Calcul formet et numérique en ingénierie : Résumé du cours

Sylvain Julmy

24 janvier 2016

Table des matières

1. Banach TODO

Banach est un résultant théorique, en pratique c'est compliqué de trouvé un bon D.
Matrice de Jacobi : On dérive F et G par rapport à x et y, on obtient donc une matrice 2x2
Norme matriciel : racine de la somme des éléments au carré

$$\sqrt{\sum M_{i,j}}$$

F est une contraction de D si

$$max_{(x,y \in D)} ||J(x,y)|| < 1$$

On cherche donc le maximum de l'expression $||\mathbf{J}(x,y)||$

1.1 Exemple

On à

$$||\mathbf{J}(x,y)|| = \frac{1}{10} (\begin{matrix} 2x & 2y \\ y^2+1 & 2xy \end{matrix})$$

on obtient donc la norme matricielle

$$||\mathbf{J}(x,y)|| = \frac{1}{10}\sqrt{4x^2 + 4y^2 + (y^2+1) + 4x^2y^2}$$

Le vecteur d'erreur est donné par

$$e_k = x_k - r = (\begin{matrix} x_k - r \\ y_k - s \end{matrix}) = (\begin{smallmatrix} \varepsilon_k \\ \delta_k \end{smallmatrix})$$

Si $\mathbf{J}(r,s)$ est nulle : convergence quadratique, sinon linéaire.

2. Courbes de Bézier

2.1 Algorithme de Casteljo

On donne trois point b_0,b_1 et b_2 et on calcule les interpolations linéaires :

$$b_0^1 = (1-t)\cdot b_0 + t\cdot b_1$$
$$b_1^1 = (1-t)\cdot b_1 + t\cdot b_2$$
$$b_0^2 = (1-t)\cdot b_0^1 + t\cdot b_1^1$$

Et on simplifie b_0^2 avec b_0^1 et b_1^1 :

$$b_0^2 = (1-t^2)b_0 + 2t(1-t)b_1 + t^2 b_2$$

On peut représenter cela sous une forme d'une matrice de vecteur triangulaire : tout ce qui se trouve au dessus de la diagonale est ignoré :

$$\begin{matrix} [b_0^0] & & \\ [b_1^0] & [b_0^1] & \\ [b_2^0] & [b_1^1] & [b_0^2] \end{matrix}$$

Où les éléments de la première colonne sont donnés et que le calcul de b_n^m est donné par

$$b_n^m = (1-t)\cdot b_n^{m-1} + t\cdot b_{n+1}^{m-1}$$

avec t donné.

Exemple

$b_0 = \begin{bmatrix}0\\0\end{bmatrix}$,$b_1 = \begin{bmatrix}0\\2\end{bmatrix}$,$b_2 = \begin{bmatrix}8\\2\end{bmatrix}$,$b_3 = \begin{bmatrix}4\\0\end{bmatrix}$: calculé b_0^3 avec $t = 0.5$.

Solution

$$\begin{matrix} \begin{bmatrix}0\\0\end{bmatrix} & & & \\ \begin{bmatrix}0\\2\end{bmatrix} & \begin{bmatrix}0\\1\end{bmatrix} & & \\ \begin{bmatrix}8\\2\end{bmatrix} & \begin{bmatrix}4\\2\end{bmatrix} & \begin{bmatrix}2.0\\1.5\end{bmatrix} & \\ \begin{bmatrix}4\\0\end{bmatrix} & \begin{bmatrix}6\\1\end{bmatrix} & \begin{bmatrix}5.0\\1.5\end{bmatrix} & \begin{bmatrix}3.5\\1.5\end{bmatrix} \end{matrix}$$

2.2 Les polynômes de Bernstein

Les polynômes de Bernstein sont donnés par la formule suivante :

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i} \text{ , pour } i = 0, 1, .., n$$

n	$i = 0$	$i = 1$	$i = 2$	$i = 3$
0	$B_0^0 = 1$			
1	$B_0^1 = 1 - t$	$B_0^1 = t$		
2	$B_0^2 = (1-t)^2$	$B_0^2 = 2t(1-t)$	$B_0^2 = t^2$	
3	$B_0^3 = (1-t)^3$	$B_1^3 = 3t(1-t)^2$	$B_1^3 = 3t^2(1-t)$	$B_3^3 = t^3$

2.3 Courbe de Bézier et polynômes de Bernstein

En prenant 3 points : $b_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, b_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, b_2 = \begin{bmatrix} 2 \\ 0.5 \end{bmatrix}$, on veut donner la paramétrisation de la courbe de Bézier associée à ces points de contrôles en utilisant les polynômes de Bernstein :

$$\begin{aligned}
\vec{x}(t) &= \vec{b_0^2}(t) \\
&= B_0^2(t)\vec{b_0} + B_1^2(t)\vec{b_1} + B_2^2(t)\vec{b_2} \\
&= (1-t)^2 \begin{bmatrix} 0 \\ 0 \end{bmatrix} + 2 \cdot t \cdot (1-t) \begin{bmatrix} 1 \\ 2 \end{bmatrix} + t^2 \begin{bmatrix} 2 \\ 0.5 \end{bmatrix}
\end{aligned}$$

ou bien avec 4 points :

$$x(t) = b_0 B_0^3(t) + b_1 B_1^3(t) + b_2 B_2^3(t) + b_3 B_3^3(t)$$

Exemple

Donner la paramétrisation de la courbe de Bézier associée aux points de contrôle :

$$b_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \text{ , } b_0 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \text{ , } b_0 = \begin{bmatrix} 2 \\ 0.5 \end{bmatrix}$$

Solution

$$\begin{aligned}
x(t) = b_0^2(t) &= \sum_{j=0}^{n} b_{i+j} B_j^r(t) \\
&= b_0 B_0^2(t) + b_1 B_1^2(t) + b_2 B_2^2(t) \\
&= \begin{bmatrix} 0 \\ 0 \end{bmatrix} \cdot (1-t)^2 + \begin{bmatrix} 1 \\ 2 \end{bmatrix} \cdot 2t(1-t) + \begin{bmatrix} 2 \\ 0.5 \end{bmatrix} \cdot t^2
\end{aligned}$$

2.4 Exercice 1

2.4.1 a

$$
\begin{matrix}
\begin{bmatrix}0.0\\0.0\end{bmatrix} & & & \\
\begin{bmatrix}1.0\\2.0\end{bmatrix} & \begin{bmatrix}0.5\\1.0\end{bmatrix} & & \\
\begin{bmatrix}3.0\\2.0\end{bmatrix} & \begin{bmatrix}2.0\\2.0\end{bmatrix} & \begin{bmatrix}1.25\\1.50\end{bmatrix} & \\
\begin{bmatrix}3.0\\0.0\end{bmatrix} & \begin{bmatrix}3.0\\1.0\end{bmatrix} & \begin{bmatrix}2.50\\1.50\end{bmatrix} & \begin{bmatrix}1.875\\1.5\end{bmatrix}
\end{matrix}
$$

2.4.2 b

voir corrigé du cours

2.4.3 c

$$
\begin{aligned}
\vec{x}(t) = \vec{b}_0^3(t) &= B_0^3(b)\vec{b}_0 + B_1^3(b)\vec{b}_1 + B_2^3(b)\vec{b}_2 \\
&= (1-t)^2 \begin{bmatrix}0\\0\end{bmatrix} + 3t(1-t) \begin{bmatrix}1\\2\end{bmatrix} + 3t^2(1-t) \begin{bmatrix}3\\2\end{bmatrix} + t^3 \begin{bmatrix}3\\0\end{bmatrix}
\end{aligned}
$$

2.5 Exercice 2

(a)

i	0	1	2	3
x_i	0.0	1.5	3.0	3.0
y_i	1.0	1.0	0.6	0.0

On souhaite que les deux points suivent la direction de la tangente,

(b)

$$
\begin{matrix}
\begin{bmatrix}0.0\\1.0\end{bmatrix} & & & \\
\begin{bmatrix}1.5\\1.0\end{bmatrix} & \begin{bmatrix}1.125\\1.000\end{bmatrix} & & \\
\begin{bmatrix}3.0\\0.6\end{bmatrix} & \begin{bmatrix}2.625\\0.700\end{bmatrix} & \begin{bmatrix}2.25000\\0.77500\end{bmatrix} & \\
\begin{bmatrix}3.0\\0.0\end{bmatrix} & \begin{bmatrix}3.000\\0.150\end{bmatrix} & \begin{bmatrix}2.90625\\2.90625\end{bmatrix} & \begin{bmatrix}2.742\\0.409\end{bmatrix}
\end{matrix}
$$

(c)

$$
\begin{aligned}
x(t) &= b_0 B_0^3(t) + b_1 B_1^3(t) + b_2 B_2^3(t) + b_3 B_3^3(t) \\
&= \begin{bmatrix} 0.0 \\ 1.0 \end{bmatrix} \cdot (1-t)^3 + \begin{bmatrix} 1.5 \\ 1.0 \end{bmatrix} \cdot 3t(1-t)^2 + \begin{bmatrix} 3.0 \\ 0.6 \end{bmatrix} \cdot 3t^2(1-t) + \begin{bmatrix} 3.0 \\ 0.0 \end{bmatrix} \cdot t^3
\end{aligned}
$$

2.6 Exercice 3

2.7 Exercice 4

2.8 Exercice 5

3. Systèmes linéaires

On cherche à résoudre $Ax = b$ avec A une matrice (m, n), $b \in \mathbb{R}^m$ et $x \in \mathbb{R}^n$, on suppose que x existe et on utilise des matrices de taille (n, n).
Opérations autorisées :
- Multiplier une équation par une constante
- Ajouter une équation à une autre

Exemple

$$\begin{matrix} 10x_1 & -7x_2 & 0 & = 7 \\ -3x_1 & 2x_2 & 6x_3 & = 4 \\ 5x_1 & -x_2 & 5x_3 & = 6 \end{matrix}$$

On peut l'écrire

$$\begin{pmatrix} 10 & -7 & 0 \\ -3 & 2 & 6 \\ 5 & -1 & 5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 7 \\ 4 \\ 6 \end{pmatrix} \iff \left(\begin{array}{ccc|c} 10 & -7 & 0 & 7 \\ -3 & 2 & 6 & 4 \\ 5 & -1 & 5 & 6 \end{array}\right)$$

On va travailler uniquement avec la matrice A

3.1 Décomposition de la matrice

a_{11} est le pivot du premier pas, on calcule les multiplicateurs l_{21} et l_{31} en effectuant :

$$l_{21} = \frac{a_{21}}{a_{11}}$$
$$l_{31} = \frac{a_{31}}{a_{11}}$$

On change ensuite chaque ligne de A avec les deux multiplicateurs :

$$\begin{pmatrix} L_1 = L_1 \\ L_2 = L_2 - l_{21} \cdot L_1 \\ L_3 = L_3 - l_{31} \cdot L_1 \end{pmatrix}$$

et pour le deuxième pas, le pivot est a_{22} on calcule donc $l_{32} = \frac{a_{32}}{a_{22}}$ et on effectue la nouvelle matrice avec le multiplicateur :

$$\begin{pmatrix} L_1 = L_1 \\ L_2 = L_2 - l_{21} \cdot L_1 \\ L_3 = L_3 - l_{31} \cdot L_1 = (L_3 - l_{31} \cdot L_1) - l_{32} L_2 \end{pmatrix}$$

On peut exprimer A sous la forme $A = LU$, en reprenant A de l'exemple précédent :

$$L = \begin{pmatrix} 1 & 0 & 0 \\ -0.3 & 1 & 0 \\ 0.5 & -25 & 1 \end{pmatrix}, U = \begin{pmatrix} 10 & -7 & 0 \\ 0 & -0.1 & 6 \\ 0 & 0 & 155 \end{pmatrix}$$

L est la matrice des multiplicateurs et U la matrice des coefficients finaux.

3.2 Résolution de $Ax = b$

- $Ax = b \longleftrightarrow LUx = b$
- $y = Ux$
- Résoudre $Ly = b$: on trouve y par substitution avant
- Résoudre $Ux = y$: on trouve x par substitution arrière

3.2.1 Exercice

Résoudre le système

$$\begin{matrix} 10x_1 & -7x_2 & 0 & = 7 \\ -3x_1 & 2.099x_2 & 6x_3 & = 3.901 \\ 5x_1 & -x_2 & 5x_3 & = 6 \end{matrix}$$

On pose

$$A = \begin{pmatrix} 10 & -7 & 0 \\ -3 & 2.099 & 6 \\ 5 & -1 & 5 \end{pmatrix}$$

on calcule

$$l_{21} = \frac{a_{21}}{a_{11}} = \frac{-3}{10} = -0.3$$
$$l_{31} = \frac{a_{31}}{a_{11}} = \frac{5}{10} = 0.5$$

ce qui nous donne

$$A' = \begin{pmatrix} 10 & -7 & 0 \\ 0 & -0.0010 & 6.0000 \\ 0 & 2.5000 & 5.0000 \end{pmatrix}$$

on calcule

$$l_{32} = \frac{a_{32}}{a_{22}} = \frac{2.5}{-0.0010} = -2500$$

ce qui donne

$$U = \begin{pmatrix} 10 & -7 & 0 \\ 0 & -0.0010 & 6.0000 \\ 0 & 0 & 1.5005 \end{pmatrix}$$

$$L = \begin{pmatrix} 1 & 0 & 0 \\ -0.3 & 1 & 0 \\ 0.5 & -2500 & 1 \end{pmatrix}$$

$$b = \begin{pmatrix} 7 \\ 3.901 \\ 6 \end{pmatrix}$$

et on résout $Ly = b$ et $Ux = y$:

$$\begin{aligned} & y_1 = b_1 = 7 \\ & y_2 = b_2 - -0.3 \cdot y_1 = 6.001 \\ & y_3 = b_3 - 0.5 \cdot y_1 - -2500 \cdot y_2 = 15005 \\ & \rightarrow \\ & y_1 = b_1 \\ & y_2 = b_2 - L_{12} \cdot b_1 \\ & y_3 = b_3 - L_{13} \cdot b_1 - L_{23} \cdot b_2 \end{aligned}$$

4. Résidus et erreurs

On veut résoudre le système $Ax = b$ en utilisant la factorisation $PA = LU$ sur :

$$A = \begin{pmatrix} 1 & 2 \\ -3 & \pi \end{pmatrix} \qquad b = \begin{pmatrix} 1 \\ \frac{2}{3} \end{pmatrix}$$

Il y aura des erreurs :
- — représentation machine de A
- — représentation machine de b
- — résolution de $PA = LU$

4.1 Norme vectorielle

La fonction $||\cdot|| : \mathbb{R}^n \to \mathbb{R}$ est appelée norme vectorielle si $\forall x, y \in \mathbb{R} \wedge \alpha \in \mathbb{R}$ les propriétés suivantes sont respectés :
- — $||x|| \geq 0$
- — $||x|| = 0 \iff x = 0$
- — $||\alpha x|| = |\alpha| \cdot ||x||$
- — $||x + y|| \geq ||x|| + ||y||$

4.2 Norme matricielle

Soit la fonction $|| \cdot ||_v$ donnée comme la norme vectorielle, une norme matricielle est une fonction $|| \cdot || : \mathbb{R}^{n\times n} \to \mathbb{R}, \forall A, B \in \mathbb{R}^{n\times n}, x \in \mathbb{R}^n et \alpha \in \mathbb{R}$ satisfait à :
- — $||A \geq 0||$
- — $||A|| = 0 \iff A = 0$
- — $||\alpha A|| = |\alpha|||A||$
- — $||A + B|| \geq ||A|| + ||B||$
- — $||AB|| \geq ||A|| \cdot ||B||$
- — $||Ax||_v \geq ||A|| \cdot ||x||_v$

Comment calculer la norme matricielle ? Propositions :
- — $\sqrt{\sum_{i=0}^{n} A_i}$, on voit la matrice comme un vecteur
- — Un préfère des normes matricielles induites par des vecteurs : $max(\frac{||A\vec{x}||}{||\vec{x}||})$ c'est incalculable en pratique sauf pour $p \in 1, 2, \infty$

4.3 Perturbation sur b

x est la solution exacte de $Ax = b$, δb est la petite perturbation sur b. $x + \delta x_b$ est la solution exacte du système perturbé

$$A(x + \delta x_b) = b + \delta b$$

On obtient donc

$$\frac{||\delta x_b||}{||x||} \geq ||A|| \cdot ||A^{-1}|| \cdot ||\frac{\delta b}{b}||$$

Conclusion : un petit $||\delta b||/||b|| \Rightarrow$ un petit $||\delta x_b||/||x||$ seulement si $||A|| \cdot ||A^{-1}||$ n'est pas trop grand. C'est la même chose pour une petite perturbation sur A. Donc on obtient la solution exacte du système par $(A + \delta A)(x + \delta x) = b + \delta b$:

$$\frac{||\delta x||}{||x + \delta x||} \geq \frac{||A|| \cdot ||A^{-1}||}{1 - ||A|| \cdot ||A^{-1}|| \cdot \frac{||\delta A||}{||A||}} \cdot (\frac{||\delta A||}{||A||} + \frac{||\delta b||}{||b||})$$

Conclusion : un petit $||\delta b||/||b||$ et un petit $||\delta A||/||A|| \Rightarrow$ un petit $||\delta x_b||/||x||$ seulement si $||A|| \cdot ||A^{-1}||$ n'est pas trop grand. Si $||A|| \cdot ||A^{-1}|| >> 1$ alors on dit que la matrice A est mal conditionnée.x

4.4 Calcule des normes vectoriels

La norme $||x||_1$ se calcule en prenant $max(|x|)$.
La norme $||x||_2$ se calcule en prenant $\sum_{i=1}^{n} |x_i|$.
La norme $||x||_\infty$ se calcule en prenant $\sqrt{\sum_{i=1}^{n} x_i^2}$

4.5 Calcule des normes matriciels

La norme $||A||_1$ se calcule en prenant

4.6 Exercices

(1)

$$\begin{array}{ccc} ||a||_1 = 4 & ||a||_2 = 9 & ||a||_\infty = \sqrt{29} \\ ||b||_1 = 7 & ||b||_2 = 16 & ||b||_\infty = \sqrt{90} \\ ||c||_1 = 4 & ||c||_2 = 10 & ||c||_\infty = \sqrt{30} \end{array}$$

(2)

5. Moindres carrés

Idée : on dispose d'un nuage de point dans un graphe et o cherche une fonction linéaire qui permet de représenter approximativement ces points. On dispose donc d'un modèle linéaire $y = at + b$.
On minimise la somme des carrés résidus :

$$f(a,b) = \sum_{i=1}^{n}(y_i - (at_i + b))^2 = min$$

$$\frac{\partial f}{\partial a} = 0 \text{ et } \frac{\partial f}{\partial b} = 0$$

On obtient le système :

$$\begin{pmatrix} \sum_{i=1}^m t_i^2 & \sum_{i=1}^m t_i \\ \sum_{i=1}^m t_i & m \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^m y_i t_i \\ \sum_{i=1}^m y_i \end{pmatrix}$$

5.1 Généralisation

Mesures : $(t_1, y_1), ..., (t_m, y_m)$
Modèles : $y(t) = \sum_{i=1}^n \beta_i \varnothing_i(t)$ avec $n < m$
On obtient

$$\begin{cases} y_1 \approx \beta_1 \varnothing_1(t_1) + ... + \beta_n \varnothing_n(t_1) \\ ... \\ y_m \approx \beta_1 \varnothing_1(t_m) + ... + \beta_n \varnothing_n(t_m) \end{cases}$$

Forme matricielle : $y \approx Ax$ avec

$$A = \begin{pmatrix} \varnothing_1(t_1) & \varnothing_2(t_1) & ... & \varnothing_n(t_1) \\ \varnothing_1(t_2) & \varnothing_2(t_2) & ... & \varnothing_n(t_2) \\ ... & ... & ... & ... \\ \varnothing_1(t_m) & \varnothing_2(t_m) & ... & \varnothing_n(t_m) \end{pmatrix}, x = \begin{pmatrix} \beta_1 \\ \beta_2 \\ ... \\ \beta_n \end{pmatrix}, y = \begin{pmatrix} y_1 \\ y_2 \\ ... \\ y_m \end{pmatrix}$$

$$A \in \mathbb{R}^{m \times n}, x \in \mathbb{R}^n, y \in \mathbb{R}^m, n \leq m$$

5.2 Solution

On cherche x tel que $Ax \approx y$, on définit $r = y - Ax$. $||r||_2^2$ est minimal $\iff (A^T A)x = A^T y$. Et on obtient $\kappa_2(A^T A) \approx [\kappa_2(A)]^2$.

5.3 Exemple : population des USA

$$A = \begin{pmatrix} 1950 & 1 \\ 1960 & 1 \\ 1970 & 1 \\ 1980 & 1 \\ 1990 & 1 \\ 2000 & 1 \end{pmatrix}, \; y = \begin{pmatrix} 150.697 \\ 179.323 \\ 203.212 \\ 226.505 \\ 249.633 \\ 281.422 \end{pmatrix}$$

$$A^T A = \begin{pmatrix} 23405500 & 11850 \\ 11850 & 6 \end{pmatrix}$$

$$A^T y = \begin{pmatrix} 2553753.44 \\ 1290.792 \end{pmatrix}$$

Maintenant il suffit de résoudre $A^T A \cdot \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} = A^T y$ et on trouve

$$\beta_1 = 2.53671 \text{ et } \beta_2 = -4794.86743$$

5.4 Matlab : population des USA

```
A = [1950,1;1960,1;1970,1;1980,1;1990,1;2000,1];
y = [150.697;179.323;203.212;226.505;249.633;281.422];

At = A';
AtA = At * A;
Aty = At * y;

beta = [2.5367;-4.7949e3];

printmat(beta);
printmat(AtA \ Aty);
```

5.5 Autre modèle

$$y(t) = Ke^{\lambda t}$$

On utilise une transformation logarithmique :

$$log(y) \approx log(K) + \lambda t = \beta_2 + \beta_1 t$$

avec

$$\beta_2 = log(K) et \beta_1 = \lambda \longrightarrow K = e^{\beta_2}, \; \lambda = \beta_1$$

5.6 Trouver les polynômes

Afin de trouver des polynômes de régression de degrés n, on commence par construire notre matrice A de hauteur $m = n - 1$ (nombre de point) comme suit, les points sont données par le vecteur $\mathbf{x} = \{x_0, x_1, x_2, ..., x_m\}$ et le vecteur $\mathbf{y} = \{y_0, y_1, y_2, ..., y_m\}$:

$$A = \begin{pmatrix} x_0^n & x_0^{n-1} & \cdots & x_0^1 & x_0^0 \\ x_1^n & x_1^{n-1} & \cdots & x_1^1 & x_1^0 \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ x_{m-1}^n & x_{m-1}^{n-1} & \cdots & x_{m-1}^1 & x_{m-1}^0 \\ x_m^n & x_m^{n-1} & \cdots & x_m^1 & x_m^0 \end{pmatrix}, \; y = \mathbf{y}$$

puis on résous le système $A^T A \cdot \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{pmatrix} = A^T y$

et pour finir on trouve notre polynôme $f(t) = \sum_{i=0}^{m} \beta_i \cdot t^i$

5.7 Exercices

7(a) : le polynôme trouvé vaut $-40.02t + 10.49$ la somme des moindres carrés $\sum_i r_i^2 = 4865.64$

7(b) : la valeur du polynôme en $x = 2.5$ vaut : $f(x) = -40.02t + 10.49$ $f(2.5) = -89.56$

7(c - 2) : le polynôme de degrés 2 vaut $f(t) = 32.93 + t^2 - 7.09t + -22.44$, la somme des carrés des résidus vaut 529.42 et la valeur du polynôme en $x = 2.5$ vaut : $f(x) = 32.93 + x^2 - 7.09x - 22.44$ $f(2.5) = 165.6088$.

7(c - 3) : le polynôme de degrés 3 vaut $f(t) = -17.15t^3 + 7.20t^2 + 15.20t + -7.00$, la somme des carrés des résidus vaut 0 et la valeur du polynôme en $x = 2.5$ vaut : $f(x) = -17.15x^3 + 7.20x^2 + 15.20x - 7.00$ $f(2.5) = -191.56$.

7(d) : Non, si on désire obtenir un polynôme de degré 4, il faut absolument avec $n + 1$ points différents, où n est le degrés du polynôme recherché.

6. Méthode QR

Buts : Transformer une matrice A (quelconque) en une matrice triangulaire supérieur R. Où la norme des colonnes est préservée.

$$A = \begin{pmatrix} \star & \star & \star & \star \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ \star & \star & \star & \star \end{pmatrix} \to \cdots \to \begin{pmatrix} \star & \star & \star & \star \\ 0 & \star & \star & \star \\ 0 & 0 & \star & \star \\ 0 & 0 & 0 & \star \end{pmatrix} = R$$

Solution : $A = QR$, Q orthogonal : $Q^{-1} = Q^T$ avec $Q^T = H_n H_{n-1} ... H_2 H_1$. H_i : matrice de Householder (orthogonal, une par colonne de A).

6.1 Matrice de Householder

On veut obtenir H de telle sorte que

$$x = \begin{pmatrix} \star \\ \star \\ \star \end{pmatrix} \to Hx = \begin{pmatrix} -\rho \\ 0 \\ 0 \end{pmatrix}$$

avec $||x||_2 = ||Hx||_2$: la norme est préservé.

Algorithme : on choisit ρ, v et γ comme

$$\rho = sign(x_1)||x||_2, \quad v = x + \rho e_1, \quad \gamma = \frac{||v||_2^2}{2} = \rho v_1$$

e_n est le vecteur unitaire en n et on résous

$$H = \begin{cases} I, & \text{si } v = 0 \\ I - \frac{vv^T}{\gamma} & \text{si } v \neq 0 \end{cases}$$

Exemple : $x = (-2, 2, 1)^T$

$$\begin{aligned} \rho &= -\sqrt{(-2)^2 + (2)^2 + (1)^2} = -3 \\ v &= (-2, 2, 1) + (-3, 0, 0) = (-5, 2, 1) \\ \gamma &= \frac{(-5)^2 + 2^2 + 1^2}{2} = \frac{30}{2} = 15 \end{aligned}$$

et on calcule H

$$\begin{aligned} H = I - \frac{vv^T}{\gamma} &= \frac{(-5,2,1) \cdot (-5,2,1)^T}{\gamma} = I - \frac{1}{15} \cdot \begin{pmatrix} 25 & -10 & -5 \\ -10 & 15 & 2 \\ -5 & 2 & 1 \end{pmatrix} \\ &= \begin{pmatrix} -\frac{2}{3} & \frac{2}{3} & \frac{1}{3} \\ \frac{2}{3} & \frac{11}{15} & -\frac{2}{15} \\ \frac{1}{3} & -\frac{2}{15} & \frac{14}{15} \end{pmatrix} \end{aligned}$$

Calcul de Ha : on veut calculer H mais pour le vecteur a : $\tau = \frac{v^T a}{\gamma}$ et on obtient $Ha = a - \tau v$

Exemple : $a = (4,0,2)^T, \rho = -3, v = (-5,2,1)^T, \gamma = 15$ ici $\tau = -\frac{18}{15}$ et on trouve $Ha = (-2, \frac{12}{5}, \frac{16}{5})^T$

Calcule de HA pour une matrice A :

$$A = \begin{pmatrix} 2 & 4 & 2 \\ -1 & 0 & -4 \\ 2 & 2 & -1 \end{pmatrix} = (a_1|a_2|a_3)$$

Exemple : on reprend les algorithmes précédents et on calcule :

$$\rho_1 = 3, v_1 = (5,-1,2)^T, \gamma_1 = 15$$

puis on peut obtenir les éléments suivants :

$$\tau_2 = \frac{v_1^T a_2}{\gamma_1} = \frac{8}{5}, \quad \tau_3 = \frac{v_1^T a_3}{\gamma_1} = \frac{4}{5}$$

et donc

$$Ha_2 = a_2 - \tau_2 v_1 = (-4, \frac{8}{5}, -\frac{6}{5})^T$$
$$Ha_3 = a_3 - \tau_3 v_1 = (-2, -\frac{16}{5}, -\frac{13}{5})^T$$

et obtient HA :

$$\begin{pmatrix} -3 & -4 & -2 \\ 0 & \frac{8}{5} & -\frac{16}{5} \\ 0 & -\frac{6}{5} & -\frac{13}{5} \end{pmatrix}$$

6.2 Méthode QR 1

Exemple :

$$A = \begin{pmatrix} 2 & 4 & 2 \\ -1 & 0 & -4 \\ 2 & 2 & -1 \end{pmatrix} = (a_1|a_2|a_3)$$

On a $H_1 A$ suivante :

$$H_1 A = \begin{pmatrix} -3 & -4 & -2 \\ 0 & \frac{8}{5} & -\frac{16}{5} \\ 0 & -\frac{6}{5} & -\frac{13}{5} \end{pmatrix}$$

et on travail sur B :

$$B = \begin{pmatrix} \frac{8}{5} & -\frac{16}{5} \\ -\frac{6}{5} & -\frac{13}{5} \end{pmatrix} = (b_2|b_3)$$

on cherche à obtenir $\widetilde{H}_2 B$:

$$\rho_2 = 2, v_2 = (\frac{18}{5}, -\frac{6}{5})^T, \gamma_2 = \frac{36}{5}$$
$$\tau_4 = \frac{v_2^T b_3}{\gamma} = -\frac{7}{6}$$
$$\widetilde{H}_2 b_3 = b_3 - \tau_4 v_2 = (1,4)^T$$
$$H_2 = \left(\begin{array}{c|c} 1 & 0 \\ \hline 0 & \widetilde{H}_2 \end{array}\right)$$

et on obtient :

$$\begin{pmatrix} -3 & -4 & -2 \\ 0 & -2 & 1 \\ 0 & 0 & -4 \end{pmatrix}$$

6.3 Résolution de Ax=b

si A est carré : $A = QR$ et $Q^{-1} = Q^T$ donc $Ax = b \leftrightarrow QRx = b \leftrightarrow Rx = Q^T b$

si A est rectangulaire : $A = QR$ et $Q^{-1} = Q^T$, $A^T Ax = A^T b$ donc $A^T Ax = A^T b \leftrightarrow Rx = Q^T b$
Dans les deux cas, on ne construit pas Q. On applique l'algorithme de calcule de HA pour une matrice A.

6.4 Itération de Jacobi

On veut résoudre $Ax = b$ par itération :

$$x_i^{k+1} = \frac{1}{a_{ii}}\Big(b_i - \sum_{j=1, j\neq i}^{n} a_{ij} x_j^{(k)}\Big) \ ; 1 \leq i \leq n$$

Sous forme matricielle :

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} D = \begin{pmatrix} a_{11} & 0 & 0 \\ 0 & a_{22} & 0 \\ 0 & 0 & a_{33} \end{pmatrix}$$

$$L = -\begin{pmatrix} 0 & 0 & 0 \\ a_{21} & 0 & 0 \\ a_{31} & a_{32} & 0 \end{pmatrix} U = -\begin{pmatrix} 0 & a_{12} & a_{13} \\ 0 & 0 & a_{23} \\ 0 & 0 & 0 \end{pmatrix}$$

et

$$A = D - L - U$$

$$x^{k+1} = D^{-1}b + D^{-1}(L + U)x^k$$

6.5 Exercices

(1) : $x = (4,4,2)^T$

$$
\begin{aligned}
&\rho = +\sqrt{4^2+4^2+2^2} = \sqrt{36} = 6\\
&v = x + (6,0,0)^T = (4,4,2)^T + (6,0,0)^T = (10,4,2)^T\\
&\gamma = 10^2 + 4^2 + 2^2 = 0.5(100+16+4) = 60\\
&H = I - \frac{vv^T}{\gamma}\\
&vv^T = \begin{pmatrix}10\\4\\2\end{pmatrix} \cdot (10,4,2) = \begin{pmatrix}100 & 40 & 20\\ 40 & 16 & 8\\ 20 & 8 & 4\end{pmatrix}\\
&\frac{vv^T}{\gamma} = \frac{1}{60} \cdot \begin{pmatrix}100 & 40 & 20\\ 40 & 16 & 8\\ 20 & 8 & 4\end{pmatrix} = \begin{pmatrix}\frac{5}{3} & \frac{2}{3} & \frac{1}{3}\\ \frac{2}{3} & \frac{4}{15} & \frac{2}{15}\\ \frac{1}{3} & \frac{2}{15} & \frac{1}{15}\end{pmatrix}\\
&I - \frac{vv^T}{\gamma} = \begin{pmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{pmatrix} - \begin{pmatrix}\frac{5}{3} & \frac{2}{3} & \frac{1}{3}\\ \frac{2}{3} & \frac{4}{15} & \frac{2}{15}\\ \frac{1}{3} & \frac{2}{15} & \frac{1}{15}\end{pmatrix} = \begin{pmatrix}-\frac{2}{3} & -\frac{2}{3} & -\frac{1}{3}\\ -\frac{2}{3} & \frac{11}{15} & -\frac{2}{15}\\ -\frac{1}{3} & -\frac{2}{15} & \frac{14}{15}\end{pmatrix}
\end{aligned}
$$

(2) : On peut voir $x' = (3,4,4,2)$ comme un sur-ensemble de $x = 4,4,2$ et comme on cherche juste à avoir un vecteur $(3,\star,0,0)$ il suffit de reproduire la matrice H de l'exercice précédent :

$$
H = \begin{pmatrix}1 & 0 & 0 & 0\\ 0 & -\frac{2}{3} & -\frac{2}{3} & -\frac{1}{3}\\ 0 & -\frac{2}{3} & \frac{11}{15} & -\frac{2}{15}\\ 0 & -\frac{1}{3} & -\frac{2}{15} & \frac{14}{15}\end{pmatrix}
$$

(3,a) : Comme on veut que les deux dernière composantes soient à zero, il faut calculer les paramètres pour la matrice $(8,4,1)$:

$$
\begin{aligned}
&\rho = 9\\
&v = (17,4,1)^T\\
&\gamma = 153
\end{aligned}
$$

(3,b) : Idem :

$$
\begin{aligned}
&\rho = -3\\
&v = (-5,-1,2)^T\\
&\gamma = 15
\end{aligned}
$$

(4,a) : On applique l'algorithme classique avec un vecteur de longueur 4 :

$$
\begin{aligned}
&\rho_1 = 6\\
&v_1 = (9,1,1,5)^T\\
&\gamma_1 = 54
\end{aligned}
$$

(4,b) :

$$\tau_2 = \frac{1}{54}(9,1,1,5)\cdot\begin{pmatrix}3\\1\\-5\\-1\end{pmatrix} = \frac{27+1-5-5}{54} = \frac{18}{54} = \frac{1}{3}$$

$$\tau_3 = \frac{1}{54}(9,1,1,5)\cdot\begin{pmatrix}-2\\1\\1\\2\end{pmatrix} = \frac{-18+1+1+10}{54} = \frac{-6}{54} = -\frac{1}{9}$$

(7,a) : $x_{1a} = 1,\, x_{2a} = 1,\, x_{1b} = 1,\, x_{2b} = 1$

(7,b) :

$$\begin{aligned} x^{(0)} &= (0,0)^T \\ x^{(1)} &= (\frac{1}{10}\cdot\Big(11 - \Big),)^T \\ x^{(2)} &= 1 \\ x^{(3)} &= 2 \\ x^{(4)} &= 3 \end{aligned}$$

7. Résolution numérique d'EDO

On veut résoudre $\frac{dy}{dt} = f(t, y)$ sur l'intervalle $[t_0, t_f]$ avec $y(t_0) = y_0$. On recherche dont l'inconnue $y(t) = ...$
Si $f(t, y)$ obéit à une condition de Lipschitz $|f(t, y_1) - f(t, y_2)| \leq |y_1 - y_2|$, $t_0 < t \leq t_f$. Pour tout y_1 et y_2, alors $\frac{dy}{dt} = f(t, y)$ admet une solution unique sur $[t_0, t_f]$.

7.1 Solution numérique

On va créer une suite de valeur pour t et y : $t = \{t_0, t_1, ..., t_n\}$ et $y = \{y_1, y_2, ..., y_n\}$ et on souhaiterais $\forall y_i \in y$: $y_i \approx y(t_i)$ et donc $y_n \approx y(t_n) = y(t_f)$.

7.2 Méthode d'Euler

On utilise la pente en (t_n, y_n) pour calculer y_{n+1} (Runge-Kutta à 1 étage) :

$$\begin{cases} s_1 = f(t_n, y_n) \\ y_{n+1} = y_n + hs_1, \quad n = 0, 1, 2, ... \\ t_{n+1} = t_n + h \end{cases}$$

Exemple : On veut résoudre $\frac{dy}{dt} = t + y$, $y(0) = 2$, $0 \leq t \leq 1$:

```
%% Euler example h=0.5
h = 0.5;
t = 0:h:1;
f_ty = @(f,t) f + t;
y = 2;

for tn = t
    s = f_ty(tn,y);
    fprintf('t=%4.2f , y=%4.2f\n',tn,y);
    y = y + h * s;
end
```

```
t=0.00 , y=2.00
t=0.50 , y=3.00
t=1.00 , y=4.75
```

```
%% Euler example h=0.2
h = 0.2;
t = 0:h:1;
f_ty = @(f,t) f + t;
y = 2;

for tn = t
    s = f_ty(tn,y);
    fprintf('t=%4.2f , y=%4.2f\n',tn,y);
    y = y + h * s;
end
```

```
t=0.00 , y=2.00
t=0.20 , y=2.40
t=0.40 , y=2.92
t=0.60 , y=3.58
t=0.80 , y=4.42
t=1.00 , y=5.46
```

7.3 Méthode du point du millieu

Il s'agit ici d'une méthode de Runge-Kutta à 2 étages :

$$\begin{cases} s_1 = f(t_n, y_n) \\ s_2 = f(t_n + \frac{1}{2}h, y_n + \frac{1}{2}hs_1) \\ y_{n+1} = y_n + hs_2 \\ t_{n+1} = t_n + h \end{cases}$$

Exemple : On veut résoudre $\frac{dy}{dt} = t + y$, $y(0) = 2$, $0 \leq t \leq 1$:

```
%% Midlle point example
h = 0.2;
t = 0:h:1;
f_ty = @(f,t) f + t;
y = 2;

for tn = t
    s1 = f_ty(tn,y);
    s2 = f_ty(tn+0.5*h,y+0.5*h*s1);
    fprintf('t=%4.2f , y=%4.2f\n',tn,y);
    y = y + h * s2;
end
```

```
t=0.00 , y=2.00
t=0.20 , y=2.46
t=0.40 , y=3.07
t=0.60 , y=3.85
t=0.80 , y=4.85
t=1.00 , y=6.11
```

7.4 Méthode du trapèze

Il s'agit aussi d'une méthode de Runge-Kutta à 2 étages :

$$\begin{cases} s_1 = f(t_n, y_n) \\ s_2 = f(t_n + h, y_n + hs_1) \\ y_{n+1} = y_n + h \cdot (\frac{1}{2}s_1 + \frac{1}{2}s_2) \\ t_{n+1} = t_n + h \end{cases}$$

Exemple : On veut résoudre $\frac{dy}{dt} = t + y$, $y(0) = 2$, $0 \leq t \leq 1$:

```
%% Trapeze method
h = 0.2;
t = 0:h:1;
f_ty = @(f,t) f + t;
y = 2;

for tn = t
    s1 = f_ty(tn,y);
    s2 = f_ty(tn+h,y+h*s1);
    fprintf('t=%4.2f , y=%4.2f\n',tn,y);
    y = y + h * (0.5*s1 + 0.5*s2);
end
```

```
t=0.00 , y=2.00
t=0.20 , y=2.46
t=0.40 , y=3.07
t=0.60 , y=3.85
t=0.80 , y=4.85
t=1.00 , y=6.11
```

7.5 Méthode de Runge-Kutta classique

La méthode la plus utilisé est la $RK4$ dite classique :

$$\begin{cases} s_1 = f(t_n, y_n) \\ s_2 = f(t_n + \frac{1}{2}h, y_n + \frac{1}{2}hs_1) \\ s_3 = f(t_n + \frac{1}{2}h, y_n + \frac{1}{2}hs_2) \\ s_4 = f(t_n + h, y_n + hs_3) \\ y_{n+1} = y_n + h \cdot (\frac{1}{6}s_1 + \frac{1}{3}s_2 + \frac{1}{3}s_3 + \frac{1}{6}s_4) \\ t_{n+1} = t_n + h \end{cases}$$

Exemple : On veut résoudre $\frac{dy}{dt} = t + y$, $y(0) = 2$, $0 \leq t \leq 1$:

```
%% RK4 method
h = 0.2;
t = 0:h:1;
f_ty = @(f,t) f + t;
y = 2;

for tn = t
    s1 = f_ty(tn,y);
    s2 = f_ty(tn+0.5*h,y+0.5*h*s1);
    s3 = f_ty(tn+0.5*h,y+0.5*h*s2);
    s4 = f_ty(tn+h,y+h*s3);
    fprintf('t=%4.4f , y=%4.4f\n',tn,y);
    y = y + h * ((1/6)*s1 + (1/3)*s2 +
↪   (1/3)*s3 + (1/6)*s4);
end
```

```
t=0.0000 , y=2.0000
t=0.2000 , y=2.4642
t=0.4000 , y=3.0755
t=0.6000 , y=3.8663
t=0.8000 , y=4.8766
t=1.0000 , y=6.1548
```

7.6 Erreurs locale et globale de la méthode d'Euler

Pour Euler, on peut montrer que
- l'erreur locale vaut $|y(t_1) - y(1)| \approx C \cdot h^2$
- l'erreur globale vaut $|y(t_n) - y(n)| \approx C \cdot h$

8. Code source

9 - Exercice 7

```
%% Série 9
%Exercice 7
close all;
clear ;
format long g;

%% part a
data = [-2,128.6;-1,2.15;0,-7;1,-1.75];
t = data(:,1);
y = data(:,2);
m = size(data,1);

beta1 = (sum(t) * sum(y) - (m * sum(t .* y)))         / (sum(t)^2 - (m * sum(t .^ 2)));
beta2 = (sum(t) * sum(t .* y) -  sum(t .^ 2) * sum(y))/ (sum(t)^2 - m * (sum(t .^ 2)));

ft = @(t) beta1*t+beta2;

ri = feval(ft,t);
ri = y - ri;
r = sum(ri .* ri);

fprintf('Polynôme de degrés 1 : %d * t + %d\n',beta1,beta2);
fprintf('Somme des carrés des résidus : %d\n',r);
fprintf('Valeur du polynome en x=2.5 : f(2.5) = %d\n',feval(ft,2.5));
fprintf('For Latex\n\t%.2ft+%.2f\n\t%.2f\n\n',beta1,beta2,r);



figure;
scatter(t,y);
hold on;
plot(-3:0.01:3,feval(ft,-3:0.01:3));
scatter(2.5,feval(ft,2.5));

%% part b

A = [-2,1;-1,1;0,1;1,1];
A = [A(:,1) .^ 2,A];
y = [128.6;2.15;-7;-1.75];

At = A';
AtA = At * A;
Aty = At * y;
beta = AtA \ Aty;

figure;
scatter(t,y); hold on;
ft = @(t) beta(1) * t.^2 + beta(2) * t + beta(3);
plot(-3:0.01:3,feval(ft,-3:0.01:3));
scatter(2.5,feval(ft,2.5));
fprintf('\nPolynôme de degré 2 : %d * t^2 + %d * t + %d\n',beta(1),beta(2),beta(3));
fprintf('Valeur du polynome en x=2.5 : f(2.5) = %d\n',feval(ft,2.5));

ri = feval(ft,t);
ri = y - ri;
r = sum(ri .* ri);
fprintf('Somme des carrés des résidus : %d\n',r);
fprintf('Valeur du polynome en x=2.5 : f(2.5) = %d\n',feval(ft,2.5));
fprintf('For Latex\n\t%.2ft+t^2%.2ft+%.2f\n\t%.2f\n\n',beta(1),beta(2),beta(3),r);
```

```matlab
%% part c

A = [-2,1;-1,1;0,1;1,1];
A = [A(:,1) .^ 2,A];
A = [A(:,2) .^ 3,A];
y = [128.6;2.15;-7;-1.75];

At = A';
AtA = At * A;
Aty = At * y;
beta = AtA \ Aty;

figure;
scatter(t,y); hold on;
ft = @(t) beta(1) * t.^3 + beta(2) * t.^2 + beta(3) * t + beta(4);

ri = feval(ft,t);
ri = y - ri;
r = sum(ri .* ri);

plot(-3:0.01:3,feval(ft,-3:0.01:3));
scatter(2.5,feval(ft,2.5));
fprintf('\nPolynôme de degré 3 : %d * t^3 + %d * t^2 + %d * t +
 ↪   %d\n',beta(1),beta(2),beta(3),beta(4));
fprintf('Somme des carrés des résidus : %d\n',r);
fprintf('Valeur du polynome en x=2.5 : f(2.5) = %d\n',feval(ft,2.5));
fprintf('For
 ↪   Latex\n\t%.2ft^3+%.2ft^2+%.2ft+%.2f\n\t%.2f\n\n',beta(1),beta(2),beta(3),beta(4),r);
```