Composition de Mathématiques

Le 11 septembre 2023 – De 13 heures à 17 heures

Si, au cours de l'épreuve, un candidat repère ce qui lui semble être une erreur d'énoncé, il le signale sur sa copie et poursuit sa composition en expliquant les raisons des initiatives qu'il est amené à prendre.

La présentation et la rédaction comptent pour une part importante dans l'appréciation de la copie.

Les calculatrices sont interdites.
Les téléphones portables doivent être éteints et rangés.

❖ I – Problème ❖

On note cot la fonction *cotangente* définie par

$$\cot x = \frac{\cos x}{\sin x}.$$

1. Démontrer que la fonction cot réalise une bijection de $]0, \pi/2[$ sur $\mathbb{R}_+^*$.

2. Démontrer que, pour tout $n \in \mathbb{N}$ et tout $\theta \in \mathbb{R}$,

$$\sin(2n+1)\theta = \sum_{k=0}^{n} \binom{2n+1}{2k} (-1)^{n-k} \cos^{2k}\theta . \sin^{(2n+1)-2k}\theta.$$

3. En déduire que, pour tout entier naturel non nul n, il existe un unique polynôme P_n à coefficients réels tel que

$$\forall\, \theta \in \,]0, \pi/2[\,, \qquad P_n(\cot^2\theta) = \frac{\sin\big[(2n+1)\theta\big]}{\sin^{2n+1}\theta}.$$

4. Donner en fonction de n le degré de P_n, son coefficient dominant, la somme de ses racines ainsi que le produit de ses racines.

5. Démontrer que, pour tout entier naturel n non nul, le polynôme P_n possède n racines réelles strictement positives distinctes.

6. a. Démontrer que

$$\forall\, \theta \in \,]0, \pi/2[\,, \qquad \cot^2\theta \leqslant \frac{1}{\theta^2} \leqslant 1 + \cot^2\theta.$$

6. b. En déduire que

$$\forall n \geqslant 1, \qquad \frac{\pi^2(2n-1)n}{3(2n+1)^2} \leqslant \sum_{k=1}^{n} \frac{1}{k^2} \leqslant \frac{\pi^2(2n-1)n}{3(2n+1)^2} + \frac{n\pi^2}{(2n+1)^2}.$$

et conclure.

❖ II – Problème ❖

Dans tout le problème, on identifiera les polynômes *et les* fonctions polynomiales.
Ainsi, la notation $\mathbb{R}[X]$ *désignera aussi bien l'espace vectoriel des polynômes à coefficients réels qu'un sous-espace de l'espace* $\mathscr{C}^\infty(\mathbb{R})$ *des applications indéfiniment dérivables de* $\mathbb{R}$ *dans* $\mathbb{R}$.

1. Quel théorème rend légitime l'identification des polynômes aux fonctions polynomiales ?

Pour un entier $n \in \mathbb{N}$, *on suppose donnée une famille*

$$(x_k)_{0\leqslant k\leqslant n} = (x_0, \dots, x_n)$$

de nombres réels deux à deux distincts. *On pose alors*

$$\forall\, 0 \leqslant k \leqslant n, \quad L_k = \prod_{\substack{0\leqslant i\leqslant n \\ i\neq k}} \frac{X - x_i}{x_k - x_i} \in \mathbb{R}[X]$$

ainsi que

$$T_{n+1} = \prod_{k=0}^{n} (X - x_k).$$

On remarquera que ces notations peuvent conduire à des confusions : si les valeurs ou le nombre des x_i *changent, les polynômes* L_k *et* T_{n+1} *changent aussi.*

Partie A. Interpolation polynomiale

Dans cette partie, l'entier $n \in \mathbb{N}$ *est fixé, ainsi que la famille des réels* $(x_i)_{0\leqslant i\leqslant n}$.
Quelles que soient les applications f *et* g *dans* $\mathscr{C}^\infty(\mathbb{R})$, *on pose*

$$B(f, g) = \sum_{i=0}^{n} f(x_i)g(x_i).$$

2. a. Démontrer que B est un produit scalaire sur le sous-espace $\mathbb{R}_n[X]$ des polynômes de degré inférieur à n. Est-ce un produit scalaire sur l'espace $\mathbb{R}[X]$?
2. b. Démontrer que la famille $(L_k)_{0\leqslant k\leqslant n}$ est une famille orthonormée de $\mathbb{R}_n[X]$ pour le produit scalaire B.
2. c. En déduire que cette famille est une base de $\mathbb{R}_n[X]$.

Pour toute fonction $f \in \mathscr{C}^\infty(\mathbb{R})$, *on note*

$$\mathscr{L}(f) = \sum_{j=0}^{n} B(f, L_j)L_j.$$

3. a. Démontrer que $\mathscr{L}$ est une application linéaire de $\mathscr{C}^\infty(\mathbb{R})$ dans $\mathbb{R}_n[X]$.
3. b. Calculer $B(f, L_j)$ pour tout $0 \leqslant j \leqslant n$. En déduire que

$$\forall\, 0 \leqslant k \leqslant n, \quad \mathscr{L}(f)(x_k) = f(x_k).$$

3. c. On suppose qu'un polynôme $P \in \mathbb{R}_n[X]$ vérifie la propriété

$$\forall\, 0 \leqslant k \leqslant n, \quad P(x_k) = f(x_k).$$

Démontrer que $P = \mathscr{L}(f)$.

D'après ce qui précède, pour toute application $f \in \mathscr{C}^\infty(\mathbb{R})$, *le polynôme* $\mathscr{L}(f)$ *est l'unique polynôme de degré inférieur à* n *qui prend les mêmes valeurs que l'application* f *aux points* $x_0, \dots, x_n$.

4. Dans cette question, on suppose que l'application f est polynomiale : $f \in \mathbb{R}[X]$.
4. a. Comparer f et $\mathscr{L}(f)$ lorsque le degré de f est inférieur à n (soit : $f \in \mathbb{R}_n[X]$).
4. b. En déduire les valeurs des sommes

$$\sum_{k=0}^{n} L_k(t) \qquad \text{et} \qquad \sum_{k=0}^{n} x_k L_k(t)$$

pour tout $t \in \mathbb{R}$.
4. c. Plus généralement, caractériser le polynôme $\mathscr{L}(X^p)$ pour tout $p \in \mathbb{N}$.
5. Démontrer que l'application linéaire $\mathscr{L}$ est surjective.
6. **Cas particulier**

Dans cette question, on fait l'hypothèse que l'application $f \in \mathscr{C}^\infty(\mathbb{R})$ est paire ; que le nombre entier n est impair ($n = 2p + 1$ pour un entier $p \in \mathbb{N}$ convenable) et que les points d'interpolation x_k vérifient la propriété suivante :

$$\forall\, 0 \leqslant k \leqslant p, \quad x_{n-k} = -x_k.$$

Pour simplifier les calculs, on notera $y_k = f(x_k)$.

On notera L, l'unique polynôme de $\mathbb{R}_n[X]$ tel que $L(x_k) = y_k$ pour tout $0 \leqslant k \leqslant n$ (cf **[3.c.]**).
6. a. L'un des x_k peut-il être nul ?
6. b. Comparer y_{n-k} et y_k pour $0 \leqslant k \leqslant p$.
6. c. Démontrer qu'il existe un, et un seul, polynôme $P \in \mathbb{R}_p[X]$ tel que

$$\forall\, 0 \leqslant k \leqslant p, \quad P(x_k^2) = y_k.$$

6. d. Exprimer le polynôme L en fonction du polynôme P. En déduire que L est pair et que le degré de L est strictement inférieur à n.

Partie B. Évaluation du polynôme interpolateur

Dans cette partie, le nombre et les valeurs des abscisses d'interpolation peuvent varier.
Nous dirons qu'un polynôme P ***interpole*** *la fonction* f ***aux points*** $a_0, \dots, a_m$ *lorsque* P *est l'unique polynôme de degré inférieur à* m *tel que*

$$\forall\, 0 \leqslant k \leqslant m, \quad P(a_k) = f(a_k).$$

Dans la première partie (cf **[3.]***), on a trouvé une expression algébrique explicite du polynôme* $P = \mathscr{L}(f)$ *qui interpole la fonction* f *aux points* $x_0, \dots, x_n$.
L'objectif de cette partie est d'obtenir un algorithme efficace pour évaluer $P(t)$ *pour tout* $t \in \mathbb{R}$.

7. On considère $(n + 2)$ réels deux à deux distincts :

$$a, b, x_1, x_2, \dots, x_n.$$

On suppose que le polynôme P interpole la fonction f aux points
$$b, x_1, x_2, \dots, x_n$$
et que le polynôme Q interpole la même fonction f aux points
$$a, x_1, x_2, \dots, x_n.$$
Démontrer que le polynôme
$$S = \frac{(X-a)P - (X-b)Q}{b-a}$$
interpole la fonction f aux points
$$a, b, x_1, x_2, \dots, x_n.$$

8. Dans cette question, on considère deux tableaux numpy qu'on représente comme des listes de réels.
$$\mathtt{X} = (x_0, x_1, \dots, x_{n-1}, x_n)$$
$$\mathtt{Y} = (y_0, \dots, y_p)$$

8. a. Par quels tableaux numpy peut-on représenter les listes suivantes ?
$$(x_0, x_1, x_2, \dots, x_{n-1}) \qquad (x_1, x_2, \dots, x_{n-1}, x_n)$$
En déduire le tableau numpy qui représente la liste suivante.
$$(x_1 - x_0, x_2 - x_1, \dots, x_n - x_{n-1})$$

8. b. On se donne un entier $k \geqslant 1$. Par quel tableau numpy peut-on représenter la liste suivante ?
$$(y_k - y_0, y_{k+1} - y_1, y_{k+2} - y_2, \dots)$$
En déduire le tableau numpy qui représente la liste suivante.
$$\left(\frac{x_1 - x_0}{y_k - y_0}, \frac{x_2 - x_1}{y_{k+1} - y_1}, \frac{x_3 - x_2}{y_{k+2} - y_2}, \dots\right)$$
☞ *On précisera la condition nécessaire et suffisante sur n et p pour que ces listes soient bien définies.*

Soient x_0, ..., x_n, des abscisses deux à deux distinctes et f, une application à valeurs réelles, définie aux abscisses x_i. Pour tout $x \in \mathbb{R}$, on pose
$$T_0(x) = 1$$
et, pour tout entier $1 \leqslant d \leqslant n+1$,
$$T_d(x) = (x - x_0) \cdots (x - x_{d-1}) = \prod_{0 \leqslant k < d} (x - x_k).$$
On remarquera que $T_{d+1}(x) = (x - x_d)T_d(x)$.

9. Démontrer que la famille $(T_k)_{0 \leqslant k \leqslant n}$ est une base de l'espace vectoriel $\mathbb{R}_n[X]$.

*Les **différences divisées** sont définies par récurrence :*
❦ *Au **rang** 0, on note*
$$\forall\, i \in [\![0, n]\!], \qquad f_0[x_i] = f(x_i).$$
❦ *Au **rang** 1, on pose*
$$f_1[x_i, x_{i+1}] = \frac{f(x_i) - f(x_{i+1})}{x_i - x_{i+1}} = \frac{f_0[x_i] - f_0[x_{i+1}]}{x_i - x_{i+1}}$$
pour tout entier $i \in [\![0, (n-1)]\!]$.
❦ *En supposant que $f_k[x_i, \dots, x_{i+k}]$ soit définie pour tout indice $i \in [\![0, (n-k)]\!]$, la valeur de*
$$f_{k+1}[x_i, \dots, x_{i+(k+1)}]$$
est, par définition, égale à
$$\frac{f_k[x_i, \dots, x_{i+k}] - f_k[x_{i+1}, \dots, x_{i+1+k}]}{x_i - x_k}.$$

10. Pour quels indices i l'expression
$$f_{k+1}[x_i, \dots, x_{i+(k+1)}]$$
est-elle définie ?

11. Quelle est la dernière valeur calculée par l'algorithme des différences divisées ?

12. Soient $(x_i)_{0 \leqslant i \leqslant n}$, une famille d'éléments deux à deux distincts de I et L, le polynôme qui interpole une fonction f en ces points.

12. a. Démontrer que le coefficient du terme de degré n de L est égal à $f_n[x_0, \dots, x_n]$.

12. b. Démontrer que, pour tout $t \in \mathbb{R}$,
$$L(t) = \sum_{k=0}^{n} f_k[x_0, \dots, x_k] T_k(t).$$

13. On suppose que les points d'interpolation et les valeurs de f sont contenus dans deux listes.
$$\mathtt{X} = (x_k)_{0 \leqslant k \leqslant n} \qquad \mathtt{Y} = \big(f(x_k)\big)_{0 \leqslant k \leqslant n}$$

13. a. Écrire (en langage Python) une fonction

```
Differences_Divisees(X, Y)
```

qui renvoie la liste
$$\mathtt{D} = \big(f_1[x_0], f_2[x_0, x_1], \dots, f_{n+1}[x_0, \dots, x_n]\big).$$

13. b. Écrire une fonction

```
Lagrange(X, D, t)
```

qui renvoie la valeur $L(t)$.

13. c. Discuter l'efficacité de l'algorithme.

Solution I ❀ Une série célèbre

1. La fonction cot est définie et dérivable sur l'intervalle $]0, \pi/2[$ (en tant que quotient de fonctions dérivables) et sur cet intervalle,

$$\cot'(x) = -(1+\cot^2 x) = \frac{-1}{\sin^2 x} < 0.$$

Elle tend vers $+\infty$ au voisinage droit de 0 et vers 0 au voisinage de $\pi/2$. Par conséquent, la fonction cot réalise une bijection (strictement décroissante) de $]0, \pi/2[$ sur $]0, 1[$.

2. Pour tout entier naturel n et pour tout réel θ,

$$\sin(2n+1)\theta = \mathfrak{Im}(e^{i(2n+1)\theta}) = \mathfrak{Im}\big((\cos\theta + i\sin\theta)^{2n+1}\big).$$

D'après la formule du binôme et en scindant la somme obtenue en deux parties selon la parité de l'indice,

$$\begin{aligned} e^{i(2n+1)\theta} &= \sum_{p=0}^{2n+1} \binom{2n+1}{p} \cos^p\theta.\sin^{(2n+1)-p}\theta.i(i^{2n-p}) \\ &= i\sum_{k=0}^{n} \binom{2n+1}{2k} (-1)^{n-k}\cos^{2k}\theta.\sin^{(2n+1)-2k}\theta & \text{(p pairs)} \\ &\qquad + \sum_{k=0}^{n} \binom{2n+1}{2k+1} (-1)^{n-k}\cos^{2k+1}\theta.\sin^{2(n-k)}\theta. & \text{(p impairs)} \end{aligned}$$

Par conséquent, pour tout $n \in \mathbb{N}$ et tout $\theta \in \mathbb{R}$,

$$\sin(2n+1)\theta = \sum_{k=0}^{n} \binom{2n+1}{2k} (-1)^{n-k}\cos^{2k}\theta.\sin^{(2n+1)-2k}\theta.$$

3. Soient P_n et R_n, deux polynômes tels que

$$\forall\, \theta \in]0, \pi/2[, \qquad P_n(\cot^2\theta) = R_n(\cot^2\theta).$$

D'après **1.**, pour tout $x \in \mathbb{R}_+^*$, il existe $0 < \theta < \pi/2$ tel que $x = \cot^2\theta$, donc P_n et Q_n coïncident sur $\mathbb{R}_+^*$. Comme $\mathbb{R}_+^*$ est un ensemble <u>infini</u>, on en déduit que $P_n = R_n$. Il existe donc au plus un polynôme vérifiant la propriété voulue.

Lorsque $0 < \theta < \frac{\pi}{2}$, on divise l'identité du **2.** par $\sin^{2n+1}\theta$ (qui est strictement positif [1]) et on trouve

$$\begin{aligned} \frac{\sin(2n+1)\theta}{\sin^{2n+1}\theta} &= \sum_{k=0}^{n} \binom{2n+1}{2k} (-1)^{n-k}\frac{\cos^{2k}\theta}{\sin^{2k}\theta} \\ &= \sum_{k=0}^{n} \binom{2n+1}{2k} (-1)^{n-k}(\cot^2\theta)^k. \end{aligned}$$

Ainsi le polynôme

$$P_n = \sum_{k=0}^{n} \binom{2n+1}{2k} (-1)^{n-k} X^k \tag{$\star$}$$

répond à la question.

4. D'après $(\star)$, le degré du polynôme P_n est égal à n, son coefficient dominant à $\binom{2n+1}{2n} = 2n+1$. La somme de ses racines se déduit du coefficient de degré $n-1$ et vaut

$$-\frac{(-1)}{2n+1}\binom{2n+1}{2(n-1)} = \frac{1}{2n+1}\binom{2n+1}{3} = \frac{(2n+1)(2n)(2n-1)}{1.2.3.(2n+1)} = \frac{n(2n-1)}{3}.$$

Le produit des racines se déduit du coefficient constant et vaut

$$(-1)^n\frac{(-1)^n}{2n+1}\binom{2n+1}{0} = \frac{1}{2n+1}.$$

REMARQUE.— Il faut savoir retrouver la somme et le produit des racines d'un polynôme à la simple lecture de ses coefficients :

$$a\prod_{k=1}^{n}(X-\alpha_k) = aX^n + \Big[-a\Big(\sum_{k=1}^{n}\alpha_k\Big)\Big]X^{n-1} + \cdots + \Big[a(-1)^n\prod_{k=1}^{n}\alpha_k\Big].$$

1. On ne divise pas par zéro, c'est mal.

5. Il est clair d'après **1.** que $\cot^2$ réalise une bijection de $]0, \pi/2[$ sur $\mathbb{R}_+^*$.

ANALYSE.— Soit z, une racine strictement positive de P_n. Il existe alors $0 < \theta < \frac{\pi}{2}$ tel que $z = \cot^2\theta$ et donc tel que $P_n(\cot^2\theta) = 0$. Alors $\sin(2n+1)\theta = 0$ d'après **3.**, donc $(2n+1)\theta$ est nécessairement un multiple entier de π.

SYNTHÈSE.— Pour tout entier $1 \leqslant k \leqslant n$, on pose

$$\theta_k = \frac{k\pi}{2n+1}.$$

Tous ces nombres appartiennent à l'intervalle ouvert $]0, \pi/2[$ et $\sin\big[(2n+1)\theta_k\big] = 0$ quel que soit $1 \leqslant k \leqslant n$. D'après **3.**, les nombres réels $\cot^2\theta_k$, $1 \leqslant k \leqslant n$, sont donc des racines strictement positives du polynôme P_n.

Comme $\cot^2$ est injective sur $]0, \pi/2[$, les n réels $\cot^2\theta_k$, $1 \leqslant k \leqslant n$, sont deux à deux distincts.

REMARQUE.— Comme $\deg P_n = n$, on sait que P_n possède au plus n racines distinctes dans $\mathbb{C}$. Par conséquent, les réels $\cot^2\theta_k$, $1 \leqslant k \leqslant n$, sont *les* racines de P_n.

6. a. L'étude des variations de sin et de tan montre que $\tan\theta > \theta > \sin\theta > 0$ pour tout $0 < \theta < \pi/2$. Par conséquent,

$$\forall\, 0 < \theta < \frac{\pi}{2}, \qquad \cot^2\theta = \frac{1}{\tan^2\theta} \leqslant \frac{1}{\theta^2} \leqslant \frac{1}{\sin^2\theta} = 1 + \cot^2\theta.$$

REMARQUE.— On peut aussi établir ces inégalités à l'aide de la formule des accroissements finis.

6. b. On applique l'encadrement précédent à chacun des θ_k définis à la question **5.** et on somme ces encadrements. Comme la somme des $\cot^2\theta_k$ a été calculée en **4.** (c'est la somme des racines de P_n), on en déduit que

$$\frac{n(2n-1)}{3} = \sum_{k=1}^{n} \cot^2\theta_k \leqslant \sum_{k=1}^{n} \frac{1}{\theta_k^2} \leqslant n + \sum_{k=1}^{n} \cot^2\theta_k = n + \frac{n(2n-1)}{3}.$$

D'après l'expression des θ_k (question **5.**), on en déduit que

$$\frac{\pi^2}{(2n+1)^2}\,\frac{n(2n-1)}{3} \leqslant \sum_{k=1}^{n} \frac{1}{k^2} \leqslant \frac{\pi^2}{(2n+1)^2}\left[n + \frac{n(2n-1)}{3}\right]$$

pour tout entier naturel $n \geqslant 1$. Les termes extrêmes de cet encadrement tendent tous deux vers $\pi^2/6$. Par encadrement, on en déduit que

$$\lim_{n\to+\infty} \sum_{k=1}^{n} \frac{1}{k^2} = \frac{\pi^2}{6}.$$

Solution II ❀ Interpolation de Lagrange

1. Si Ω est une partie *infinie* de $\mathbb{K}$, alors l'application

$$\begin{array}{ccc} \mathbb{K}[X] & \longrightarrow & \mathscr{A}(\Omega, \mathbb{K}) \\ P & \longmapsto & \widetilde{P} \end{array}$$

qui associe l'application polynomiale $\widetilde{P} : \Omega \to \mathbb{K}$ au polynôme $P \in \mathbb{K}[X]$ est <u>injective</u>.

Autrement dit, pour chaque application polynomiale $f \in \mathscr{A}(\Omega, \mathbb{K})$, il existe un, et un seul, polynôme P tel que $\widetilde{P} = f$ et cette unicité permet d'identifier la fonction f au polynôme P.

Partie A. Interpolation polynomiale

2. a. Il est clair que B est une application de $\mathscr{C}^\infty(\mathbb{R}) \times \mathscr{C}^\infty(\mathbb{R})$ dans $\mathbb{R}$; que cette application est bilinéaire et symétrique et que, pour tout $f \in \mathscr{C}^\infty(\mathbb{R})$,

$$B(f, f) = \sum_{i=0}^{n} f^2(x_i) \geqslant 0$$

(puisque les $f(x_i)$ sont des réels — pas des complexes !).

Enfin, si $B(f, f) = 0$, alors

$$\forall\, 0 \leqslant i \leqslant n, \quad f(x_i) = 0$$

puisqu'une somme de réels positifs est nulle si, et seulement si, tous les termes sont nuls.

❧ Si $f \in \mathbb{R}_n[X]$, alors f est un polynôme de degré inférieur à n qui admet au moins $(n+1)$ racines distinctes (les x_i), donc f est le polynôme nul et cela permet de conclure : l'application B est bien un produit scalaire sur $\mathbb{R}_n[X]$.

❧ Si $f \in \mathbb{R}[X]$, alors f est divisible par le polynôme T_{n+1}, mais nécessairement nulle : $B(T_{n+1}, T_{n+1}) = 0$ alors que $T_{n+1} \neq 0$. Dans ce cas, l'application B n'est pas un produit scalaire.

2. b. Tout d'abord, les polynômes L_k sont bien définis : comme les abscisses x_i sont deux à deux distinctes, le dénominateur ne contient aucun facteur nul.

Ensuite, en tant que produit de n facteurs de degré 1, chaque L_k est un polynôme de degré n, donc $(L_k)_{0\leqslant k\leqslant n}$ est bien une famille de vecteurs de $\mathbb{R}_n[X]$.

❧ Nous allons maintenant calculer les $L_k(x_j)$, avant de calculer les produits scalaires.

❧ Si $j = k$, alors

$$L_k(x_j) = \prod_{\substack{0\leqslant i\leqslant n \\ i\neq k}} \frac{x_k - x_i}{x_k - x_i} = 1.$$

Si $j \neq k$, alors

$$L_k(x_j) = \prod_{\substack{0 \leqslant i \leqslant n \\ i \neq k}} \frac{x_j - x_i}{x_k - x_i} = 0.$$

En effet, comme $j \neq k$ et que i prend toutes les valeurs à l'exception de k, l'indice prend la valeur j et dans ce cas, le facteur $(x_j - x_i)$ est nul.

Ainsi,

$$\forall\, 0 \leqslant i, j \leqslant n, \quad L_k(x_j) = \delta_{i,j}.$$

❧ On en déduit que, quels que soient $0 \leqslant k, \ell \leqslant n$,

$$B(L_k, L_\ell) = \sum_{j=0}^{n} \delta_{j,k}\delta_{j,\ell} = \delta_{k,\ell}$$

(seul reste le terme pour $j = k$).

La famille $(L_k)_{0 \leqslant k \leqslant n}$ est donc une famille orthonormée de $\mathbb{R}_n[X]$.

2. c. On a déjà remarqué (à la question précédente) que $(L_k)_{0 \leqslant k \leqslant n}$ était une famille de $\mathbb{R}_n[X]$.

On sait que toute famille orthonormée est libre.

Enfin, on sait que la dimension $\dim \mathbb{R}_n[X] = n + 1$ est égale au cardinal de la famille $(L_k)_{0 \leqslant k \leqslant n}$. Cela prouve que la famille orthonormée est en fait une base orthonormée de $\mathbb{R}_n[X]$.

3. a. Pour tout $f \in \mathscr{C}^\infty(\mathbb{R})$, par définition, $\mathscr{L}(f)$ est une combinaison linéaire des polynômes L_k, donc **[2.c.]** $\mathscr{L}(f) \in \mathbb{R}_n[X]$.

D'autre part, l'application $\mathscr{L}$ est linéaire par linéarité à gauche de B.

$$\begin{aligned} \mathscr{L}(\lambda f + g) &= \sum_{k=0}^{n} \big[\lambda B(f, L_k) + B(g, L_k)\big] \cdot L_k \\ &= \lambda \mathscr{L}(f) + \mathscr{L}(g) \end{aligned}$$

3. b. On calcule d'abord les produits scalaires $B(f, L_k)$ avant de simplifier l'expression de $\mathscr{L}(f)(x_k)$.

❧ Pour tout entier $0 \leqslant k \leqslant n$,

$$\begin{aligned} B(f, L_k) &= \sum_{j=0}^{n} f(x_j) L_k(x_j) \\ &= \sum_{j=0}^{n} f(x_j)\delta_{j,k} && \textbf{[2.b.]} \\ &= f(x_k). \end{aligned}$$

❧ Toujours par **[2.b.]**, pour tout entier $0 \leqslant k \leqslant n$,

$$\begin{aligned} \mathscr{L}(f)(x_k) &= \sum_{j=0}^{n} f(x_j) L_j(x_k) && \text{(les indices ont changé)} \\ &= \sum_{j=0}^{n} f(x_j)\delta_{j,k} = f(x_k). \end{aligned}$$

3. c. D'après la question précédente,

$$\forall\, 0 \leqslant k \leqslant n, \quad \big(P - \mathscr{L}(f)\big)(x_k) = f(x_k) - f(x_k) = 0.$$

Par ailleurs, comme P et $\mathscr{L}(f)$ sont des polynômes de degré inférieur à n, alors $P - \mathscr{L}(f)$ appartient à $\mathbb{R}_n[X]$ et comme ce polynôme admet au moins $(n + 1)$ racines distinctes (les x_k), il s'agit du polynôme nul. Ainsi,

$$P = \mathscr{L}(f).$$

4. a. Si $f \in \mathbb{R}_n[X]$, alors f et $\mathscr{L}(f)$ sont deux polynômes de $\mathbb{R}_n[X]$ tels que

$$\forall\, 0 \leqslant k \leqslant n, \quad \mathscr{L}(f)(x_k) = f(x_k).$$

On déduit de **[3.c.]** que $\mathscr{L}(f) = f$.

4. b. Il est clair que $1 \in \mathbb{R}_n[X]$. D'après **[4.a.]**,

$$\mathscr{L}(1) = 1.$$

Or, pour tout $t \in \mathbb{R}$,

$$\sum_{k=0}^{n} L_k(t) = \sum_{k=0}^{n} 1.L_k(t) \overset{\textbf{[3.b.]}}{=} \mathscr{L}(1)(t)$$

et donc

$$\forall\, t \in \mathbb{R}, \quad \sum_{k=0}^{n} L_k(t) = 1.$$

❧ De même, d'après **[3.b.]**,

$$\forall\, t \in \mathbb{R}, \quad \sum_{k=0}^{n} x_k L_k(t) = \mathscr{L}(X)(t)$$

Si $n \geqslant 1$, alors $X \in \mathbb{R}_n[X]$. On déduit alors de **[4.a.]** que $\mathscr{L}(X) = X$ et donc que

$$\forall\, t \in \mathbb{R}, \quad \sum_{k=0}^{n} x_k L_k(t) = t.$$

❧ Si $n = 0$, on peut calculer directement :

$$\sum_{k=0}^{n} x_k L_k(t) = x_0 L_0(t) = x_0$$

puisque $L_0 = 1$ (un produit sans aucun facteur est, par convention, égal à 1).

REMARQUE.— On trouvera une explication théorique de ce résultat au **[4.c.]**.

4. c. D'après **[4.a.]**,

$$\forall\, 0 \leqslant p \leqslant n, \quad \mathscr{L}(X^p) = X^p.$$

❧ On suppose maintenant que $p > n$, donc

$$\mathbb{R}_n[X] \ni \mathscr{L}(X^p) \neq X^p \notin \mathbb{R}_n[X].$$

Comme $\mathscr{L}(X^p) \in \mathbb{R}_n[X]$, on peut déduire de **[4.a.]** que

$$\mathscr{L}\big(\mathscr{L}(X^p)\big) = \mathscr{L}(X^p)$$

et donc que

$$\mathscr{L}\big(\mathscr{L}(X^p) - X^p\big) = 0.$$

D'après **[3.c.]**, le polynôme

$$\mathscr{L}(X^p) - X^p$$

admet $x_0, \dots, x_n$ pour racines. Comme ces réels sont deux à deux distincts, il existe un polynôme $Q \in \mathbb{R}[X]$ tel que

$$X^p - \mathscr{L}(X^p) = (X - x_0)\cdots(X - x_n)Q.$$

Autrement dit,

$$X^p = T_{n+1}Q + \mathscr{L}(X^p).$$

Comme $\deg T_{n+1} = n+1 > n \geqslant \deg \mathscr{L}(X^p)$, on reconnaît ici la division euclidienne de X^p par T_{n+1} et $\mathscr{L}(X^p)$ est le reste de cette division euclidienne.

❧ On peut ainsi conclure de manière générale : pour toute fonction polynomiale P (quel que soit son degré), le polynôme interpolateur $\mathscr{L}(P)$ est le reste de la division euclidienne de P par le polynôme T_{n+1}.

❧ (Retour au **[4.b.]**) Si $n = 0$, il s'agit de calculer la division euclidienne de X par $(X - x_0)$.

On sait que, pour tout polynôme P, la division euclidienne de P par $(X - x_0)$ s'écrit

$$P = (X - x_0).Q + P(x_0).$$

En particulier pour $P = X$,

$$\forall\, t \in \mathbb{R}, \quad \sum_{k=0}^{n} x_k L_k(t) = x_0,$$

ce que confirme le calcul mené à la question précédente.

5. Par définition **[3.a.]**, l'application $\mathscr{L}$ est linéaire de $\mathscr{C}^\infty(\mathbb{R})$ dans $\mathbb{R}_n[X]$.

D'après **[4.a.]**,

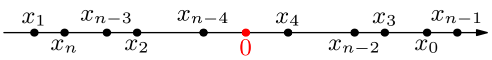
$$\forall\, f \in \mathbb{R}_n[X], \quad f = \mathscr{L}(f) \in \operatorname{Im} \mathscr{L}$$

donc $\mathscr{L}$ est bien surjective.

6. Le nombre de points d'interpolation est **pair** :

$$n + 1 = 2p + 2$$

et ces abscisses ne sont a priori pas ordonnées.



6. a. Si $x_k = 0$, alors il faut que $x_{n-k} = -x_k = 0$. Comme les x_i sont deux à deux distincts, il faut donc que $(n-k) = k$ et donc que $n = 2k$.

Or, par hypothèse, l'entier n est *impair*, donc les x_k sont différents de 0.

6. b. Par parité de f,

$$\forall\, 0 \leqslant k \leqslant p, \quad y_{n-k} = f(x_{n-k}) = f(-x_k) = f(x_k) = y_k.$$

6. c. S'il existe deux entiers $0 \leqslant k, \ell \leqslant p$ tels que

$$x_k^2 = x_\ell^2,$$

alors $x_k = \pm x_\ell$, c'est-à-dire

$$x_k = x_\ell \qquad \text{ou} \qquad x_k = x_{n-\ell}.$$

Comme les x_i sont deux à deux distincts, on en déduit que $k = \ell$ ou que $k = n - \ell$.

Mais $0 \leqslant k, \ell \leqslant p$, donc, dans le second cas,

$$k = n - \ell = (2p+1) - \ell \geqslant p + 1 > p,$$

ce qui est contradictoire.

On a ainsi démontré que $(x_k^2)_{0\leqslant k\leqslant p}$ est une famille de réels deux à deux distincts. D'après **[3.]**, quelle que soit la famille $(y_k)_{0\leqslant k\leqslant p}$, il existe un, et un seul, polynôme $P \in \mathbb{R}_p[X]$ tel que

$$\forall\, 0 \leqslant k \leqslant p, \quad P(x_k^2) = y_k.$$

6. d. Comme $P \in\in \mathbb{R}_p[X]$, il existe des réels $a_0, \dots, a_p$ tels que

$$P = \sum_{k=0}^{p} a_k X^k.$$

On pose alors

$$Q = P(X^2) = \sum_{k=0}^{p} a_k (X^2)^k = \sum_{k=0}^{p} a_k X^{2k} \in \mathbb{R}_{2p}[X]$$

et comme $n = 2 + 1$, on en déduit que ce polynôme Q appartient bien à $\mathbb{R}_n[X]$.

❧ Par **[6.c.]**,

$$\forall\, 0 \leqslant k \leqslant p, \quad Q(x_k) = P(x_k^2) = y_k.$$

Par ailleurs, toujours pour $0 \leqslant k \leqslant p$,

$$Q(x_{n-k}) = Q(-x_k) = P\big((-x_k)^2\big) = P(x_k^2) = y_k \overset{[6.b.]}{=} y_{n-k}.$$

Lorsque k varie de 0 à p (en croissant), l'indice $(n-k)$ varie (en décroissant) de n à $n - p = (2p+1) - p = p + 1$.

On a donc démontré que

$$\forall\, 0 \leqslant k \leqslant n, \quad Q(x_k) = y_k.$$

❧ On déduit de **[3.]** que $L = \mathscr{L}(P) = Q$.

❧ La formule développée de Q nous montre que L est bien un polynôme pair et que le degré de L est inférieur à $2p$ et donc strictement inférieur à $n = 2p + 1$.

Partie B. Évaluation du polynôme interpolateur

7. ❧ Comme les $(n+1)$ abscisses $b, x_1, \dots, x_n$ sont deux à deux distinctes, le polynôme interpolateur P existe bien et son degré est inférieur à n **[3.]**.

De même, le polynôme interpolateur Q existe bien (son degré est inférieur à n).

On en déduit que

$$\deg(X - a)P \leqslant n + 1 \quad \text{et que} \quad \deg(X - b)Q \leqslant n + 1,$$

donc $\deg S \leqslant n + 1$.

❧ Il est clair que

$$S(a) = \frac{(a-a)P(a) - (a-b)Q(a)}{b-a} = Q(a)$$

et que

$$S(b) = \frac{(b-a)P(b) - (b-b)Q(b)}{b-a} = P(b).$$

Mais f coïncide avec P en b et avec Q en a, donc

$$S(a) = f(a) \quad \text{et} \quad S(b) = f(b).$$

Les deux polynômes P et Q coïncident avec f aux abscisses x_k ($1 \leqslant k \leqslant n$), donc

$$S(x_k) = \frac{(x_k - a)P(x_k) - (x_k - b)Q(x_k)}{b-a} = \frac{(x_k - a)f(x_k) - (x_k - b)f(x_k)}{b-a} = f(x_k).$$

❧ Comme S et f coïncident aux $(n+2)$ abscisses a, b, x_1, …, x_n et que $\deg S \leqslant n+1$, on peut conclure avec **[3.c.]** : le polynôme S est le polynôme interpolateur de f aux points a, b, x_1, …, x_n.

8. a. Bienvenue dans le monde merveilleux des *tranches* !

❧ La liste $(x_0, \dots, x_{n-1})$ contient tous les éléments *sauf le dernier* et la liste $(x_1, \dots, x_n)$ contient tous les éléments *à partir de l'indice* 1. Traduction numpy :

$$\texttt{X[:-1]} = (x_0, x_1, x_2, \dots, x_{n-1})$$
$$\texttt{X[1:]} = (x_1, x_2, \dots, x_{n-1}, x_n)$$

❧ L'addition et la soustraction agissent sur les tableaux numpy comme sur les vecteurs de $\mathbb{R}^d$. Donc

$$\texttt{X[1:]-X[:-1]} = (x_1 - x_0, x_2 - x_1, \dots, x_n - x_{n-1}).$$

8. b. ❧ Le dernier élément de la liste

$$(y_k - y_0, y_{k+1} - y_1, y_{k+2} - y_2, \dots)$$

est de la forme $y_p - y_m$ avec

$$(p - m) = k - 0 = (k+1) - 1 = (k+2) - 2 = \cdots$$

donc $m = p - k$. On considère donc ici la liste

$$(y_k - y_0, \dots, y_p - y_{p-k}) = (y_k, \dots, y_p) - (y_0, \dots, y_{p-k}).$$

Au second membre,

— La première liste contient tous les éléments *à partir de l'indice* k

— et la deuxième liste contient tous les éléments *sauf les* k *derniers*.

Donc

$$(y_k - y_0, \dots, y_p - y_{p-k}) = \texttt{Y[:-k]-Y[k:]}.$$

❧ Avec deux tableaux numpy de même taille

$$\texttt{A} = (a_0, \dots, a_m)$$
$$\texttt{B} = (b_0, \dots, b_m)$$

on peut effectuer le produit terme à terme

$$\texttt{A*B} = (a_0 b_0, \dots, a_m b_m)$$

et, seulement si tous les b_k sont différents de 0, le quotient terme à terme

$$\texttt{A/B} = \Big(\frac{a_0}{b_0}, \dots, \frac{a_m}{b_m}\Big).$$

On dispose ici de la liste

$$\texttt{X[1:]-X[:-1]} = (x_1 - x_0, \dots, x_n - x_{n-1})$$

et de la liste

$$\texttt{Y[k:]-Y[:-k]} = (y_k - y_0, \dots, y_p - y_{p-k}).$$

Ces deux listes ont même taille si, et seulement si,

$$n - 1 = p - k$$

et, dans ce cas, la liste

$$\Big(\frac{x_1 - x_0}{y_k - y_0}, \dots, \frac{x_{p-(k-1)} - x_{p-k}}{y_p - y_{p-k}}\Big)$$

est représentée par

```
(X[1:]-X[:-1])/(Y[k:]-Y[:-k]).
```

9. Il est clair que

$$\forall\, 0 \leqslant k \leqslant n, \quad \deg T_k = k$$

donc la famille $(T_k)_{0 \leqslant k \leqslant n}$ est une base de $\mathbb{R}_n[X]$ (échelonnée en degré).

10. Le premier indice, i, doit être au moins égal à 0 et le dernier indice, $i + (k+1)$, doit être au plus égal à n, donc l'expression

$$f_{k+1}[x_i, \dots, x_{i+(k+1)}]$$

est définie pour tout indice

$$i \in [\![0, n - (k+1)]\!] = [\![0, n - k[\![.$$

11. On a justifié à la question précédente qu'il y avait $(n - k)$ valeurs possibles pour f_{k+1}.

On commence avec $n+1$ valeurs pour f_0 (il s'agit de l'échantillon des valeurs de la fonction f), puis seulement n valeurs pour f_1 (ce sont les taux d'accroissement)... Il n'y a donc aucune valeur pour $k = n$ et la dernière valeur calculée correspond donc à l'unique valeur de f_n : il s'agit de

$$f_n[x_0, \dots, x_n].$$

12. a. Nous allons procéder par récurrence sur n !

Initialisation.

Pour $n = 0$, on interpole sur un point x_0. D'après **[4.b.]**,

$$L = f(x_0) = f_0[x_0].$$

Itération.

On suppose que, pour un certain entier $n \in \mathbb{N}$, en interpolant sur $(n+1)$ points x_0, …, x_n, le coefficient du terme de degré n de $\mathscr{L}(f)$ est égal à $f_n[x_0, \dots, x_n]$.

NB : on sous-entend ici que les points d'interpolation peuvent être choisis de manière absolument quelconque (la seule restriction étant qu'ils doivent être deux à deux distincts).

Interpolons maintenant sur $(n+2)$ points arbitrairement choisis (deux à deux distincts néanmoins) :

$$x_0, \dots, x_n, x_{n+1}$$

et appliquons le résultat du **[7.]** : on note P et Q, les deux polynômes qui interpolent f aux $(n+1)$ points

$$b = x_0, x_1 \dots, x_n \quad \text{et} \quad x_1, \dots, x_n, a = x_{n+1}$$

respectivement. En appliquant l'hypothèse de récurrence à ces deux polynômes,

— le coefficient du terme de degré n de P est égal à

$$f_n[x_0, \dots, x_n],$$

— le coefficient du terme de degré n de Q est égal à

$$f_n[x_1, \dots, x_{n+1}].$$

D'après **[7.]**, le polynôme L qui interpole f aux $(n+2)$ points $b = x_0$, …, x_n, $a = x_{n+1}$ est égal à

$$\frac{(X - x_{n+1})P - (X - x_0)Q}{x_0 - x_{n+1}}$$

donc le coefficient du terme de degré $(n+1)$ de L est égal à

$$\frac{f_n[x_0, \dots, x_n] - f_n[x_1, \dots, x_{n+1}]}{x_0 - x_{n+1}},$$

c'est-à-dire à

$$f_{n+1}[x_0, \dots, x_{n+1}].$$

REMARQUE.— On a pris $i = 0$ et $k = n$ dans la relation de récurrence qui définit les différences divisées.

12. b. On procède à nouveau par récurrence sur $n \in \mathbb{N}$.

Contrairement à la précédente démonstration par récurrence, l'hypothèse de récurrence ne porte pas sur toutes les familles $(x_k)_{0 \leqslant k \leqslant n}$ de $(n+1)$ abscisses deux à deux distinctes, une seule famille suffit !

On considère donc une suite $(x_k)_{k \in \mathbb{N}}$ d'abscisses deux à deux distinctes et, pour tout $n \in \mathbb{N}$, on note L_n, le polynôme de $\mathbb{R}_n[X]$ qui interpole f aux $(n+1)$ premières abscisses x_0, …, x_n.

Initialisation.

Pour $n = 0$, en interpolant en un seul point x_0, on a vu à la question précédente que

$$\forall\, t \in \mathbb{R}, \quad L_0(t) = f(x_0) = f_0[x_0]$$

et donc que

$$L_0(t) = \sum_{k=0}^{n} f_k[x_0, \dots, x_k] T_k(t).$$

Itération.

On suppose que, pour un certain entier $n \in \mathbb{N}$,

$$\forall\, t \in \mathbb{R}, \quad L_n(t) = \sum_{k=0}^{n} f_k[x_0, \dots, x_k] T_k(t).$$

On considère alors le polynôme

$$P = L_{n+1} - f_{n+1}[x_0, \dots, x_{n+1}] T_{n+1}.$$

Comme L_{n+1} et T_{n+1} sont tous les deux des polynômes dont le degré est inférieur à $(n+1)$, on sait déjà que $P \in \mathbb{R}_{n+1}[X]$. De plus, d'après **[12.a.]**, le coefficient du terme de degré $(n+1)$ de L_{n+1} est égal à

$$f_{n+1}[x_0, \dots, x_{n+1}]$$

et le polynôme T_{n+1} est unitaire de degré $(n+1)$ — ça se voit clairement sur la définition —, donc le coefficient du terme de degré $(n+1)$ de P est nul et par conséquent

$$P \in \mathbb{R}_n[X].$$

Pour $0 \leqslant k \leqslant n$,

$$\begin{aligned} P(x_k) &= L_{n+1}(x_k) - f_{n+1}[x_0, \dots, x_{n+1}] T_{n+1}(x_k) \\ &= f(x_k) - 0 = f(x_k) \end{aligned}$$

car L_{n+1} interpole f aux points x_0, …, x_{n+1} et ces x_k sont racines évidentes de T_{n+1} (attention, x_{n+1} n'est pas racine de T_{n+1}).

Bref : $P \in \mathbb{R}_n[X]$ interpole f aux points x_0, …, x_n et donc, par hypothèse de récurrence,

$$P = \sum_{k=0}^{n} f_k[x_0, \dots, x_k] T_k.$$

On en déduit que

$$L_{n+1} = \sum_{k=0}^{n+1} f_k[x_0, \dots, x_k] T_k$$

et la démonstration est complète.

13. a. Le principe est simple : on calcule successivement les listes

$$\begin{aligned} F_0 &= (f_0[x_0], f_0[x_1], \dots, f_0[x_n]) \in \mathbb{R}^{n+1} \\ F_1 &= (f_1[x_0, x_1], \dots, f_1[x_{n-1}, x_n]) \in \mathbb{R}^n \\ &\ \vdots \\ F_{n-1} &= (f_{n-1}[x_0, \dots, x_{n-1}], f_{n-1}[x_1, \dots, x_n]) \in \mathbb{R}^2 \\ F_n &= (f_n[x_0, x_1, \dots, x_n]) \in \mathbb{R} \end{aligned}$$

et on récupère le premier élément de chaque liste.

```
def differences_divisees(X, Y):
  N = len(X)   # N = n+1
  F = Y[:]     # F_0
  D = [F[0]]
  for k in range(1, N):
    # F_k en fonction de F_{k-1}
    F = (F[:-1]-F[1:])/(X[:-k]-X[k:])
    D.append(F[0])
  return np.array(D)
```

13. b. La somme se calcule de proche en proche avec la relation suivante.

$$\sum_{k=0}^{d+1} f_k[\dots] T_k(t) = f_{d+1}[\dots] T_{d+1}(t) + \sum_{k=0}^{d} f_k[\dots] T_k(t)$$

Pour limiter au minimum le nombre de calculs effectués, il faut se rappeler la relation

$$T_{d+1}(t) = (t - x_d) T_d(t).$$

```
def Lagrange(X, D, t):
  L, Tk = D[0], 1
  for k, fk in enumerate(D[1:]):
    Tk *= (t-X[k])
    L += fk*Tk
  return L
```

13. c. Pour calculer

$$L(t) = \sum_{k=0}^{n} f_k[x_0, \dots, x_k] T_k(t)$$

une fois que la liste des $f_k[x_0, \dots, x_k]$ est calculée, il reste à calculer

— les $T_k(t)$: n différences $(t - x_k)$ et n produits (avec la formule de récurrence ;

— les n produits $f_k[x_0, \dots, x_k]T_k(t)$ (pour $k = 0$, il n'est pas nécessaire de calculer le produit !)

— la somme de $(n+1)$ termes.

Autrement dit, chaque évaluation de $L(t)$ coûte $\mathcal{O}(n)$ multiplications (on ne compte pas les additions, qui sont sensiblement moins coûteuses que les multiplications).

Auparavant, il faut calculer les $f_k[x_0, \dots, x_k]$. Pour cela, on calcule successivement

— la liste F_1 : $2n$ différences et n quotients ;

— la liste F_2 : $2(n-1)$ différences et $(n-1)$ quotients ;

— ... la liste F_{k+1} : $2(n-k)$ différences et $(n-k)$ quotients ;

— ... la liste F_n : deux différences et un quotient.

Le coût global pour le calcul de la liste D est donc $\mathcal{O}(n^2)$ multiplications.

Mais il faut être bien conscient que le calcul de D est fait une fois pour toutes. Pour évaluer N fois le polynôme L (en N valeurs de t), le nombre de multiplications à effectuer sera donc de l'ordre de

$$\mathcal{O}(N.n) + \mathcal{O}(n^2).$$

❧ Si on utilise naïvement la formule théorique :

$$L(t) = \sum_{k=0}^{n} y_k L_k(t),$$

il faut effectuer n multiplications (et n additions) après avoir évalué les $L_k(t)$.

Pour évaluer $L_k(t)$, on peut calculer d'abord le dénominateur (qui ne dépend pas de t) et évaluer le numérateur (pour chaque valeur de t) : chaque numérateur requiert environ n multiplications et comme il y a $(n+1)$ numérateurs à évaluer, il faut environ n^2 multiplications pour chaque évaluation de $L(t)$.

Autrement dit, pour évaluer N valeurs de L, le nombre de multiplications à effectuer sera de l'ordre de

$$\mathcal{O}(N.n^2).$$

❧ En conclusion, pour évaluer *une* valeur de $L(t)$, l'algorithme naïf vaut bien l'algorithme des différences divisées. En revanche, si on veut tracer le graphe de L et qu'on cherche à calculer un "grand" nombre N de valeurs de L, l'algorithme naïf n'est vraiment pas un bon choix !

L'interpolation polynomiale consiste à trouver une fonction polynomiale qui prend des valeurs données en des points donnés.

Si les données sont des points du graphe d'une fonction f :

$$\forall\, 0 \leqslant k \leqslant n, \qquad (x_k, y_k) = \big(x_k, f(x_k)\big)$$

on peut légitimement se demander si le polynôme interpolateur L constitue aussi une bonne approximation de f.

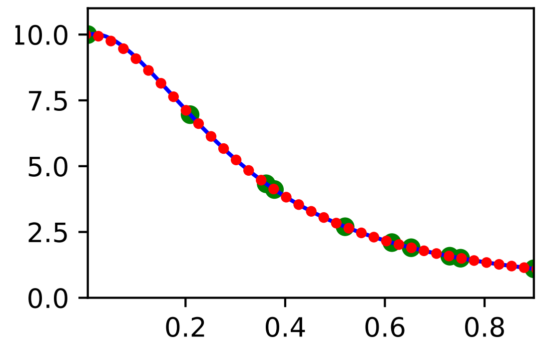
C'est souvent le cas, mais pas toujours ! L'étude théorique est un peu délicate, mais on peut l'aborder dans le cadre du programme.

De manière plus élémentaire, on peut multiplier les tentatives avec une fonction telle que

$$\forall\, x \in [0,1], \quad f(x) = \frac{1}{\frac{1}{10} + x^2}$$

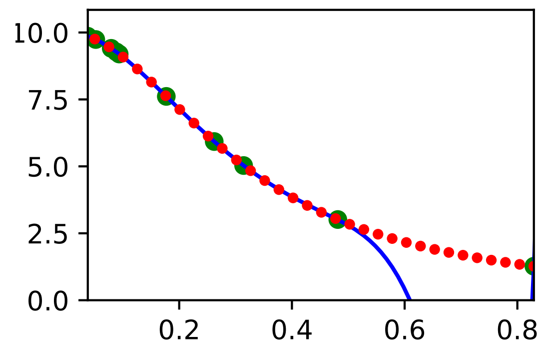
et des abscisses x_k choisies au hasard dans $[0,1]$.

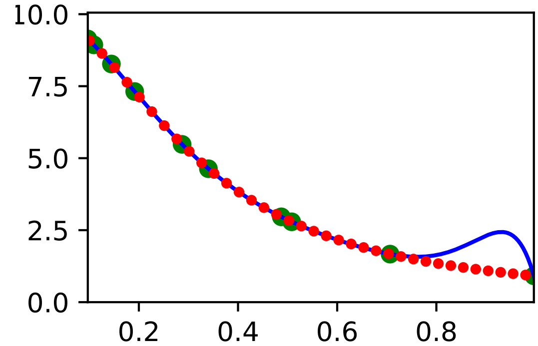
La plupart du temps, on obtient une figure comme celle-ci.

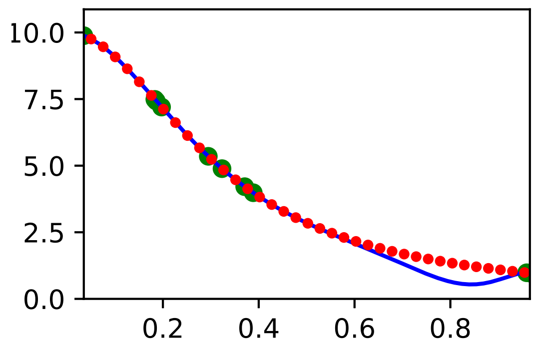


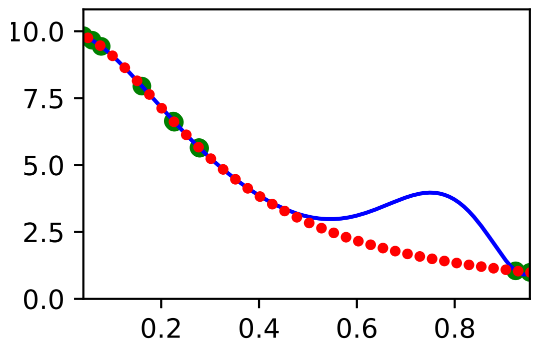
Les gros points verts sont les données, c'est-à-dire les points d'interpolation ; les points rouges sont situés sur le graphe de la fonction f et la courbe bleue est le graphe du polynôme interpolateur.

Mais parfois (environ une fois sur vingt ou trente), on obtient une figure où le polynôme interpolateur s'éloigne nettement de la fonction f.






10.0
7.5
5.0
2.5
0.0
0.2
0.4
0.6
0.8


10.0
7.5
5.0
2.5
0.0
0.2
0.4
0.6
0.8