
   - ⋆ Si la matrice $\mathbb{A}$ est symétrique et définie positive, la méthode de GAUSS-SEIDEL converge.

   Comme $4 > 1+1$, la matrice $\mathbb{A}_2$ est à diagonale dominante stricte : les méthodes de JACOBI et de GAUSS-SEIDEL convergent.

   Comme $4 < 3+3$, la matrice $\mathbb{A}_1$ n'est pas à diagonale dominante stricte : les méthodes de JACOBI et de GAUSS-SEIDEL peuvent ne pas converger. Cependant elle est symétrique et définie positive (car les valeurs propres [1] sont $\lambda_1 = \lambda_2 = 1$ et $\lambda_3 = 10$) : la méthode de GAUSS-SEIDEL converge.
2. Pour les systèmes donnés les méthodes de JACOBI et GAUSS-SEIDEL s'écrivent

	$\mathbb{A}_1\mathbf{x} = \mathbf{b}$	$\mathbb{A}_2\mathbf{x} = \mathbf{b}$
JACOBI	$\begin{pmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ x_3^{(k+1)} \end{pmatrix} = \frac{1}{4} \begin{pmatrix} 10 - 3x_2^{(k)} - 3x_3^{(k)} \\ 10 - 3x_1^{(k)} - 3x_3^{(k)} \\ 10 - 3x_1^{(k)} - 3x_2^{(k)} \end{pmatrix}$	$\begin{pmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ x_3^{(k+1)} \end{pmatrix} = \frac{1}{4} \begin{pmatrix} 6 - x_2^{(k)} - x_3^{(k)} \\ 6 - x_1^{(k)} - x_3^{(k)} \\ 6 - x_1^{(k)} - x_2^{(k)} \end{pmatrix}$
Gauss-SEIDEL	$\begin{pmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ x_3^{(k+1)} \end{pmatrix} = \frac{1}{4} \begin{pmatrix} 10 - 3x_2^{(k)} - 3x_3^{(k)} \\ 10 - 3x_1^{(k+1)} - 3x_3^{(k)} \\ 10 - 3x_1^{(k+1)} - 3x_2^{(k+1)} \end{pmatrix}$	$\begin{pmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ x_3^{(k+1)} \end{pmatrix} = \frac{1}{4} \begin{pmatrix} 6 - x_2^{(k)} - x_3^{(k)} \\ 6 - x_1^{(k+1)} - x_3^{(k)} \\ 6 - x_1^{(k+1)} - x_2^{(k+1)} \end{pmatrix}$

3. On obtient les suites suivantes

   3.1. JACOBI pour le système (5.1) :

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(0)} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \Longrightarrow \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(1)} = \frac{1}{4} \begin{pmatrix} 10 - 3\times 0 - 3\times 0 \\ 10 - 3\times 0 - 3\times 0 \\ 10 - 3\times 0 - 3\times 0 \end{pmatrix} = \begin{pmatrix} \frac{5}{2} \\ \frac{5}{2} \\ \frac{5}{2} \end{pmatrix}$$

$$\Longrightarrow \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(2)} = \frac{1}{4} \begin{pmatrix} 10 - 3\times \frac{5}{2} - 3\times \frac{5}{2} \\ 10 - 3\times \frac{5}{2} - 3\times \frac{5}{2} \\ 10 - 3\times \frac{5}{2} - 3\times \frac{5}{2} \end{pmatrix} = \begin{pmatrix} -\frac{5}{4} \\ -\frac{5}{4} \\ -\frac{5}{4} \end{pmatrix} \Longrightarrow \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(3)} = \frac{1}{4} \begin{pmatrix} 10 - 3\times \frac{-5}{4} - 3\times \frac{-5}{4} \\ 10 - 3\times \frac{-5}{4} - 3\times \frac{-5}{4} \\ 10 - 3\times \frac{-5}{4} - 3\times \frac{-5}{4} \end{pmatrix} = \begin{pmatrix} \frac{35}{8} \\ \frac{35}{8} \\ \frac{35}{8} \end{pmatrix}$$

   3.2. GAUSS-SEIDEL pour le système (5.1) :

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(0)} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \Longrightarrow \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(1)} = \frac{1}{4} \begin{pmatrix} 10 - 3\times 0 - 3\times 0 \\ 10 - 3\times \frac{5}{2} - 3\times 0 \\ 10 - 3\times \frac{5}{2} - 3\times \frac{5}{8} \end{pmatrix} = \begin{pmatrix} \frac{5}{2} \\ \frac{5}{8} \\ \frac{5}{32} \end{pmatrix}$$

$$\Longrightarrow \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(2)} = \frac{1}{4} \begin{pmatrix} 10 - 3\times \frac{5}{8} - 3\times \frac{5}{32} \\ 10 - 3\times \frac{245}{128} - 3\times \frac{5}{32} \\ 10 - 3\times \frac{245}{128} - 3\times \frac{485}{512} \end{pmatrix} = \begin{pmatrix} \frac{245}{128} \\ \frac{485}{512} \\ \frac{725}{2048} \end{pmatrix} \Longrightarrow \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(3)} = \begin{pmatrix} \frac{12485}{8192} \\ \frac{35765}{32768} \\ \frac{70565}{131072} \end{pmatrix}$$

1. $\det \mathbb{A}_1(\lambda) = (4-\lambda)^3 + 27 + 27 - 9(4-\lambda) - 9(4-\lambda) - 9(4-\lambda) = 64 - 48\lambda + 12\lambda^2 - \lambda^3 + 54 - 108 + 27\lambda = -\lambda^3 + 12\lambda^2 - 21\lambda + 10$. Une racine évidente est $\lambda = 1$ et on obtient $\det \mathbb{A}_1(\lambda) = (\lambda - 1)(-\lambda^2 + 11\lambda - 10) = (\lambda - 1)^2(\lambda - 10)$.

3.3. JACOBI pour le système (5.2) :

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(0)} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \Longrightarrow \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(1)} = \frac{1}{4}\begin{pmatrix} 6-1\times 0-1\times 0 \\ 6-1\times 0-1\times 0 \\ 6-1\times 0-1\times 0 \end{pmatrix} = \begin{pmatrix} \frac{3}{2} \\ \frac{3}{2} \\ \frac{3}{2} \end{pmatrix}$$

$$\Longrightarrow \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(2)} = \frac{1}{4}\begin{pmatrix} 6-1\times \frac{3}{2}-1\times \frac{3}{2} \\ 6-1\times \frac{3}{2}-1\times \frac{3}{2} \\ 6-1\times \frac{3}{2}-1\times \frac{3}{2} \end{pmatrix} = \begin{pmatrix} \frac{3}{4} \\ \frac{3}{4} \\ \frac{3}{4} \end{pmatrix} \Longrightarrow \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(3)} = \frac{1}{4}\begin{pmatrix} 6-1\times \frac{3}{4}-1\times \frac{3}{4} \\ 6-1\times \frac{3}{4}-1\times \frac{3}{4} \\ 6-1\times \frac{3}{4}-1\times \frac{3}{4} \end{pmatrix} = \begin{pmatrix} \frac{9}{8} \\ \frac{9}{8} \\ \frac{9}{8} \end{pmatrix}$$

3.4. GAUSS-SEIDEL pour le système (5.2) :

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(0)} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \Longrightarrow \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(1)} = \frac{1}{4}\begin{pmatrix} 6-1\times 0-1\times 0 \\ 6-1\times \frac{3}{2}-1\times 0 \\ 6-1\times \frac{3}{2}-1\times \frac{9}{8} \end{pmatrix} \begin{pmatrix} \frac{3}{2} \\ \frac{9}{8} \\ \frac{27}{32} \end{pmatrix}$$

$$\Longrightarrow \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(2)} = \frac{1}{4}\begin{pmatrix} 6-1\times \frac{9}{8}-1\times \frac{27}{32} \\ 6-1\times \frac{129}{128}-1\times \frac{27}{32} \\ 6-1\times \frac{129}{128}-1\times \frac{531}{512} \end{pmatrix} = \begin{pmatrix} \frac{129}{128} \\ \frac{531}{512} \\ \frac{2025}{2048} \end{pmatrix} \Longrightarrow \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}^{(3)} = \frac{1}{4}\begin{pmatrix} 6-1\times \frac{531}{512}-1\times \frac{2025}{2048} \\ 6-1\times \frac{8139}{8192}-1\times \frac{2025}{2048} \\ 6-1\times \frac{8139}{8192}-1\times \frac{32913}{32768} \end{pmatrix} = \begin{pmatrix} \frac{8139}{8192} \\ \frac{32913}{32768} \\ \frac{131139}{131072} \end{pmatrix}$$

4. Calcul de la solution exacte à l'aide de la méthode d'élimination de GAUSS :
   - ⋆ Système (5.1) :

$$\left(\begin{array}{ccc|c} 4 & 3 & 3 & 10 \\ 3 & 4 & 3 & 10 \\ 3 & 3 & 4 & 10 \end{array}\right) \xrightarrow[L_3\leftarrow L_3-\frac{3}{4}L_1]{L_2\leftarrow L_2-\frac{3}{4}L_1} \left(\begin{array}{ccc|c} 4 & 3 & 3 & 10 \\ 0 & 7/4 & 3/4 & 5/2 \\ 0 & 3/4 & 7/4 & 5/2 \end{array}\right) \xrightarrow{L_3\leftarrow L_3-\frac{3/4}{7/4}L_2} \left(\begin{array}{ccc|c} 4 & 3 & 3 & 10 \\ 0 & 7/4 & 3/4 & 5/2 \\ 0 & 0 & 10/7 & 10/7 \end{array}\right) \Longrightarrow \mathbf{x} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

   - ⋆ Système (5.2) :

$$\left(\begin{array}{ccc|c} 4 & 1 & 1 & 6 \\ 1 & 4 & 1 & 6 \\ 1 & 1 & 4 & 6 \end{array}\right) \xrightarrow[L_3\leftarrow L_3-\frac{1}{4}L_1]{L_2\leftarrow L_2-\frac{1}{4}L_1} \left(\begin{array}{ccc|c} 4 & 1 & 1 & 6 \\ 0 & 15/4 & 3/4 & 9/2 \\ 0 & 3/4 & 15/4 & 9/2 \end{array}\right) \xrightarrow{L_3\leftarrow L_3-\frac{3/4}{15/4}L_2} \left(\begin{array}{ccc|c} 4 & 1 & 1 & 6 \\ 0 & 15/4 & 3/4 & 9/2 \\ 0 & 0 & 18/5 & 18/5 \end{array}\right) \Longrightarrow \mathbf{x} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

A. Python : guide de survie pour les TP

Le but de ce chapitre est de fournir suffisamment d'informations pour pouvoir tester les méthodes numériques vues dans ce polycopié. **Il n'est ni un manuel de Python ni une initiation à la programmation.** On suppose que vous avez déjà des notions de programmation et de manipulation de fichier.

Python est un langage développé dans les années 1 980 (le nom est dérivé de la série télévisée britannique des *Monty Python's Flying Circus*). Il est disponible pour tous les principaux systèmes d'exploitation (Linux, Unix, Windows, Mac OS, etc.). Un programme écrit sur un système fonctionne sans modification sur tous les systèmes. Les programmes Python ne sont pas compilés en code machine, mais sont gérés par un interpréteur. Le grand avantage d'un langage interprété est que les programmes peuvent être testés et mis au point rapidement, ce qui permet à l'utilisateur de se concentrer davantage sur les principes sous-jacents du programme et moins sur la programmation elle-même. Cependant, un programme Python peut être exécuté uniquement sur les ordinateurs qui ont installé l'interpréteur Python.

A.1. Obtenir Python et son éditeur IDLE

Pour installer Python il suffit de télécharger la version 2.6 qui correspond au système d'exploitation (Windows ou Mac) à l'adresse `www.python.org`. Pour ceux qui est des systèmes Linux, il est très probable que Python est déjà installé.

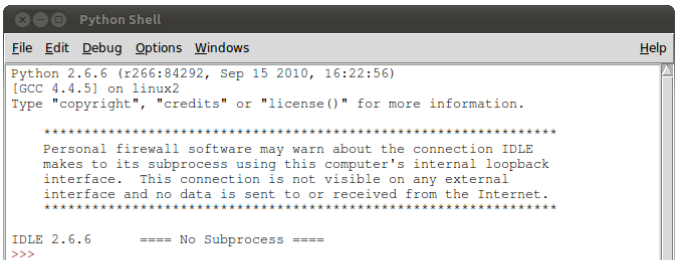
Si on n'a jamais programmé, le plus simple pour exécuter les instructions Python est d'utiliser des environnements spécialisés comme IDLE ou IDLEX (ou encore SPYDER). Ces environnements se composent d'une fenêtre appelée indifféremment *console*, *shell* ou *terminal* Python.

A.1.1. Utilisation de base d'Idle

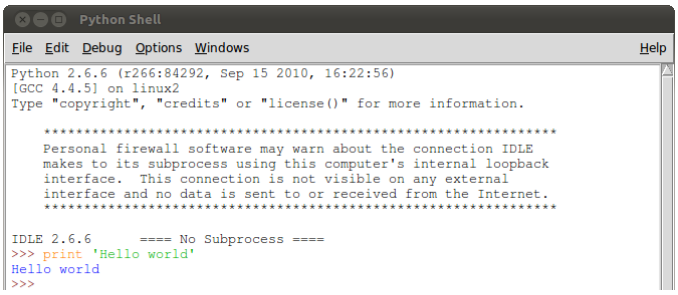
Pour commencer on va démarrer Python en lançant IDLE :

- ⋆ sous Windows : menu «Démarrer» → programme «Python» → «IDLE»
- ⋆ sous Ubuntu : menu «Applications» → menu «Programmation» → «IDLE»
- ⋆ sous Mac/Linux : ouvrir un terminal/console et taper `idle-python2.6`

Une nouvelle fenêtre va s'ouvrir, c'est la fenêtre principale d'IDLE appelée la fenêtre de l'INTERPRÉTEUR :



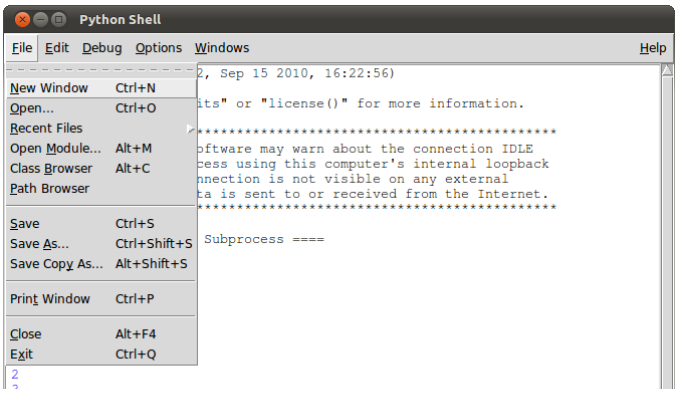
L'INTERPRÉTEUR permet d'entrer directement des commandes et dès qu'on écrit une commande, Python l'exécute et renvoie instantanément le résultat. L'invite de commande se compose de trois chevrons (>>>) et représente le prompt : cette marque visuelle indique que Python est prêt à lire une commande. Il suffit de saisir à la suite une instruction puis d'appuyer sur la touche «Entrée». Pour commencer, comme le veux la tradition informatique, on va demander à Python d'afficher les fameux mots «Hello world» :



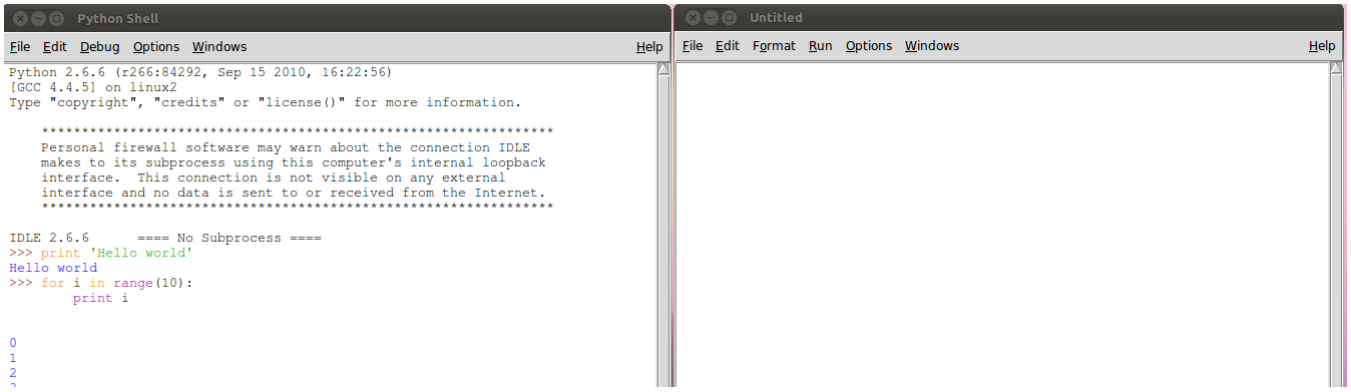
La console Python fonctionne comme une simple calculatrice : on peut saisir une expression dont la valeur est renvoyée dès qu'on presse la touche «Entrée». Si on observe l'image suivante, on voit le résultat affiché après l'entrée de commandes supplémentaires.

Pour naviguer dans l'historique des instructions saisies dans l'INTERPRÉTEUR on peut utiliser les raccourcis `Alt+p` (p comme *previous*) et `Alt+n` (n comme *next*). [1]

Si on ferme Python et qu'on le relance, comment faire en sorte que l'ordinateur se souvienne de ce que nous avons tapé ? On ne peut pas sauvegarder directement ce qui se trouve dans la fenêtre de l'interpréteur, parce que cela comprendrait à la fois les commandes tapées et les réponses du système. Il faut alors avoir un fichier avec uniquement les commandes qu'on a tapées et sauver le tout comme un document. Ainsi plus tard on pourra ouvrir ce fichier et lancer Python sans avoir à retaper toutes les commandes. Tout d'abord, commençons par un support propre en ouvrant une nouvelle fenêtre.

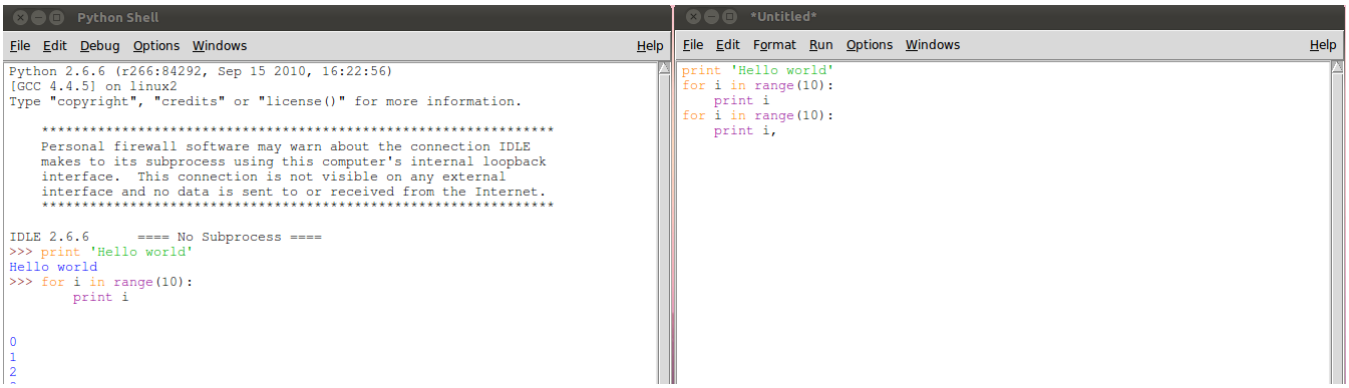


Voici ce que cela donne :

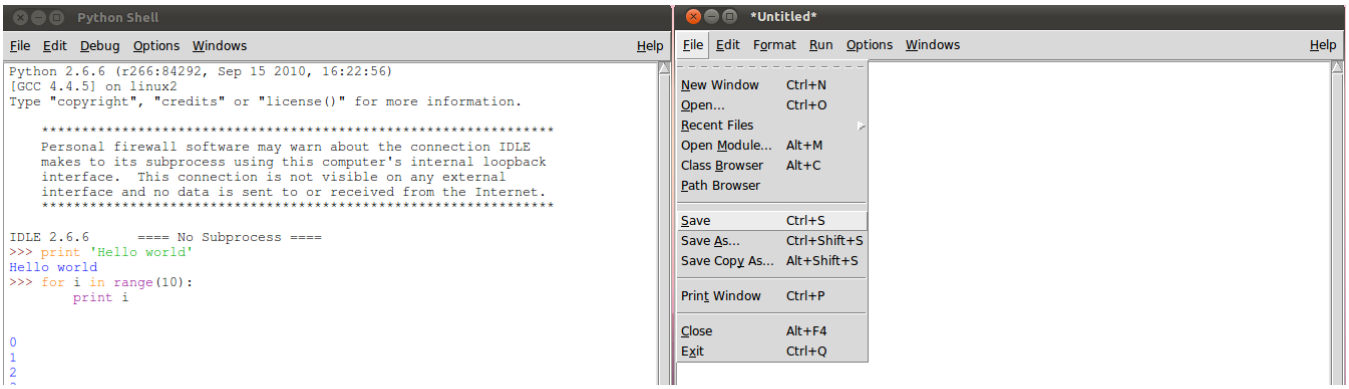


1. Il ne s'agit pas, pour l'instant, de s'occuper des règles exactes de programmation, mais seulement d'expérimenter le fait d'entrer des commandes dans Python.

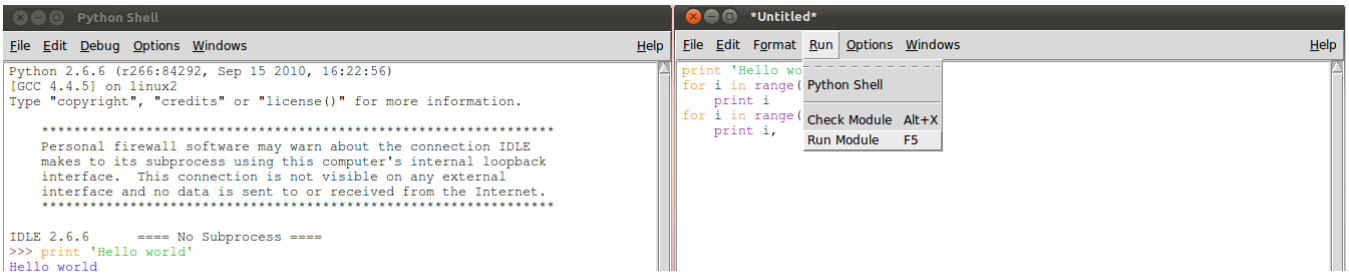
On voit qu'il n'y a rien dans cette nouvelle fenêtre (pas d'en-tête comme dans l'INTERPRÉTEUR). Ce qui veut dire que ce fichier est uniquement pour les commandes : Python n'interviendra pas avec ses réponses lorsque on écrira le programme et ce tant que on ne le lui demandera pas. On appellera cela la fenêtre de PROGRAMME, pour la différencier de la fenêtre de l'INTERPRÉTEUR. En fait, ce qu'on veut faire, c'était de sauver les quelques instructions qu'on a essayées dans l'interpréteur. Alors faisons-le soit en tapant soit en copiant-collant ces commandes dans la fenêtre PROGRAMME :



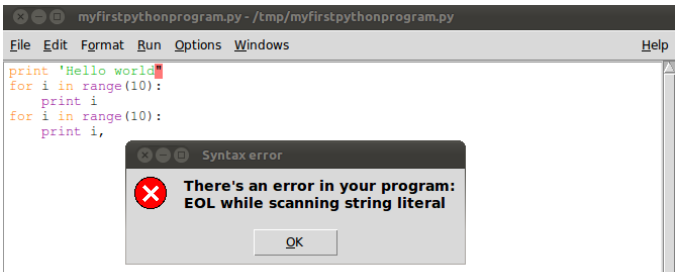
On note qu'on s'est débarrassés du prompt de Python (>>>). Sauvons maintenant le fichier : la commande «Save» (Sauver) se trouve dans le menu «File» (Fichier) ou utiliser le raccourcis `Ctrl+S` :



Ayant sauvé le programme, pour le faire tourner et afficher les résultats dans la fenêtre de l'INTERPRÉTEUR il suffit d'utiliser la commande «Run script» (lancer le script) dans le menu «Run» de la fenêtre PROGRAMME ou appuyer sur la touche «F5»



Si on a fait une faute de frappe, Python le remarque et demande de corriger. Il est souvent assez pertinent pour diriger vers le problème et dans le cas ci-dessous il dit qu'on a oublié quelque chose à la fin de la ligne : il faut remplacer `"` par `'`.



Cette faute de frappe étant corrigée, on fait tourner le programme et on regarde le résultat dans l'INTERPRÉTEUR :

Maintenant qu'on a sauvé le programme, on est capable de le recharger : on va tout fermer et relancer IDLE. La commande «Open» (Ouvrir) se trouve dans le menu «File» (Fichier). Si tout se passe bien, on va avoir une nouvelle fenêtre PROGRAMME avec l'ancien programme.

A.2. Notions de base de Python

Indentation Le corps d'un bloc de code (boucles, sous-routines, etc.) est défini par son indentation : l'indentation est une partie intégrante de la syntaxe de Python.

Commentaires Le symbole dièse (#) indique le début d'un commentaire : tous les caractères entre # et la fin de la ligne sont ignorés par l'interpréteur.

Variables et affectation Dans la plupart des langages informatiques, le nom d'une variable représente une valeur d'un type donné stockée dans un emplacement de mémoire fixe. La valeur peut être modifiée, mais pas le type. Ce n'est pas le cas en Python, où les variables sont typées dynamiquement. La session interactive suivante avec l'INTERPRÉTEUR Python illustre ce propos (>>> est le prompt) :

```
>>> b = 2 # b is an integer
>>> print(b)
2
>>> b = b*2.0 # b is a float
>>> print(b)
4.0
```

L'affectation `b = 2` crée une association entre le nom *b* et le nombre entier 2. La déclaration suivante `b*2.0` évalue l'expression et associe le résultat à *b* ; l'association d'origine avec l'entier 2 est détruite. Maintenant *b* se réfère à la valeur en virgule flottante 4.0. Il faut bien prendre garde au fait que l'instruction d'affectation (=) n'a pas la même signification que le symbole d'égalité (=) en mathématiques (ceci explique pourquoi l'affectation de 3 à `x`, qu'en Python s'écrit `x = 3`, en algorithmique se note souvent $x \leftarrow 3$). On peut aussi effectuer des affectations parallèles :

```
>>> a, b = 128, 256
>>> print(a)
128
>>> print(b)
256
```

⚠ **ATTENTION.** *Python est sensible à la casse. Ainsi, les noms n et N représentent différents objets. Les noms de variables peuvent être non seulement des lettres, mais aussi des mots ; ils peuvent contenir des chiffres (à condition toutefois de ne pas commencer par un chiffre), ainsi que certains caractères spéciaux comme le tiret bas «_» (appelé underscore en anglais).*

Chaîne de caractères (Strings) Une *chaîne de caractères* est une séquence de caractères entre guillemets (simples ou doubles). Les chaînes de caractères sont concaténées avec l'opérateur plus (+), tandis que l'opérateur (:) est utilisé pour extraire une portion de la chaîne. Voici un exemple :

```
>>> string1 = 'Press return to exit'
>>> string2 = 'the program'
>>> print string1 + ' ' + string2 # Concatenation
Press return to exit the program
>>> print string1[0:12] # Slicing
Press return
```

Une chaîne de caractères est un objet immuable, *i.e.* ses caractères ne peuvent pas être modifiés par une affectation, et sa longueur est fixe. Si on essaye de modifier un caractère d'une chaîne de caractères, Python renvoie une erreur comme dans l'exemple suivant :

```
>>> s = 'Press return to exit'
>>> s[0] = 'p'
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: 'str' object does not support item assignment
```

Listes Une liste est une suite d'objets, rangés dans un certain ordre. Chaque objet est séparé par une virgule et la suite est encadrée par des crochets. Une liste n'est pas forcement homogène : elle peut contenir des objets de types différents les uns des autres. La première manipulation que l'on a besoin d'effectuer sur une liste, c'est d'en extraire et/ou modifier un élément : la syntaxe est `ListName[index]`. Voici un exemple :

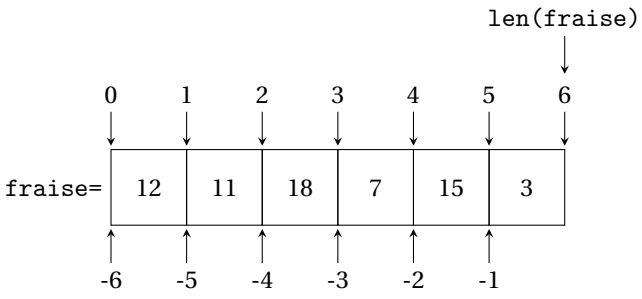
```
>>> fraise = [12, 10, 18, 7, 15, 3] # Create a list
>>> print fraise
[12, 10, 18, 7, 15, 3]
>>> fraise[2]
18
>>> fraise[1] = 11
>>> print fraise
[12, 11, 18, 7, 15, 3]
```

⚠ **ATTENTION.** *En Python, les éléments d'une liste sont* indexés à partir de 0 *et non de* 1.

Si on tente d'extraire un élément avec un index dépassant la taille de la liste, Python renvoi un message d'erreur :

```
>>> fraise[0], fraise[1], fraise[2], fraise[3], fraise[4], fraise[5]
(12, 11, 18, 7, 15, 3)
>>> fraise[6]
Traceback (most recent call last):
  File "<pyshell#4>", line 1, in <module>
    fraise[6]
IndexError: list index out of range
```

On peut extraire une sous-liste en déclarant l'indice de début (inclus) et l'indice de fin (exclu), séparés par deux-points : `ListName[i:j]`, ou encore une sous-liste en déclarant l'indice de début (inclus), l'indice de fin (exclu) et le pas, séparés par des deux-points : `ListName[i:j:k]`. Cette opération est connue sous le nom de *slicing* (en anglais). Un petit dessin et quelques exemples permettrons de bien comprendre cette opération fort utile :




```
>>> fraise[2:4]
[18, 7]
>>> fraise[2:]
```

```
[18, 7, 15, 3]
>>> fraise[:2]
[12, 11]
>>> fraise[:]
[12, 11, 18, 7, 15, 3]
>>> fraise[2:5]
[18, 7, 15]
>>> fraise[2:6]
[18, 7, 15, 3]
>>> fraise[2:7]
[18, 7, 15, 3]
>>> fraise[2:6:2]
[18, 15]
>>> fraise[-2:-4]
[]
>>> fraise[-4:-2]
[18, 7]
>>> fraise[-1]
3
```

À noter que lorsqu'on utilise des tranches, les dépassements d'indices sont licites.

Voici quelques opérations et méthodes très courantes associées aux listes :

`a.append(x)`	ajoute l'élément `x` en fin de la liste `a`
`a.extend(L)`	ajoute les éléments de la liste `L` en fin de la liste `a`, équivaut à `a + L`
`a.insert(i,x)`	ajoute l'élément `x` en position `i` de la liste `a`, équivaut à `a[i:i]=x`
`a.remove(x)`	supprime la première occurrence de l'élément `x` dans la liste `a`
`a.pop([i])`	supprime l'élément d'indice `i` dans la liste `a` et le renvoi
`a.index(x)`	renvoie l'indice de la première occurrence de l'élément `x` dans la liste `a`
`a.count(x)`	renvoie le nombre d'occurrence de l'élément `x` dans la liste `a`
`a.sort(x)`	modifie la liste `a` en la triant
`a.reverse(x)`	modifie la liste `a` en inversant les éléments
`len(a)`	renvoie le nombre d'éléments de la liste `a`
`x in a`	renvoi `True` si la liste `a` contient l'élément `x`, `True` sinon
`x not in a`	renvoi `True` si la liste `a` ne contient pas l'élément `x`, `True` sinon
`max(a)`	renvoi le plus grand élément de la liste `a`
`min(a)`	renvoi le plus petit élément de la liste `a`

```
>>> a = [2, 37, 20, 83, -79, 21] # Create a list
>>> print a
[2, 37, 20, 83, -79, 21]
>>> a.append(100) # Append 100 to list
>>> print a
[2, 37, 20, 83, -79, 21, 100]
>>> L = [17, 34, 21]
>>> a.extend(L)
>>> print a
[2, 37, 20, 83, -79, 21, 100, 17, 34, 21]
>>> a.count(21)
2
>>> a.remove(21)
>>> print a
[2, 37, 20, 83, -79, 100, 17, 34, 21]
>>> a.count(21)
1
>>> a.pop(4)
-79
>>> print a
[2, 37, 20, 83, 100, 17, 34, 21]
>>> a.index(100)
4
>>> a.reverse()
>>> print a
[21, 34, 17, 100, 83, 20, 37, 2]
```

```
>>> a.sort()
>>> print a
[2, 17, 20, 21, 34, 37, 83, 100]
>>> len(a) # Determine length of list
8
>>> a.insert(2,7) # Insert 7 in position 2
>>> print a
[2, 17, 7, 20, 21, 34, 37, 83, 100]
>>> a[0] = 21 # Modify selected element
>>> print a
[21, 17, 7, 20, 21, 34, 37, 83, 100]
>>> a[2:4] = [-2,-5,-1978] # Modify selected elements
>>> print a
[21, 17, -2, -5, -1978, 21, 34, 37, 83, 100]
```

⚠ **ATTENTION.** *Si `a` est une liste, la commande `b=a` ne crée pas un nouvel objet `b` mais simplement une référence (pointeur) vers `a`. Ainsi, tout changement effectué sur `b` sera répercuté sur `a` aussi ! Pour créer une copie `c` de la liste `a` qui soit vraiment indépendante on utilisera la commande `deepcopy` du module `copy` comme dans l'exemple suivant :*

```
>>> import copy
>>> a = [1.0, 2.0, 3.0]
>>> b = a # 'b' is an alias of 'a'
>>> b[0] = 5.0 # Change 'b'
>>> print a # The change is reflected in 'a'
[5.0, 2.0, 3.0]
>>> print b
[5.0, 2.0, 3.0]
>>> a = [1.0, 2.0, 3.0]
>>> c = copy.deepcopy(a) # 'c' is an independent copy of 'a'
>>> c[0] = 5.0 # Change 'c'
>>> print a # 'a' is not affected by the change
[1.0, 2.0, 3.0]
>>> print c
[5.0, 2.0, 3.0]
```

Qu'est-ce qui se passe lorsque on copie une liste `a` avec la commande `b=a` ? En effet, une liste fonctionne comme un carnet d'adresses qui contient les emplacements en mémoire des différents éléments de la liste. Lorsque on écrit `b=a` on dit que `b` contient les mêmes adresses que `a` (on dit que les deux listes «pointent» vers le même objet). Ainsi, lorsqu'on modifie la valeur de l'objet, la modification sera visible depuis les deux alias.

Matrices Les matrices peuvent être représentées comme des listes imbriquées : chaque ligne est un élément d'une liste. Par exemple, le code

```
>>> a = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
```

définit `a` comme la matrice 3×3

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}.$$

La commande `len` (comme *length*) renvoie la longueur d'une liste. On obtient donc le nombre de ligne de la matrice avec `len(a)` et son nombre de colonnes avec `len(a[0])`. En effet,

```
>>> print a
[[1, 2, 3], [4, 5, 6], [7, 8, 9]]
>>> print a[1] # Print second row (element 1)
[4, 5, 6]
>>> print a[1][2] # Print third element of second row
6
>>> print len(a)
3
>>> print len(a[0])
3
```

⚠ **Attention.** *Dans Python les indices commences à zéro, ainsi `a[0]` indique la première ligne, `a[1]` la deuxième etc.*

$$\mathbb{A} = \begin{pmatrix} a_{00} & a_{01} & a_{02} & \dots \\ a_{10} & a_{11} & a_{12} & \dots \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix}$$

Fonction `range` La fonction `range` crée un itérateur. Au lieu de créer et garder en mémoire une liste d'entiers, cette fonction génère les entiers au fur et à mesure des besoins :

- ⋆ `range(n)` renvoi un itérateur parcourant $0, 1, 2, \dots, n-1$;
- ⋆ `range(n,m)` renvoi un itérateur parcourant $n, n+1, n+2, \dots, m-1$;
- ⋆ `range(n,m,p)` renvoi un itérateur parcourant $n, n+p, n+2p, \dots, m-1$.

```
>>> range(10)
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
>>> range(0)
[]
>>> range(1)
[0]
>>> range(3,7)
[3, 4, 5, 6]
>>> range(0,20,5)
[0, 5, 10, 15]
>>> range(0,20,-5)
[]
>>> range(0,-20,-5)
[0, -5, -10, -15]
>>> range(20,0,-5)
[20, 15, 10, 5]
```

Instruction `print` Pour afficher à l'écran des objets on utilise la commande `print object1, object2, ...` qui convertis `object1, object2` en chaînes de caractères et les affiche sur la même ligne séparés par des espace. Le retour à la ligne peut être forcé par le caractère `\n`, la tabulation par le caractère `\t` :

```
>>> a = 12345,6789
>>> b = [2, 4, 6, 8]
>>> print a,b
(12345, 6789) [2, 4, 6, 8]
>>> print "a=", a, "\nb=", b
a= (12345, 6789)
b= [2, 4, 6, 8]
>>> print "a=", a, "\tb=", b
a= (12345, 6789) ⟶b= [2, 4, 6, 8]
```

Pour mettre en colonne des nombres on pourra utiliser l'opérateur % : la commande `print '%format1, %format2,...'%(n1,n2,...)` affiche les nombres `n1,n2,...` selon les règles `%format1, %format2,...`. Typiquement on utilise

wd	pour un entier
$w.d$f	pour un nombre en notation *floating point*
$w.d$e	pour un nombre en notation scientifique

où w est la largeur du champ total, d le nombre de chiffres après la virgule. Voici quelques exemples :

```
>>> a = 1234.56789
>>> n = 9876
>>> print '%7.2f' %a
1234.57
>>> print 'n = %6d' %n
n =  9876
>>> print 'n = %06d' %n
n = 009876
>>> print '%12.4e %6d' %(a,n)
 1.2346e+03 9876
```

Opérations arithmétiques Dans Python on a les opérations arithmétiques usuelles :

`+`	Addition
`-`	Soustraction
`*`	Multiplication
`/`	Division
`**`	Exponentiation
`//`	Quotient de la division euclidienne
`%`	Reste de la division euclidienne

Quelques exemples :

```
>>> a = 100
>>> b = 17
>>> c = a-b
>>> a = 2
>>> c = b+a
>>> a,b,c
(2, 17, 19)
>>> a = 3
>>> b = 4
>>> c = a
>>> a = b
>>> b = c
>>> a, b, c
(4, 3, 3)
```

Certains de ces opérations sont aussi définies pour les chaînes de caractères et les listes comme dans l'exemple suivant :

```
>>> s = 'Hello '
>>> t = 'to you'
>>> a = [1, 2, 3]
>>> print 3*s # Repetition
Hello Hello Hello
>>> print 3*a # Repetition
[1, 2, 3, 1, 2, 3, 1, 2, 3]
>>> print a + [4, 5] # Append elements
[1, 2, 3, 4, 5]
>>> print s + t # Concatenation
Hello to you
>>> print 3 + s # This addition makes no sense
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: unsupported operand type(s) for +: 'int' and 'str'
```

Il existe aussi les opérateurs augmentés :

On écrit	Équivaut à
`a += b`	`a = a + b`
`a -= b`	`a = a - b`
`a *= b`	`a = a*b`
`a /= b`	`a = a/b`
`a **= b`	`a = a**b`
`a %= b`	`a = a%b`

Opérateurs de comparaison et connecteurs logiques Les opérateurs de comparaison renvoient `True` si la condition est vérifiée, `False` sinon. Ces opérateurs sont

On écrit	Ça signifie
`<`	$<$
`>`	$>$
`<=`	$\leq$
`>=`	$\geq$
`==`	$=$
`!=`	$\neq$
`in`	$\in$

⚠ **ATTENTION.** *Bien distinguer l'instruction d'affectation = du symbole de comparaison ==.*

Pour combiner des conditions complexes (par exemple $x > -2$ et $x^2 < 5$), on peut combiner des variables booléennes en utilisant les connecteurs logiques :

On écrit	Ça signifie
`and`	et
`or`	ou
`not`	non

Deux nombres de type différents (entier, à virgule flottante, etc.) sont convertis en un type commun avant de faire la comparaison. Dans tous les autres cas, deux objets de type différents sont considérés non égaux. Voici quelques exemples :

```
>>> a = 2 # Integer
>>> b = 1.99 # Floating
>>> c = '2' # String
>>> print a>b
True
>>> print a==c
False
>>> print (a>b) and (a==c)
False
>>> print (a>b) or (a==c)
True
>>> print (a>b) or (a==b)
True
```

A.3. Fonctions et Modules

A.3.1. Fonctions

Supposons de vouloir calculer les images de certains nombres par une fonction polynomiale donnée. Si la fonction en question est un peu longue à saisir, par exemple $f : x \mapsto 2x^7 - x^6 + 5x^5 - x^4 + 9x^3 + 7x^2 + 8x - 1$, il est rapidement fastidieux de la saisir à chaque fois que l'on souhaite calculer l'image d'un nombre par cette fonction. Une première idée est d'utiliser l'historique de l'INTERPRÉTEUR pour éviter de saisir à chaque fois la fonction, néanmoins ce n'est pas très pratique, surtout si on veut y travailler un autre jour. Il est tout à fait possible de définir une fonction (au sens du langage Python) qui ressemble à une fonction mathématique. La syntaxe est la suivante :

```
def FunctionName(parameters):
⟶statements
⟶return values
```

La déclaration d'une nouvelle fonction commence par le mot-clé `def`. Ensuite, toujours sur la même ligne, vient le nom de la fonction (ici `FunctionName`) suivi des paramètres formels [2] de la fonction `parameters`), placés entre parenthèses, le tout terminé par deux-points (on peut mettre autant de paramètres formels qu'on le souhaite et éventuellement aucun). Une fois la première ligne saisie, on appuie sur la touche «Entrée» : le curseur passe à la ligne suivante avec une indentation. Si l'instruction `return` est absente, la fonction renvoi l'objet `None`.

⚠ **ATTENTION.** *Dès que Python atteint l'instruction `return something`, il renvoi l'objet `something` et abandonne aussitôt après l'exécution de la fonction (on parle de code mort pour désigner les lignes qui suivent l'instruction `return`).*

Voici un bêtisier pour mieux comprendre les règles : dans le premier cas il manque les deux-points en fin de ligne, dans le deuxième il manque l'indentation, dans le troisième il manque le mot `return` et donc tout appel de la fonction aura comme réponse `None`, dans le quatrième l'instruction `print 'Hello'` n'est jamais lue par Python car elle apparait après l'instruction `return`.

```
>>> def f(x)
SyntaxError: invalid syntax
>>> def f(x):
return 2*x**7-x**6+5*x**5-x**4+9*x**3+7*x**2+8*x-1
  File "<pyshell#7>", line 2
    return 2*x**7-x**6+5*x**5-x**4+9*x**3+7*x**2+8*x-1
```

2. Les paramètres figurant entre parenthèses dans l'en-tête d'une fonction se nomment *paramètres formels*, par opposition aux paramètres fournis lors de l'appel de la fonction qui sont appelés *paramètres effectifs*.

```
        ^
IndentationError: expected an indented block
>>> def f(x):
⟶2*x**7-x**6+5*x**5-x**4+9*x**3+7*x**2+8*x-1


>>> f(2)
>>> print f(2)
None
>>> def f(x):
⟶a = 2*x**7-x**6+5*x**5-x**4+9*x**3+7*x**2+8*x-1
⟶return a
⟶print 'Hello'

>>> f(2)
451
```

⚠ **ATTENTION.** *Les variables définies à l'intérieur d'une fonction ne sont pas «visibles» depuis l'extérieur de la fonction. On exprime cela en disant qu'une telle variable est locale à la fonction. De plus, si une variable existe déjà avant l'exécution de la fonction, tout se passe comme si, durant l'exécution de la fonction, cette variable était masquée momentanément, puis restituée à la fin de l'exécution de la fonction.*

Dans l'exemple suivant, la variable `x` est une variable locale à la fonction `f` : crée au cours de l'exécution de la fonction `f`, elle est supprimée une fois l'exécution terminée :

```
>>> def f(y):
⟶x = 2
⟶return 4.*y

>>> f(5)
20.0
>>> x
Traceback (most recent call last):
  File "<pyshell#35>", line 1, in <module>
    x
NameError: name 'x' is not defined
```

Dans l'exemple suivant, la variable `x` est une variable qui vaut 6 à l'extérieur de la fonction et 7 au cours de l'exécution de la fonction `f` :

```
>>> x = 6.
>>> def f(y):
⟶x = 7
⟶return x*y

>>> print x
6.0
>>> print f(1.)
7.0
>>> print x
6.0
```

Dans l'exemple suivant la fonction `derivatives` approche les dérivées première et seconde d'une fonction f par les formules

$$f'(x) \simeq \frac{f(x+h)-f(x-h)}{2h}, \qquad f''(x) \simeq \frac{f(x+h)-2f(x)+f(x-h)}{h^2}$$

```
import math
def derivatives(f,x,h):
⟶df = (f(x+h)-f(x-h))/(2.*h)
⟶ddf = (f(x+h)-2.*f(x)+f(x-h))/h**2
⟶return df,ddf
```

Si on veut calculer la valeur des dérivées première et seconde de la fonction $x \mapsto \cos(x)$ en $x = \frac{\pi}{2}$ il suffit d'écrire

```
df, ddf = derivatives(math.cos,math.pi/2,1.0e-5)
print 'First derivative =', df
print 'Second derivative =', ddf
```

⚠ **Attention.** *Si une liste est passée comme paramètre d'une fonction et cette fonction la modifie, cette modification se répercute sur la liste initiale. Si ce n'est pas le résultat voulu, il faut travailler sur une copie de la liste.*

```
def squares(a):
    for i in range(len(a)):
        a[i] = a[i]**2

a = [1,2,3,4]
print a # The output is [1, 2, 3, 4]
squares(a)
print a # The output is [1, 4, 9, 16]
```

A.3.2. Modules

Un module est une collection de fonctions. Pour importer un module, il faut utiliser la commande `import ModuleName`. Il est alors possible d'obtenir une aide sur le module avec la commande `help(ModuleName)`. La liste des fonctions définies dans un module peut être affichée par la commande `dir(ModuleName)`. Les fonctions s'utilisent sous la forme `ModuleName.FunctionName(parameters)`. Il est également possible d'importer le contenu du module sous la forme `from ModuleName import *` et alors les fonctions peuvent être utilisées directement par `FunctionName(parameters)`.

Python offre par défaut une bibliothèque de plus de deux cents modules qui évite d'avoir à réinventer la roue dès que l'on souhaite écrire un programme. Ces modules couvrent des domaines très divers : mathématiques (fonctions mathématiques usuelles, calculs sur les réels, sur les complexes, combinatoire...), administration système, programmation réseau, manipulation de fichiers, etc. Ici on en présente seulement quelques-uns, à savoir ce dont on se servira dans les TP.

Le module `math` Dans Python seulement quelque fonction mathématique est prédéfinie :

`abs(a)`	Valeur absolue de a
`max(suite)`	Plus grande valeur de la suite
`min(suite)`	Plus petite valeur de la suite
`round(a,n)`	Arrondi a à n décimales près
`pow(a,n)`	Exponentiation, renvoi a^n
`sum(L)`	Somme des éléments de la suite
`divmod(a,b)`	Renvoi quotient et reste de la division de a par b
`cmp(a,b)`	Renvoi $\begin{cases} -1 & \text{si } a < b, \\ 0 & \text{si } a = b, \\ 1 & \text{si } a > b. \end{cases}$

Toutes les autres fonctions mathématiques sont définies dans le module `math`. Comme mentionné précédemment, on dispose de plusieurs syntaxex pour importer un module :

```
>>> import math
>>> print math.pi
3.14159265359
>>> print math.sin(math.pi)
1.22464679915e-16
>>> print math.log(1.0)
0.0
```

ou

```
>>> from math import *
>>> print pi
3.14159265359
>>> print sin(pi)
1.22464679915e-16
>>> print log(1.0)
0.0
```

Voici la liste des fonctions définies dans le module `math` :

```
>>> import math
>>> dir(math)
['__doc__', '__name__', '__package__', 'acos', 'acosh', 'asin', 'asinh', 'atan', 'atan2', 'atanh', 'ceil',
    ➥ 'copysign', 'cos', 'cosh', 'degrees', 'e', 'exp', 'fabs', 'factorial', 'floor', 'fmod', 'frexp', '
    ➥fsum', 'hypot', 'isinf', 'isnan', 'ldexp', 'log', 'log10', 'log1p', 'modf', 'pi', 'pow', 'radians',
    ➥'sin', 'sinh', 'sqrt', 'tan', 'tanh', 'trunc']
```

Notons que le module définit les deux constantes π et e.

Le module `matplotlib` pour le tracé de données Le tracé de courbes scientifiques peut se faire à l'aide du module `matplotlib`. Pour l'utiliser, il faut importer le module `pylab`. La référence complète de `matplotlib` est lisible à l'adresse : `http://matplotlib.sourceforge.net/matplotlib.pylab.html`. Il est en particulier recommandé de regarder les "screenshots" (captures d'écrans), qui sont donnés avec le code utilisé pour les générer. Dans ces rappels on ne verra que la représentation de fonction 1D.

⚠ **ATTENTION.** *Lorsque l'on utilise* IDLE, *après la commande* `show()` *nécessaire pour visualiser les graphes, l'interpréteur python se bloque (c'est un bug de l'éditeur). Pour pallier à ce problème on peut utiliser* IDLEX, *téléchargeable à l'adresse* `http://idlex.sourceforge.net`, *qui améliore* IDLE *et qui ne pose pas de problèmes avec* `matplotlib`.

Pour tracer le graphe d'une fonction $f\colon [a,b] \to \mathbb{R}$, Python a besoin d'une grille de points x_i où évaluer la fonction, ensuite il relie entre eux les points $(x_i, f(x_i))$ par des segments. Plus les points sont nombreux, plus le graphe de la fonction spline est proche du graphe de la fonction f. Pour générer les points x_i on peut utiliser l'instruction `linspace(a,b,n)` qui construit la liste de $n+1$ éléments

$$\left[a, a+\frac{b-a}{n}, a+2\frac{b-a}{n}, \ldots, b\right]$$

ou l'instruction `arange(a,b,h)` qui construit la liste de $n = E(\frac{b-a}{h})+1$ éléments

$$[a, a+h, a+2h, \ldots, a+nh]$$

Voici un exemple avec une sinusoïde :

```
from matplotlib.pylab import *
x = linspace(-5,5,101) # x = [-5,-4.9,-4.8,...,5] with 101 elements
y = sin(x) # operation is broadcasted to all elements of the array
plot(x,y)
show()
```

ou encore

```
from matplotlib.pylab import *
x = arange(-5,5,0.1) # x = [-5,-4.9,-4.8,...,5] with 101 elements
y = sin(x) # operation is broadcasted to all elements of the array
plot(x,y)
show()
```

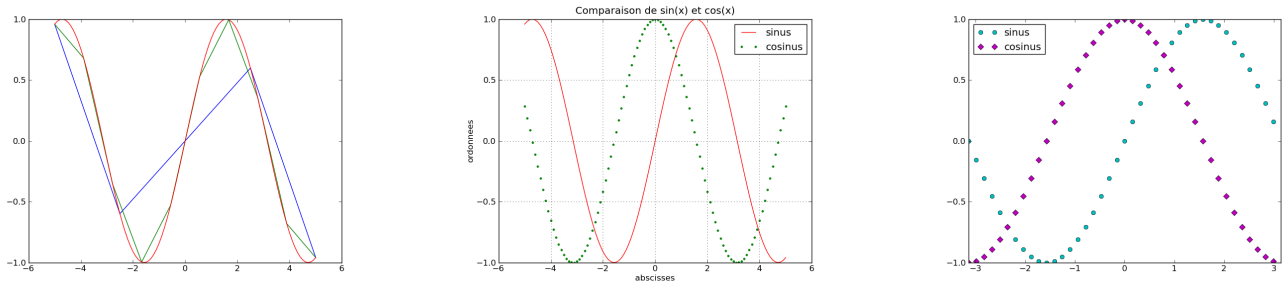
On obtient une courbe sur laquelle on peut zoomer, modifier les marge et sauvegarder dans différents formats (jpg, png, eps...). On peut même tracer plusieurs courbes sur la même figure. Par exemple, si on veut comparer les graphes de la fonction précédente en modifiant la grille de départ, on peut écrire

```
from matplotlib.pylab import *

a = linspace(-5,5,5) # a = [-5,-3,-1,1,3,5] with 6 elements
fa = sin(a)
b = linspace(-5,5,10) # a = [-5,-4,-3,...,5] with 11 elements
fb = sin(b)
c = linspace(-5,5,101) # b = [-5,-4.9,-4.8,...,5] with 101 elements
fc = sin(c)
plot(a,fa,b,fb,c,fc)
show()
```

Le résultat est affiché à la figure A.1a (la courbe bleu correspond à la grille la plus grossière, la courbe rouge correspond à la grille la plus fine).

Pour tracer plusieurs courbes, on peut les mettre les unes à la suite des autres en spécifiant la couleur et le type de trait, changer les étiquettes des axes, donner un titre, ajouter une grille, une légende...



(a) (b) (c)

FIGURE A.1.: Exemples `pylab`

`-`	solid line	`--`	dashed line
`-.`	dash-dot line	`:`	dotted line
`.`	points	`,`	pixels
`o`	circle symbols	`^`	triangle up symbols
`v`	triangle down symbols	`<`	triangle left symbols
`>`	triangle right symbols	`s`	square symbols
`+`	plus symbols	`x`	cross symbols
`D`	diamond symbols		
`b`	blue	`g`	green
`r`	red	`c`	cyan
`m`	magenta	`y`	yellow
`k`	black	`w`	white

TABLE A.1.: Quelques options de `pylab`

```
from matplotlib.pylab import *
x = linspace(-5,5,101) # x = [-5,-4.9,-4.8,...,5] with 101 elements
y1 = sin(x) # operation is broadcasted to all elements of the array
y2 = cos(x)
plot(x,y1,"r-",x,y2,"g.")
legend(['sinus','cosinus'])
xlabel('abscisses')
ylabel('ordonnees')
title('Comparaison de sin(x) et cos(x)')
grid(True)
show()
```

`"r-"` indique que la première courbe est à tracer en rouge avec un trait continu, et `"g."` que la deuxième est à tracer en vert avec des points. Le résultat est affiché à la figure A.1b. Voir la documentation de `pylab` pour connaître les autres options de ce tracé.

On peut déplacer la légende en spécifiant l'une des valeurs suivantes : `best`, `upper right`, `upper left`, `lower right`, `lower left`, `center right`, `center left`, `lower center`, `upper center`, `center` :

```
from matplotlib.pylab import *
x = arange(-pi,pi,0.05*pi)
plot(x,sin(x),'co',x,cos(x),'mD')
legend(['sinus','cosinus'],loc='upper left')
axis([-pi, pi, -1, 1]) # axis([xmin, xmax, ymin, ymax])
show()
```

Le résultat est affiché à la figure A.1c.

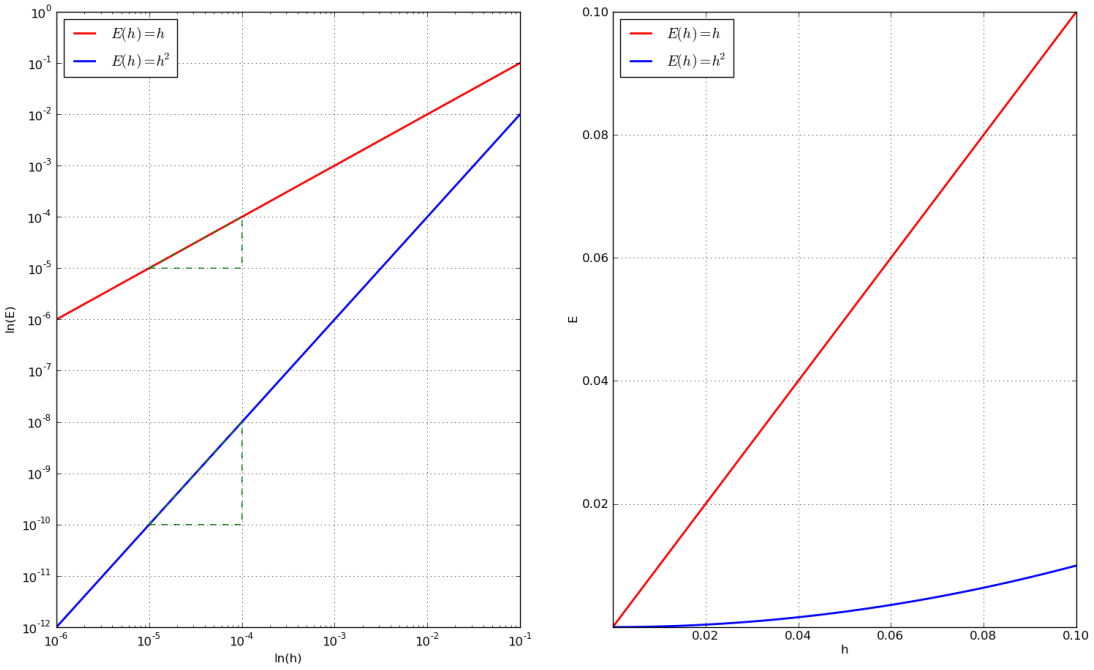
Représentations des erreurs : échelles logarithmique et semi-logarithmique

Quand on étudie les propriétés de convergence d'une méthode numérique, on trace souvent des graphes représentant

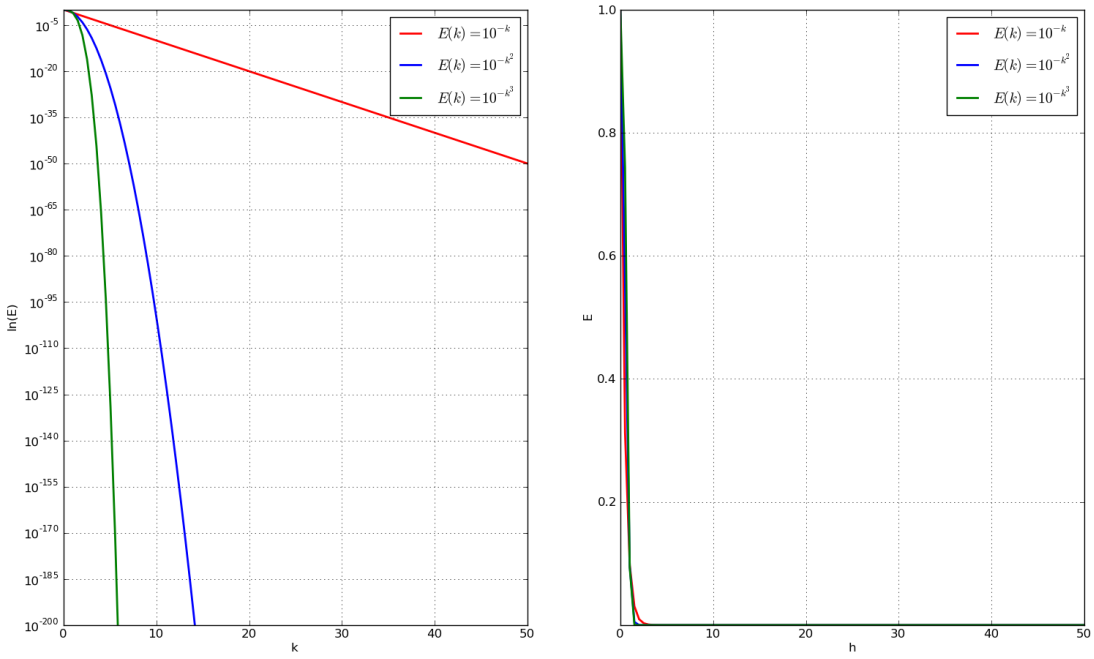
- l'erreur E en fonction de h, le pas de discrétisation (par exemple pour une formule de quadrature ou le calcul approchée de la solution d'une EDO) ;
- l'erreur E en fonction de k, le pas d'itération (par exemple pour les méthodes de recherche des zéros d'une fonction).

Pour ces graphes on a recours à des représentations en échelle logarithmique ou semi-logarithmique.

- Utiliser une échelle logarithmique signifie représenter $\ln(h)$ sur l'axe des abscisses et $\ln(E)$ sur l'axe des ordonnées. Le but de cette représentation est clair : si $E = Ch^p$ alors $\ln(E) = \ln(C) + p\ln(h)$. En échelle logarithmique, p représente donc la pente de la ligne droite $\ln(E)$. Ainsi, quand on veut comparer deux méthodes, celle présentant la pente la plus forte est celle qui a l'ordre le plus élevé (la pente est $p = 1$ pour les méthodes d'ordre un, $p = 2$ pour les méthodes d'ordre deux, et ainsi de suite). Il est très simple d'obtenir avec Python des graphes en échelle logarithmique : il suffit de taper `loglog` au lieu de `plot`. Par exemple, on a tracé sur la figure à gauche des droites représentant le comportement de l'erreur de deux méthodes différentes. La ligne rouge correspond à une méthode d'ordre un, la ligne bleu à une méthode d'ordre deux (comparer les deux triangles). Sur la figure à droite on a tracé les mêmes données qu'à gauche mais avec la commande `plot`, c'est-à-dire en échelle linéaire pour les axes x et y. Il est évident que la représentation linéaire n'est pas la mieux adaptée à ces données puisque la courbe $E(h) = h^2$ se confond dans ce cas avec l'axe des x quand $x \in [10^{-6}; 10^{-2}]$, bien que l'ordonnée correspondante varie entre $x \in [10^{-12}; 10^{-4}]$, c'est-à-dire sur 8 ordres de grandeur.



- Plutôt que l'échelle log-log, nous utiliserons parfois une échelle semi-logarithmique, c'est-à-dire logarithmique sur l'axe des y et linéaire sur l'axe des x. Cette représentation est par exemple préférable quand on trace l'erreur E d'une méthode itérative en fonction des itérations k ou plus généralement quand les ordonnées s'étendent sur un intervalle beaucoup plus grand que les abscisses. Si $E(k) = C^{k^n} E(0)$ avec $C \in]0;1[$ alors $\ln(E) = \ln(E(0)) + k^n \ln(C)$, c'est-à-dire une droite si $n = 1$, une parabole si $n = 2$ etc. La commande Python pour utiliser l'échelle semi-logarithmique est `semilogy`.
  Par exemple, on a tracé sur la figure à gauche des courbes représentant le comportement de l'erreur de trois méthodes différentes. La ligne rouge correspond à une méthode d'ordre un, la parabole bleu à une méthode d'ordre deux et la cubique verte à une méthode d'ordre trois. Sur la figure à droite on a tracé les mêmes données qu'à gauche mais avec la commande `plot`, c'est-à-dire en échelle linéaire pour les axes x et y. Il est évident que la représentation linéaire n'est pas la mieux adaptée à ces données.



A.4. Structure conditionnelle

Supposons vouloir définir la fonction valeur absolue :

$$|x| = \begin{cases} x & \text{si } x \geq 0, \\ -x & \text{sinon.} \end{cases}$$

On a besoin d'une instruction qui opère une disjonction de cas. En Python il s'agit de l'instruction de choix introduite par le mot-clé `if`. La syntaxe est la suivante :

```
if condition_1:
    instruction_1.1
    instruction_1.2
    ...
elif condition_2:
    instruction_2.1
    instruction_2.2
    ...
...
else:
    instruction_n.1
    instruction_n.2
    ...
```

où `condition_1`, `condition_2`... représentent des ensembles d'instructions dont la valeur est `True` ou `False` (on les obtient en général en utilisant les opérateurs de comparaison). La première condition `condition_i` ayant la valeur `True` entraîne l'exécution des instructions `instruction_i.1`, `instruction_i.2`... Si toutes les conditions sont fausses, les instructions `instruction_n.1`, `instruction_n.2`... sont exécutées.

⚠ **ATTENTION.** *Bien noter le rôle essentiel de l'indentation qui permet de délimiter chaque bloc d'instructions et la présence des deux points après la condition du choix et après le mot clé* `else`.

Voici un exemple pour établir si un nombre est positif :

```
def sign_of(a):
    if a < 0.0:
        sign = 'negative'
    elif a > 0.0:
```

```
        sign = 'positive'
    else:
        sign = 'zero'
    return sign

a = 2.0
print 'a is ' + sign_of(a) # Output: a is positive
a = -2.0
print 'a is ' + sign_of(a) # Output: a is negative
a = 0.0
print 'a is ' + sign_of(a) # Output: a is zero
```

La fonction valeur absolue peut être définie comme suit :

```
def val_abs(x):
    if x>0:
        return x
    else:
        return -x

val_abs(5) # Output 5
val_abs(-5) # Output 5
```

A.5. Boucles

Les structure de répétition se classent en deux catégories :

répétition conditionnelle : le bloc d'instructions est à répéter autant de fois qu'une condition est vérifiée,

répétition inconditionnelle : le bloc d'instructions est à répéter un nombre donné de fois.

Boucle `while` : répétition conditionnelle Le constructeur `while` a la forme générale suivante (attention à l'indentation et aux deux points) :

```
while condition:
    instruction_1
    instruction_2
    ...
```

où `condition` représente des ensembles d'instructions dont la valeur est `True` ou `False`. Tant que la condition `condition_i` a la valeur `True`, on exécute les instructions `instruction_i`.

⚠ **ATTENTION.** *Si la condition ne devient jamais fausse, le bloc d'instructions est répété indéfiniment et le programme ne se termine pas.*

Voici un exemple pour créer la liste $\left[1, \frac{1}{2}, \frac{1}{3}, \ldots\right]$:

```
nMax = 5
n = 1
a = [] # Create empty list
while n<nMax:
    a.append(1.0/n) # Append element to list
    n += 1
print a # Output [1.0, 0.5, 0.33333333333333331, 0.25]
```

Dans l'exemple suivant on calcul la somme des n premiers entiers :

```
def somme(n):
    s ,i = 0, 0
    while i<n:
        i += 1
        s += i
    return s
somme(100) # Output 5050
```

Boucle `for` : répétition inconditionnelle Lorsque l'on souhaite répéter un bloc d'instructions un nombre déterminé de fois, on peut utiliser un *compteur actif*, c'est-à-dire une variable qui compte le nombre de répétitions et conditionne la sortie de la boucle. C'est la structure introduite par le mot-clé `for` qui a la forme générale suivante (attention à l'indentation et aux deux points) :

```
for target in sequence:
⟶instruction_1
⟶instruction_2
⟶...
```

où `target` est le *compteur actif* et `sequence` est un itérateur (souvent généré par la fonction `range` ou une liste ou une chaîne de caractères). Tant que `target` appartient à `sequence`, on exécute les instructions `instruction_i`.

Voici un exemple pour créer la liste $[1, \frac{1}{2}, \frac{1}{3}, \dots]$ avec un itérateur généré par la fonction `range` :

```
nMax = 5
a = [] # Create empty list
for n in range(1,nMax):
⟶a.append(1.0/n) # Append element to list
print a # The output is [1.0, 0.5, 0.33333333333333331, 0.25]
```

Interrompre une boucle L'instruction `break` sort de la plus petite boucle `for` ou `while` englobante. L'instruction `continue` continue sur la prochaine itération de la boucle. Les instructions de boucle ont une clause `else` qui est exécutée lorsque la boucle se termine par épuisement de la liste (avec `for`) ou quand la condition devient fausse (avec `while`), mais pas quand la boucle est interrompue par une instruction `break`. Ceci est expliqué dans la boucle suivante, qui recherche des nombres premiers :

```
for n in range(2,10):
⟶for x in range(2,n):
⟶⟶if n%x==0:
⟶⟶⟶print n, 'egale', x, '*', n/x
⟶⟶⟶break
⟶else:
⟶⟶print n, 'est un nombre premier'
```

ce qui donne

```
2 est un nombre premier
3 est un nombre premier
4 egale 2 * 2
5 est un nombre premier
6 egale 2 * 3
7 est un nombre premier
8 egale 2 * 4
9 egale 3 * 3
```

List-comprehensions Les listes définies par compréhension permettent de générer des listes de manière très concise sans avoir à utiliser des boucles. La syntaxe pour définir une liste par compréhension est très proche de celle utilisée en mathématiques pour définir un ensemble :

$$\{\ f(x)\ \mid\ x\ \in\ E\ \}$$

```
[ f(x) for x in E ]
```

Voici quelques exemples :

```
liste = [2, 4, 6, 8, 10]
[3*x for x in liste] # Output [6, 12, 18, 24, 30]
[[x,x**3] for x in liste] # Output [[2, 8], [4, 64], [6, 216], [8, 512], [10, 1000]]
[3*x for x in liste if x>5] # Output [18, 24, 30]
[3*x for x in liste if x**2<50] # Output [6, 12, 18]
liste2 = range(3)
[x*y for x in liste for y in liste2] # Output [0, 2, 4, 0, 4, 8, 0, 6, 12, 0, 8, 16, 0, 10, 20]
```

Conversion des dégrées CELSIUS en dégrées Kelvin :

```
Cdegrees = range(0,101,5);
Fdegrees = [(9./5)*C+32 for C in Cdegrees]
for i in range(len(Cdegrees)):
⟶print '%5d %5.1f' % (Cdegrees[i], Fdegrees[i])
```

ce qui donne

```
    0 32.0
    5 41.0
   10 50.0
   15 59.0
   20 68.0
   25 77.0
   30 86.0
   35 95.0
   40 104.0
   45 113.0
   50 122.0
   55 131.0
   60 140.0
   65 149.0
   70 158.0
   75 167.0
   80 176.0
   85 185.0
   90 194.0
   95 203.0
  100 212.0
```

On construit la liste des années bissextiles entre l'année 2000 et l'année 2099 :

```
>>> [b for b in range(2000,2100) if (b%4==0 and b%100!=0) or (b%400==0)]
[2000, 2004, 2008, 2012, 2016, 2020, 2024, 2028, 2032, 2036, 2040, 2044, 2048, 2052, 2056, 2060, 2064,
    ➥2068, 2072, 2076, 2080, 2084, 2088, 2092, 2096]
```

On construit la liste des diviseurs d'un entier $n \in \mathbb{N}$:

```
>>> n = 100
>>> [d for d in range(1,n+1) if (n%d==0)]
[1, 2, 4, 5, 10, 20, 25, 50, 100]
```