

## DS 4 - sujet A

### THÈME : ALGÈBRE BILINÉAIRE

*La présentation, la lisibilité, l'orthographe, la qualité de la rédaction, la clarté et la précision des raisonnements entreront pour une part importante dans l'appréciation des copies. Les candidats sont invités à encadrer dans la mesure du possible les résultats de leurs calculs. Ils ne doivent faire usage d'aucun document : l'utilisation de toute calculatrice et de tout matériel électronique est interdite.*

#### Exercice 1 : Méthode de la puissance itérée

Soit  $n \in \mathbb{N} \setminus \{0; 1\}$ . Dans la suite, on considère  $E = \mathcal{M}_{n,1}(\mathbb{R})$  muni du produit scalaire canonique

$$\forall X, Y \in E, \quad \langle X, Y \rangle = {}^tXY.$$

On note aussi  $\|\cdot\|$ , la norme associée à ce produit scalaire.

Soit  $A$  une matrice de  $\mathcal{M}_n(\mathbb{R})$  inversible et diagonalisable avec exactement  $n$  valeurs propres  $\lambda_1, \dots, \lambda_n$ . On suppose les valeurs propres ordonnées de sorte que

$$|\lambda_1| < |\lambda_2| < \dots < |\lambda_n|.$$

Le but de cet exercice est de déterminer une approximation des vecteurs propres de  $A$  associés aux deux plus grandes valeurs propres (en valeur absolue). Pour cela, on définit la suite de vecteurs  $(X_k)_{k \in \mathbb{N}}$  par

$$X_0 \in \mathcal{M}_{n,1}(\mathbb{R}) \setminus \{0_{n,1}\} \quad \text{et} \quad \forall k \in \mathbb{N}, \quad X_{k+1} = \frac{AX_k}{\|AX_k\|}$$

On admet que la suite  $(X_k)_{k \in \mathbb{N}}$  est bien définie et converge vers un vecteur propre associé à  $\lambda_n$ , c'est-à-dire que, pour tout indice  $i \in \llbracket 1; n \rrbracket$ , la suite dont le terme général est la  $i$ -ème coordonnée de  $X_k$  converge.

- **Méthode de la puissance itérée**

1. Écrire une fonction `Scal` qui prend en argument  $X, Y \in E$  et renvoie le produit scalaire  $\langle X, Y \rangle$ . En déduire une seconde fonction `norme` qui prend en argument  $X \in E$  et renvoie sa norme  $\|X\|$ .
2. Écrire une fonction d'entête `puissance(A, X0, k)` qui renvoie le terme  $X_k$ .

- **La plus petite valeur propre**

3.
  - a) Rappeler le lien entre le spectre de  $A$  et celui de  $A^{-1}$  ?
  - b) En déduire un programme qui donne une approximation d'un vecteur propre d'une matrice associée à  $\lambda_1$ , la plus petite valeur propre de  $A$  en valeur absolue.

- **Test**

4. Voici un premier test, est-ce que le résultat obtenu est cohérent ?

Editeur

```
X0=np.array([[1],[0]])
A=np.array([[7,-10],[3,-4]])
Xt=puissance(X0,A,50)
```

Console

```
>>> Xt
array([[0.89442719],
       [0.4472136 ]])

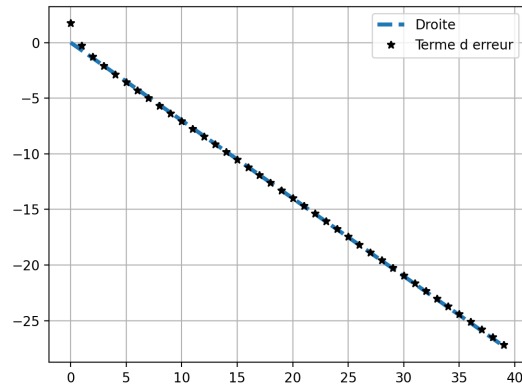
>>> 2*Xt-np.dot(A,Xt)
array([[ -2.22044605e-15],
       [-1.55431223e-15]])
```

On pourra calculer rapidement le spectre de  $A$  ainsi que l'espace propre associé à la plus grande valeur propre.

5. Que testent les lignes de commandes suivantes ?

Editeur

```
m=40
E=np.zeros(m)
for i in range(m):
    X=puissance(X0,A,i)
    e=norme(2*X-A@X)
    E[i]=np.log(e)
absc=np.arange(0,m,1)
plt.plot(absc,-0.7*absc,'--',
         label='Droite')
plt.plot(absc,E,'*',color='black',
         label='Terme d erreur')
plt.legend(), plt.grid(), plt.show()
```



• **Méthode de déflation**

6. Justifier que  $A$  et  ${}^tA$  ont même spectre.

Soient  $(X_i)_{i \in \llbracket 1; n \rrbracket}, (Y_i)_{i \in \llbracket 1; n \rrbracket}$  une base de vecteurs propres de  $A$  et  ${}^tA$  telles que

$$\forall i \in \llbracket 1; n \rrbracket, \quad AX_i = \lambda_i X_i \quad \text{et} \quad {}^tA Y_i = \lambda_i Y_i.$$

7. À l'aide de  $\langle AX_i, Y_n \rangle$ , justifier que  $Y_n \in \text{Vect}(X_1, \dots, X_{n-1})^\perp$ .

8. En raisonnant par l'absurde, en déduire que  $\langle X_n, Y_n \rangle \neq 0$ .

9. On définit la matrice  $B \in \mathcal{M}_n(\mathbb{R})$  par  $B = A - \frac{\lambda_n}{\langle X_n, Y_n \rangle} X_n {}^t Y_n$ .

Vérifier que  $A$  et  $B$  ont une base commune de vecteurs propres et que  $\text{Sp}(B) = \{\lambda_1, \dots, \lambda_{n-1}, 0\}$ .

10. En déduire un programme Python qui prend en argument la matrice  $A$  puis renvoie une approximation d'un vecteur propre pour la seconde valeur propre.

**Problème : Exemple de polynômes orthogonaux**  
**Polynômes de Laguerre**

Dans tout l'exercice, on considère un entier  $n \in \mathbb{N}^*$ .

**Partie I - Produit scalaire sur  $\mathbb{R}_n[x]$**

• **Généralités**

Pour tout couple  $(P, Q) \in \mathbb{R}_n[x]^2$ , on note :

$$\langle P, Q \rangle = \int_0^{+\infty} P(t)Q(t)e^{-t} dt.$$

11. Justifier que l'intégrale définissant  $\langle P, Q \rangle$  est convergente.

12. Montrer que l'application  $\langle \cdot, \cdot \rangle : \mathbb{R}_n[x] \times \mathbb{R}_n[x] \rightarrow \mathbb{R}$  est un produit scalaire.

• **Calcul d'un produit scalaire**

13. a) Soit  $k \in \llbracket 1; n \rrbracket$ . Établir que :

$$\int_0^{+\infty} t^k e^{-t} dt = k \int_0^{+\infty} t^{k-1} e^{-t} dt$$

b) Conclure que  $\langle x^k, 1 \rangle = k!$  pour tout entier  $k \in \llbracket 0, n \rrbracket$ .

**Partie II - Construction d'une base orthogonale**

On considère l'endomorphisme  $\alpha$  définie sur  $\mathbb{R}_n[x]$  par :

$$\forall P \in \mathbb{R}_n[x], \quad \alpha(P) = xP''(x) + (1-x)P'(x).$$

• **Propriétés de l'application  $\alpha$**

14. Exprimer la matrice de  $\alpha$  dans la base canonique de  $\mathbb{R}_n[x]$ .

15. En déduire que  $\alpha$  est diagonalisable et préciser son spectre.

• **Vecteurs propres de l'application  $\alpha$**

On fixe un entier  $k \in \llbracket 0, n \rrbracket$ .

16. a) Quelle est la dimension de  $\ker(\alpha + k \text{Id}_{\mathbb{R}_n[x]})$ ?  
 b) En déduire qu'il existe un unique polynôme  $P_k \in \mathbb{R}_n[x]$ , de coefficient dominant égal à 1, vérifiant  $\alpha(P_k) = -kP_k$ .
17. Justifier que  $P_k$  est de degré  $k$ .
18. Déterminer  $P_0$  et  $P_1$ . Vérifier que  $P_2(x) = x^2 - 4x + 2$ .

• **Orthogonalité de la famille  $(P_0, \dots, P_n)$**

On fixe un couple  $(P, Q) \in \mathbb{R}_n[x]^2$ .

19. a) Montrer que  $\langle \alpha(P), Q \rangle = - \int_0^{+\infty} tP'(t)Q'(t)e^{-t} dt$ .  
 b) En déduire que  $\langle \alpha(P), Q \rangle = \langle P, \alpha(Q) \rangle$ .
20. Montrer que  $(P_0, \dots, P_n)$  est une base orthogonale de  $\mathbb{R}_n[x]$ . Comment en déduire une base orthonormée de  $\mathbb{R}_n[x]$ ?

**Partie III - Méthode de quadrature de Gauss**

On admet que le polynôme  $P_n$  admet  $n$  racines réelles distinctes que l'on note  $x_1, \dots, x_n$ . On souhaite montrer qu'il existe  $(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^n$  tel que :

$$\forall P \in \mathbb{R}_{n-1}[x], \int_0^{+\infty} P(t)e^{-t} dt = \sum_{k=1}^n \lambda_k P(x_k) \quad (*)$$

21. Montrer qu'un  $n$ -uplet  $(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^n$  vérifie (\*) si et seulement si

$$[\lambda_1 \quad \lambda_2 \quad \dots \quad \lambda_n] \begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \dots & x_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^{n-1} \end{bmatrix} = [0! \quad 1! \quad \dots \quad (n-1)!].$$

22. a) On définit l'application linéaire

$$\varphi : P \in \mathbb{R}_{n-1}[x] \mapsto (P(x_1), P(x_2), \dots, P(x_n)) \in \mathbb{R}^n.$$

Vérifier que  $\varphi$  est un isomorphisme et expliciter sa matrice dans les bases canoniques de  $\mathbb{R}_{n-1}[x]$  et  $\mathbb{R}^n$  à l'aide de la matrice carrée précédente.

- b) En déduire qu'il existe un unique  $n$ -uplet  $(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^n$  vérifiant (\*).

23. En utilisant la division euclidienne par  $Q(x) = \prod_{i=1}^n (x - x_i)$ , montrer que la relation (\*) est vraie pour tout polynôme de  $\mathbb{R}_{2n-1}[x]$ .
24. Déterminer un polynôme  $P \in \mathbb{R}_{2n}[x]$  tel que

$$\int_0^{+\infty} P(t)e^{-t} dt \neq \sum_{k=1}^n \lambda_k P(x_k).$$

• **Python**

Soit  $f$ , une fonction continue sur  $\mathbb{R}^+$  telle que l'intégrale  $\int_0^{+\infty} f(t)e^{-t} dt$  soit convergente. On approxime cette intégrale par

$$\int_0^{+\infty} f(t)e^{-t} dt \approx \sum_{i=1}^n \lambda_i f(x_i).$$

25. Écrire un programme qui prend en argument  $n \in \mathbb{N}^*$  et renvoie la matrice ligne  $[0! \ 1! \ \dots \ (n-1)!]$ .
26. Écrire un programme python qui prend en argument une matrice ligne  $X = [x_1, \dots, x_n]$  et renvoie la matrice

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \dots & x_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^{n-1} \end{bmatrix}.$$

27. On admet qu'il existe une fonction racines qui prend en argument  $n$  et renvoie les racines de  $P_n$ . Construire un programme qui prend en argument une fonction  $f$ , un entier  $n$  et renvoie une approximation de l'intégrale généralisée  $\int_0^{+\infty} f(t)e^{-t} dt$  en utilisant les résultats précédents.

## Exercice 2

Soit  $n$  un entier supérieur ou égal à 2. On désigne par  $E$  un espace vectoriel euclidien de dimension  $n$ . Le produit scalaire de deux vecteurs  $x$  et  $y$  de  $E$  est noté  $\langle x, y \rangle$  et  $\|x\|$  représente la norme du vecteur  $x$ . Pour tout vecteur  $u$  non nul de  $E$ , on note  $\varphi_u$  l'application de  $E$  dans lui-même définie par :

$$\forall x \in E, \quad \varphi_u(x) = 2 \frac{\langle x, u \rangle}{\langle u, u \rangle} u - x.$$

- **Étude de l'application  $\varphi_u$**

28. Montrer que  $\varphi_u$  est un endomorphisme de  $E$ .

29. En calculant  $\varphi_u \circ \varphi_u$ , montrer que  $\varphi_u$  est un isomorphisme de  $E$  et déterminer  $\varphi_u^{-1}$ .

30. a) Soit  $x \in E$ , calculer  $\langle \varphi_u(x), \varphi_u(x) \rangle$ .

b) Justifier la formule de polarisation :

$$\forall x, y \in E, \quad \langle x, y \rangle = \frac{1}{4} (\|x + y\|^2 - \|x - y\|^2).$$

c) En déduire que  $\varphi_u$  conserve le produit scalaire, c'est-à-dire que :

$$\forall (x, y) \in E^2, \quad \langle \varphi_u(x), \varphi_u(y) \rangle = \langle x, y \rangle.$$

31. On note  $D_u$  la droite vectorielle dirigée par  $u$  et  $H_u = D_u^\perp$ .

Déterminer tous les espaces propres de  $\varphi$ . Est-ce que  $\varphi$  est diagonalisable?

- **Étude d'un exemple dans le cas  $n = 3$**

Soit  $H$  le sous-espace vectoriel de  $\mathbb{R}^3$  muni de sa structure euclidienne canonique et constitué des vecteurs  $X = (x, y, z)$  tels que  $x + y + z = 0$ . On note  $p$  le projecteur<sup>1</sup> sur  $H^\perp$  parallèlement à  $H$  et  $q$  le projecteur sur  $H$  parallèlement à  $H^\perp$ .

32. Expliciter trois vecteurs  $w_1, w_2, w_3$  constituant une base orthonormée de  $\mathbb{R}^3$  tels

$$H = \text{Vect}(w_1, w_2) \quad \text{et} \quad H^\perp = \text{Vect}(w_3).$$

33. a) Justifier que si  $X \in \mathbb{R}^3$ , alors  $p(X) = \langle X, w_3 \rangle w_3$ .

b) En déduire la matrice dans la base canonique de  $\mathbb{R}^3$  de  $p$  puis celle de  $q$ .

34. Soit  $v$  un vecteur unitaire de  $H^\perp$ . Exprimer  $\varphi_v$  à l'aide de  $p$  et  $q$ . En déduire la matrice de  $\varphi_v$  dans la base canonique de  $\mathbb{R}^3$ .

- **Étude d'une réciproque**

Soit  $\psi$  un endomorphisme de  $E$  tel qu'il existe une droite vectorielle  $\Delta$  de  $E$  vérifiant :

$$\forall x \in \Delta, \quad \psi(x) = x \quad \text{et} \quad \forall x \in \Delta^\perp, \quad \psi(x) = -x.$$

35. Montrer que  $\psi \circ \psi = \text{id}_E$  et que  $\psi$  conserve le produit scalaire.

36. Montrer qu'il existe au moins un vecteur  $u$  de  $E$  tel que  $\psi = \varphi_u$ .

### Aide Python

- `import numpy as np`

→ `np.array([[a1,1, ..., a1,p], [...], ..., [an,1, ..., an,p]])` pour la création de matrices et vecteurs.

→ Le produit matriciel  $AB$  s'obtient par la commande `np.dot(A, B)` ou `A@B`.

→ `np.arange(a, b, eps)` renvoie la liste des flottants de  $a$  à  $b$  (exclu) de pas constant `eps`.

→ `np.transpose(M)` renvoie la transposée de  $M$ .

→ `np.sum(M)` renvoie la somme de tous les éléments de  $M$ .

→ `np.cumsum(M)` renvoie la matrice ligne des sommes cumulées de tous les éléments de  $M$  lus dans le sens de lecture.

→ `np.prod(M)` renvoie le produit de tous les éléments de  $M$ .

→ `np.cumprod(M)` renvoie la matrice ligne des produits cumulés de tous les éléments de  $M$  lus dans le sens de lecture.

- `import numpy.linalg as al`

→ `al.inv(M)` renvoie l'inverse de  $M$ .

→ `al.matrix_rank(M)` renvoie le rang de  $M$ .

→ `al.solve(A, B)` renvoie la solution (si elle existe) de l'équation  $AX = B$  d'inconnue  $X$ .

– FIN –

---

1. Nous verrons que  $p$  est le projecteur orthogonal sur  $H^\perp$ .

## DS 4 - sujet \*

### THÈME : ALGÈBRE BILINÉAIRE

*La présentation, la lisibilité, l'orthographe, la qualité de la rédaction, la clarté et la précision des raisonnements entreront pour une part importante dans l'appréciation des copies. Les candidats sont invités à encadrer dans la mesure du possible les résultats de leurs calculs. Ils ne doivent faire usage d'aucun document : l'utilisation de toute calculatrice et de tout matériel électronique est interdite.*

#### Exercice 1 : Méthode de la puissance itérée

Soit  $n \in \mathbb{N} \setminus \{0; 1\}$ . Dans la suite, on considère  $E = \mathcal{M}_{n,1}(\mathbb{R})$  muni du produit scalaire canonique

$$\forall X, Y \in E, \quad \langle X, Y \rangle = {}^tXY.$$

On note aussi  $\|\cdot\|$ , la norme associée à ce produit scalaire.

Soit  $A$  une matrice de  $\mathcal{M}_n(\mathbb{R})$  inversible et diagonalisable avec exactement  $n$  valeurs propres  $\lambda_1, \dots, \lambda_n$ . On suppose les valeurs propres ordonnées de sorte que

$$|\lambda_1| < |\lambda_2| < \dots < |\lambda_n|.$$

Le but de cet exercice est de déterminer une approximation des vecteurs propres de  $A$  associés aux deux plus grandes valeurs propres (en valeur absolue). Pour cela, on définit la suite de vecteurs  $(X_k)_{k \in \mathbb{N}}$  par

$$X_0 \in \mathcal{M}_{n,1}(\mathbb{R}) \setminus \{0_{n,1}\} \quad \text{et} \quad \forall k \in \mathbb{N}, \quad X_{k+1} = \frac{AX_k}{\|AX_k\|}$$

On admet que la suite  $(X_k)_{k \in \mathbb{N}}$  est bien définie et converge vers un vecteur propre associé à  $\lambda_n$ , c'est-à-dire que, pour tout indice  $i \in \llbracket 1; n \rrbracket$ , la suite dont le terme général est la  $i$ -ème coordonnée de  $X_k$  converge.

- **Méthode de la puissance itérée**

1. Écrire une fonction `Scal` qui prend en argument  $X, Y \in E$  et renvoie le produit scalaire  $\langle X, Y \rangle$ . En déduire une seconde fonction `norme` qui prend en argument  $X \in E$  et renvoie sa norme  $\|X\|$ .
2. Écrire une fonction d'entête `puissance(A, X0, k)` qui renvoie le terme  $X_k$ .

- **La plus petite valeur propre**

3. En utilisant l'inverse de  $A$ , déduire un programme qui donne une approximation d'un vecteur propre d'une matrice associée à  $\lambda_1$ , la plus petite valeur propre de  $A$  en valeur absolue.

- **Test**

4. Voici un premier test, est-ce que le résultat obtenu est cohérent?

Editeur

```
X0=np.array([[1],[0]])
A=np.array([[7,-10],[3,-4]])
Xt=puissance(X0,A,50)
```

Console

```
>>> Xt
array([[0.89442719],
       [0.4472136 ]])

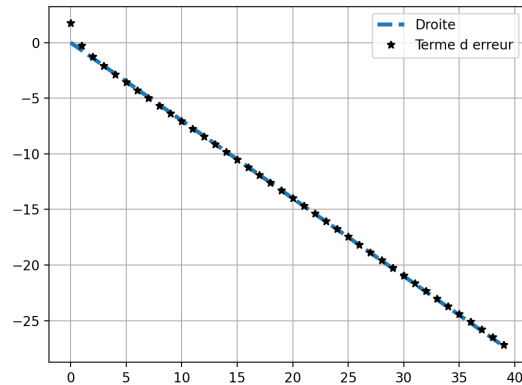
>>> 2*Xt-np.dot(A,Xt)
array([[ -2.22044605e-15],
       [-1.55431223e-15]])
```

On pourra calculer rapidement le spectre de  $A$  ainsi que l'espace propre associé à la plus grande valeur propre.

5. Que testent les lignes de commandes suivantes ?

Editeur

```
m=40
E=np.zeros(m)
for i in range(m):
    X=puissance(X0,A,i)
    e=norme(2*X-A@X)
    E[i]=np.log(e)
absc=np.arange(0,m,1)
plt.plot(absc,-0.7*absc,'--',
         label='Droite')
plt.plot(absc,E,'*',color='black',
         label='Terme d erreur')
plt.legend(), plt.grid(), plt.show()
```



• **Méthode de déflation**

Comme  $A$  et  ${}^tA$  ont même spectre, on peut considérer  $(X_i)_{i \in \llbracket 1;n \rrbracket}, (Y_i)_{i \in \llbracket 1;n \rrbracket}$  une base de vecteurs propres de  $A$  et  ${}^tA$  telles que

$$\forall i \in \llbracket 1;n \rrbracket, \quad AX_i = \lambda_i X_i \quad \text{et} \quad {}^tAY_i = \lambda_i Y_i.$$

- 6. À l'aide de  $\langle AX_i, Y_n \rangle$ , justifier que  $Y_n \in \text{Vect}(X_1, \dots, X_{n-1})^\perp$ .
- 7. En déduire que  $\langle X_n, Y_n \rangle \neq 0$ .

8. On définit la matrice  $B \in \mathcal{M}_n(\mathbb{R})$  par  $B = A - \frac{\lambda_n}{\langle X_n, Y_n \rangle} X_n {}^tY_n$ .

Justifier que  $\text{Sp}(B) = \{\lambda_1, \dots, \lambda_{n-1}, 0\}$ .

- 9. En déduire un programme Python qui prend en argument la matrice  $A$  puis renvoie une approximation d'un vecteur propre pour la seconde valeur propre.

**Problème : polynômes orthogonaux**

Le problème traite de quelques propriétés des polynômes de Hermite qui constituent une famille orthogonale pour un certain produit scalaire qui sera étudié dans ce problème.

On notera  $[x]$  la partie entière d'un réel  $x$  et  $\varphi$  la fonction définie sur  $\mathbb{R}$  par

$$\varphi(t) = e^{-t^2/2}.$$

On admet aussi pour tout  $n \in \mathbb{N}$ , la convergence et l'égalité

$$I_{2n} := \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} t^{2n} e^{-t^2/2} dt = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} t^{2n} \varphi(t) dt = \frac{(2n)!}{2^n n!}.$$

**Partie I : Préliminaires**

Deux résultats utiles par la suite.

• **Résultat 1**

Soit  $C$  un réel positif. Pour tout entier naturel  $n$  on pose

$$u_n = \frac{C^n}{\left[\frac{n}{2}\right]!}.$$

- 10. Déterminer  $\lim_{n \rightarrow +\infty} C^{2n}/n!$ . En déduire la limite de la suite  $(u_n)_{n \in \mathbb{N}}$  en considérant les suites  $(u_{2n})_{n \in \mathbb{N}}$  et  $(u_{2n+1})_{n \in \mathbb{N}}$ .
- 11. Montrer que la série de terme général  $u_{2k} + u_{2k+1}$  (où  $k \in \mathbb{N}$ ) converge et donner sa somme.
- 12. En déduire la convergence de la série de terme général  $u_n$  et la valeur de la somme  $\sum_{n=0}^{+\infty} u_n$ .

• **Résultat 2**

Soit  $a$  un réel strictement positif et soit  $g$  une fonction réelle indéfiniment dérivable sur  $[-a, a]$  pour laquelle existe un réel positif  $K$  tel que, pour tout entier  $n$  :

$$\max_{t \in [-a, a]} |g^{(n)}(t)| \leq \frac{K^n n!}{\left[\frac{n}{2}\right]!}$$

13. Montrer que pour tout  $\lambda \in [-a, a]$

$$\lim_{n \rightarrow +\infty} \int_0^\lambda \frac{(\lambda - t)^n}{n!} g^{(n+1)}(t) dt = 0.$$

14. En déduire l'égalité suivante, valable pour tout  $\lambda \in [-a, a]$  :

$$g(\lambda) = \sum_{k=0}^{+\infty} \frac{g^{(k)}(0)}{k!} \lambda^k.$$

### Partie II : Les polynômes de Hermite

15. Pour toute fonction polynomiale P, justifier la convergence de l'intégrale

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} P(t) e^{-\frac{t^2}{2}} dt.$$

16. Montrer que l'application  $(P, Q) \mapsto \langle P, Q \rangle = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} e^{-\frac{t^2}{2}} P(t)Q(t) dt$  est un produit scalaire sur  $\mathbb{R}[x]$ .

On notera  $\|\cdot\|$  la norme associée. Ainsi si  $n$  est un entier naturel, la restriction de ce produit scalaire aux polynômes de degré au plus  $n$  fait de  $(\mathbb{R}_n[x], \langle \cdot, \cdot \rangle)$  un espace euclidien.

17. À l'aide de la base canonique construire une base orthogonale de  $(\mathbb{R}_2[x], \langle \cdot, \cdot \rangle)$  formée de polynômes dont le coefficient de plus haut degré est 1.

Pour tout entier naturel  $n$  on considère l'application  $H_n$  de  $\mathbb{R}$  dans  $\mathbb{R}$  définie pour tout réel  $t$  par

$$H_n(t) = (-1)^n e^{\frac{t^2}{2}} \varphi^{(n)}(t)$$

où selon l'usage  $\varphi^{(n)}(t)$  désigne la valeur en  $t$  de la dérivée  $n$  ième de  $\varphi$  (en particulier  $\varphi^{(0)}(t) = \varphi(t)$ ).

18. Pour tout réel  $x$  calculer  $H_0(x), H_1(x), H_2(x), H_3(x)$ .

19. Soit  $n \in \mathbb{N}^*$ . Établir les relations

$$(\bullet) \quad H_{n+1}(x) = xH_n(x) - nH_{n-1}(x) \quad \text{et} \quad (\bullet\bullet) \quad H'_n = nH_{n-1}.$$

Pour établir  $(\bullet)$ , on pourra utiliser  $\varphi'(t) = -t\varphi(t)$ .

20. Montrer que, pour tout  $n \in \mathbb{N}$ ,  $H_n$  est une fonction polynomiale dont on précisera, en fonction de  $n$ , le degré, la parité et le coefficient de plus haut degré.

#### • Calcul des polynômes de Hermite - méthode 1

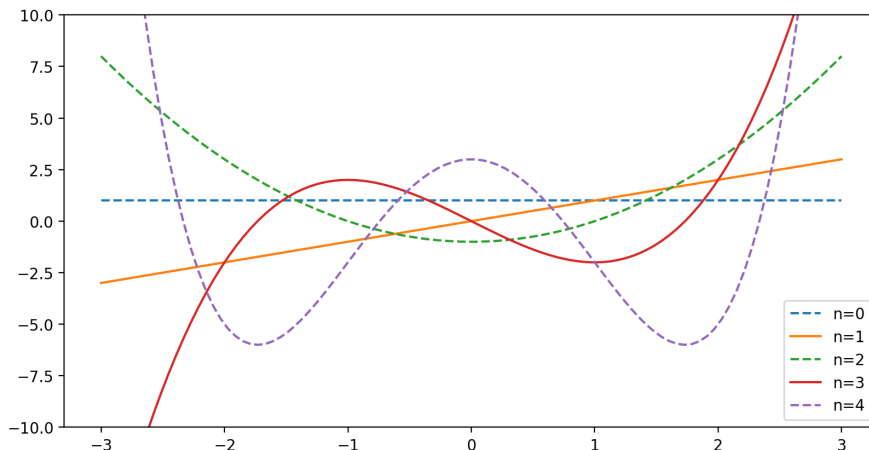
21. Pour tout  $n \in \mathbb{N}$ , on pose  $h_n = H_n(0)$ . Utiliser  $(\bullet)$  pour écrire un programme python `Hzero` qui prend en argument  $n$  et renvoie  $h_n(0)$ .

22. Tout polynôme  $P = \sum_{i=0}^d a_i x^i$  de degré au plus  $d$  peut être codé par une matrice ligne  $MP = [a_0 \ a_1 \ \dots \ a_d]$ . Écrire un programme python qui prend en argument une matrice ligne  $MP$  codant un polynôme  $P$  et qui renvoie la matrice ligne  $MQ$  codant le polynôme  $Q$  défini par

$$\forall x \in \mathbb{R}, \quad Q(x) = \int_0^x P(t) dt.$$

23. Déduire des deux questions précédentes, un programme python qui prend en argument  $n$  et renvoie la matrice ligne  $MH_n$  codant le polynôme  $H_n$ .

24. Comment tracer avec python la courbe représentative de  $H_n$  sur  $[-3;3]$ ? Ci-dessous, le résultat obtenu pour les 5 premiers polynômes. Que peut-on conjecturer sur le nombre de racines de  $H_n$ ?



• **Calcul des polynômes de Hermite - méthode 2**

25. En utilisant la relation (•), compléter ce programme qui prend en argument  $m$  et renvoie la matrice  $(M_{n,k})_{k,n \in \llbracket 0;n \rrbracket}$  dont le coefficient d'indice  $(n, k)$  est le coefficient du polynôme  $H_n$  devant  $x^k$ .

Editeur

```
(1) def Hmatrice(m):
(2)     H=np.zeros([m,m])
(3)     H[0,0]=...
(4)     H[1,1]=...
(5)     for n in range(1,m-1):
(6)         for k in range(m):
(7)             H[n+1,k]=...
(8)     return H
```

Console

```
>>> Hmatrice(5)
array([[ 1.,  0.,  0.,  0.,  0.],
       [ 0.,  1.,  0.,  0.,  0.],
       [-1.,  0.,  1.,  0.,  0.],
       [ 0., -3.,  0.,  1.,  0.],
       [ 3.,  0., -6.,  0.,  1.]])
```

**Partie III :  $(H_n)_{n \in \mathbb{N}}$  comme famille de polynômes orthogonaux**

26. Montrer que si  $P$  est un polynôme et  $n$  un entier naturel non nul alors  $\lim_{x \rightarrow +\infty} P(x)\varphi^{(n-1)}(x) = 0$ .

De même on montrerait et on admet que  $\lim_{x \rightarrow -\infty} P(x)\varphi^{(n-1)}(x) = 0$ .

27. Pour tout  $n \in \mathbb{N}$  calculer

$$\langle H_n, H_0 \rangle = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} H_n(t) e^{-\frac{t^2}{2}} dt.$$

Pour  $n$  non nul on utilisera la définition de  $H_n$ .

28. Soit  $(n, m) \in \mathbb{N}^* \times \mathbb{N}^*$ . En remarquant que

$$\langle H_n, H_m \rangle = \frac{(-1)^n}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} H_m(t) \varphi^{(n)}(t) dt$$

montrer que

$$\langle H_n, H_m \rangle = m \langle H_{n-1}, H_{m-1} \rangle.$$

29. En déduire que  $(H_n)_{n \in \mathbb{N}}$  est une famille orthogonale de  $\mathbb{R}[x]$ . Pour  $n \in \mathbb{N}$ , que vaut  $\langle H_n, H_n \rangle$ ?

**Partie IV : Un développement en série de Hermite**

Soit  $n$  un entier naturel non nul.

30. Soit  $P$  un polynôme de degré au plus  $n$ . Justifier l'égalité suivante :

$$P = \sum_{k=0}^n \langle P, H_k \rangle \frac{H_k}{k!}.$$

Pour tout couple  $(b, c)$  de réels vérifiant  $b \leq c$  on admet qu'il existe un réel  $K$  (dépendant de  $b$  et  $c$ ) tel que pour tout entier  $n$  et tout  $x \in [b, c]$  :

$$\left| \frac{H_n(x)}{n!} \right| \leq \frac{K^n}{\lfloor \frac{n}{2} \rfloor!}.$$

31. Soit  $x$  un réel donné. Établir, pour tout réel  $\lambda$ , la convergence de la série de terme général  $\frac{H_n(x)}{n!} \lambda^n$ .

32. Le réel  $x$  reste fixé. Soit  $g_x$  la fonction définie pour tout réel  $\lambda$  par

$$g_x(\lambda) = e^{-\frac{(\lambda-x)^2}{2}}.$$

a) Pour tout réel  $\lambda$  et tout entier naturel  $n$ , exprimer  $g_x^{(n)}(\lambda)$  en fonction de  $H_n$ .

b) Montrer que  $g_x$  vérifie les hypothèses du résultat 2 et en déduire que pour tout  $(x, \lambda) \in \mathbb{R}^2$

$$e^{\lambda x - \frac{\lambda^2}{2}} = \sum_{k=0}^{+\infty} \frac{H_k(x)}{k!} \lambda^k.$$

33. On note  $\exp$  la fonction  $t \in \mathbb{R} \mapsto e^t$ . Pour tout entier naturel  $n$  justifier rapidement la convergence de l'intégrale

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} e^t H_n(t) e^{-\frac{t^2}{2}} dt$$



dont, par analogie, on note  $\langle \exp, H_n \rangle$  la valeur. Calculer  $\langle \exp, H_n \rangle$  puis, pour tout réel  $x$ , conclure à l'égalité

$$\exp x = \sum_{n=0}^{+\infty} \langle \exp, H_n \rangle \frac{H_n(x)}{n!}.$$

Pour calculer  $\langle \exp, H_n \rangle$  on pourra utiliser la définition de  $H_n$  et intégrer par parties (avec soin) afin d'obtenir  $\langle \exp, H_n \rangle = \langle \exp, H_{n-1} \rangle$ .

### Partie V : Méthode de quadrature de Gauss

On admet que le polynôme  $H_n$  admet  $n$  racines réelles distinctes que l'on note  $x_1, \dots, x_n$ . On souhaite montrer qu'il existe  $(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^n$  tel que :

$$\forall P \in \mathbb{R}_{n-1}[x], \quad \int_{-\infty}^{+\infty} P(t)e^{-t^2/2} dt = \sum_{k=1}^n \lambda_k P(x_k) \quad (\star)$$

On définit alors pour tout entier naturel  $n$ ,

$$J_n = \int_{-\infty}^{+\infty} t^n e^{-t^2/2} dt.$$

34. Montrer qu'un  $n$ -uplet  $(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^n$  vérifie  $(\star)$  si et seulement si

$$\begin{bmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_n \\ \vdots & \vdots & & \vdots \\ x_1^{n-1} & x_2^{n-1} & \dots & x_n^{n-1} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} = \begin{bmatrix} J_0 \\ J_1 \\ \vdots \\ J_{n-1} \end{bmatrix}.$$

35. a) On définit l'application linéaire

$$\varphi : P \in \mathbb{R}_{n-1}[x] \mapsto (P(x_1), P(x_2), \dots, P(x_n)) \in \mathbb{R}^n.$$

Vérifier que  $\varphi$  est un isomorphisme et expliciter sa matrice dans les bases canoniques de  $\mathbb{R}_{n-1}[x]$  et  $\mathbb{R}^n$ .

b) En déduire qu'il existe un unique  $n$ -uplet  $(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^n$  vérifiant  $(\star)$ .

36. En utilisant la division euclidienne par  $Q(x) = \prod_{i=1}^n (x - x_i)$ , montrer que la relation  $(\star)$  est vraie pour tout polynôme de  $\mathbb{R}_{2n-1}[x]$ .

37. Déterminer un polynôme  $P \in \mathbb{R}_{2n}[x]$  tel que

$$\int_{-\infty}^{+\infty} P(t)e^{-t^2/2} dt \neq \sum_{k=1}^n \lambda_k P(x_k).$$

#### • Python

Soit  $f$ , une fonction continue sur  $\mathbb{R}^+$  telle que l'intégrale  $\int_{-\infty}^{+\infty} f(t)e^{-t^2/2} dt$  soit convergente. On approxime cette intégrale par

$$\int_{-\infty}^{+\infty} f(t)e^{-t^2/2} dt \approx \sum_{i=1}^n \lambda_i f(x_i).$$

38. Écrire un programme qui prend en argument  $n \in \mathbb{N}^*$  et renvoie la matrice ligne  $[J_0 \ J_1 \ \dots \ J_{n-1}]$ .

On pourra utiliser la fonction `np.cumprod`.

39. Écrire un programme python qui prend en argument une matrice ligne  $X = [x_1, \dots, x_n]$  et renvoie la matrice

$$\begin{bmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_n \\ \vdots & \vdots & & \vdots \\ x_1^{n-1} & x_2^{n-1} & \dots & x_n^{n-1} \end{bmatrix}.$$

40. On admet qu'il existe une fonction racines qui prend en argument  $n$  et renvoie les racines de  $H_n$ . Construire un programme qui prend en argument une fonction  $f$ , un entier  $n$  et renvoie une approximation de l'intégrale généralisée  $\int_{-\infty}^{+\infty} f(t)e^{-t^2/2} dt$  en utilisant les résultats précédents.

**Bonus** Extrait de la lettre de Hermite à Mittag-Leffler à propos des travaux présentés par Poincaré pour le prix du Roi de Suède.

« Le mémoire de M. Poincaré est d'une profondeur et d'une puissance d'invention bien rares, il fera certainement école dans la science tant au point de vue de l'analyse que des conséquences astronomiques. »

Mais de qui parle le mathématicien français Charles Hermite, de son ancien étudiant Henri Poincaré ou de son cousin Raymond Poincaré?

## Aide Python

- `import numpy as np`
- `np.array[[a1,1, ..., a1,p], [...], ..., [an,1, ..., an,p]]` pour la création de matrices et vecteurs.
- Le produit matriciel  $AB$  s'obtient par la commande `np.dot(A, B)` ou `A@B`.
- `np.arange(a, b, eps)` renvoie la liste des flottants de  $a$  à  $b$  (exclu) de pas constant `eps`.
- `np.transpose(M)` renvoie la transposée de  $M$ .
- `np.sum(M)` renvoie la somme de tous les éléments de  $M$ .
- `np.cumsum(M)` renvoie la matrice ligne des sommes cumulées de tous les éléments de  $M$  lus dans le sens de lecture.
- `np.prod(M)` renvoie le produit de tous les éléments de  $M$ .
- `np.cumprod(M)` renvoie la matrice ligne des produits cumulés de tous les éléments de  $M$  lus dans le sens de lecture.
  
- `import numpy.linalg as al`
- `al.inv(M)` renvoie l'inverse de  $M$ .
- `al.matrix_rank(M)` renvoie le rang de  $M$ .
- `al.solve(A, B)` renvoie la solution (si elle existe) de l'équation  $AX = B$  d'inconnue  $X$ .

– FIN –

DS 4A - solution

Exercice 1

1. Si les coefficients de X et Y sont respectivement donnés par  $x_i$  et  $y_i$ , on sait que

$$\langle X, Y \rangle = \sum_{i=1}^n x_i y_i \quad \text{et} \quad \|X\| = \sqrt{\langle X, X \rangle} = \sqrt{\sum_{i=1}^n x_i^2}.$$

On en déduit le programme :

```
def scal(X, Y):
    n=len(X)
    s=0
    for i in range(n):
        s+=X[i]*Y[i]
    return s

def norme(X):
    return scal(X, X)**(1/2)

# seconde solution
def normebis(X):
    n=len(X)
    s=0
    for i in range(n):
        s+=X[i]**2
    return s**(1/2)

# Un petit test
X=np.array([3,0,4])
print(norme(X))
5.0
```

2.

```
def puissance(X0, A, k):
    X=X0
    for i in range(k):
        X=np.dot(A, X)
        X=X/norme(X)
    return X
```

Précisons que A n'admet pas 0 comme valeur propre, ainsi le seul vecteur X vérifiant  $AX = 0$  est le vecteur nul. En partant de  $X_0 \neq 0$ , on s'assure qu'aucun terme de la suite  $(X_k)_k$  n'est nulle. Cette suite est bien définie.

3.a) Comme A est inversible,  $0 \notin \text{Sp}(A)$ . Soient  $X \in E \setminus \{0_E\}$  et  $\lambda \in \mathbb{R}$

$$AX = \lambda X$$

est équivalent à

$$\frac{1}{\lambda}X = A^{-1}X$$

Ainsi  $\lambda \in \text{Sp}(A) \iff \frac{1}{\lambda} \in \text{Sp}(A)$ .

3.b) Pour obtenir un tel vecteur propre, il suffit d'appliquer la méthode de la puissance itérée à  $A^{-1}$  car la plus grande valeur propre de  $A^{-1}$  devient la plus petite de A. Ainsi, un code possible est

```
import numpy.linalg as al

def puissance_min(X0, A, k):
    Ainvs=al.inv(A)
    return puissance(X0, Ainvs, k)
```

4. Le programme est testé sur la matrice

$$A = \begin{bmatrix} 7 & -10 \\ 3 & -4 \end{bmatrix}.$$

Déterminons le spectre de A. Soit  $\lambda \in \mathbb{R}$

$$\det(A - \lambda I_2) = (7 - \lambda)(-4 - \lambda) - (-10) \times 3 = \lambda^2 - 3\lambda + 2$$

$$\det(A - \lambda I_2) = (\lambda - 1)(\lambda - 2).$$

Ainsi  $\text{Sp}(A) = \{1, 2\}$ .

De plus  $X = \begin{bmatrix} x \\ y \end{bmatrix} \in E_2(A)$  si et seulement si

$$\begin{cases} 7x - 10y = 2x \\ 3x - 4y = 2y \end{cases} \iff x = 2y$$

Le résultat obtenu est cohérent car :

- $X_t$  est proche d'un vecteur propre de A pour  $\lambda_2 = 2$  car sa première composante est pratiquement le double de la première;
- On a  $2X_t \approx AX_t$  et 2 est bien la plus grande valeur propre.

5. Le programme calcule pour  $i \in [0; 39]$ , l'erreur

$$E_i = \ln(\|AX_i - 2X_i\|)$$

et stocke les résultat dans une matrice ligne. Les résultats obtenus donnent

$$E_i \approx \alpha i \quad \text{où} \quad \alpha = -0.7$$

c'est-à-dire  $\|AX_i - 2X_i\| \approx e^{-\alpha i}$ .

On obtient une décroissance très rapide. Au bout de quelques étapes, on a déjà une bonne approximation d'un vecteur propre pour la valeur propre  $\lambda_n$ .

6. Soit  $\lambda \in \mathbb{R}$ . On rappelle qu'une matrice est inversible si et seulement si sa transposée l'est.

$\lambda \in \text{Sp}(A)$  ssi  $A - \lambda I_n$  n'est pas inversible  
ssi  ${}^t(A - \lambda I_n)$  n'est pas inversible  
ssi  ${}^tA - \lambda I_n$  n'est pas inversible

ssi  $\lambda \in \text{Sp}({}^t A)$ .  
 D'où l'égalité :  $\text{Sp}(A) = \text{Sp}({}^t A)$ .

On aurait pu aussi utiliser l'invariance du rang par transposition.

7. D'une part,  $X_i$  est vecteur propre de  $A$  :

$$\langle AX_i, Y_n \rangle = \lambda_i \langle X_i, Y_n \rangle.$$

D'autre part,  $Y_n$  est vecteur propre de  ${}^t A$

$$\begin{aligned} \langle AX_i, Y_n \rangle &= {}^t (AX_i) Y_n = {}^t X_i {}^t A Y_n = \lambda_n {}^t X_i Y_n \\ &= \lambda_n \langle X_i, Y_n \rangle. \end{aligned}$$

Ainsi  $\lambda_i \langle X_i, Y_n \rangle = \lambda_n \langle X_i, Y_n \rangle$ .

D'où  $\langle X_i, Y_n \rangle = 0$  car  $\lambda_i \neq \lambda_n$ . Finalement

$$Y_n \in \text{Vect}(X_1, X_2, \dots, X_{n-1})^\perp.$$

8. Raisonnons par l'absurde en supposant que  $\langle X_n, Y_n \rangle = 0$ .  
 On en déduit alors que

$$\begin{aligned} Y_n &\in \text{Vect}(X_1, X_2, \dots, X_{n-1}, X_n)^\perp \\ &\in E^\perp \quad \text{car } (X_i)_i \text{ est une base de } E \\ Y_n &\in \{0_E\}. \end{aligned}$$

Absurde  $Y_n$  est un vecteur propre, il est non nul. Finalement

$$\langle X_n, Y_n \rangle \neq 0.$$

9. Soit  $i \in \llbracket 1; n-1 \rrbracket$ .

$$BX_i = AX_i - \frac{\lambda_n}{\langle X_n, Y_n \rangle} X_n {}^t Y_n X_i = AX_i = \lambda_i X_i.$$

Car  ${}^t Y_n X_i = \langle X_i, Y_n \rangle = 0$  (question 7). De plus

$$BX_n = AX_n - \lambda_n \frac{X_n {}^t Y_n X_n}{\langle X_n, Y_n \rangle} = AX_n - \lambda_n X_n = 0$$

car  $X_n$  est vecteur propre de  $A_n$  pour  $\lambda_n$ .

Finalement  $(X_1, X_2, \dots, X_n)$  est une base de vecteurs propres pour  $B$  et

$$\text{Sp}(B) = \{\lambda_1, \lambda_2, \dots, \lambda_{n-1}, 0\}.$$

10. On peut appliquer la méthode de la puissance pour obtenir une bonne approximation  $\tilde{\lambda}_n$  ainsi que des vecteurs propres  $\tilde{X}_n$  et  $\tilde{Y}_n$ . On calcule ensuite

$$\tilde{B} = A - \frac{\tilde{\lambda}_n}{\langle \tilde{X}_n, \tilde{Y}_n \rangle} \tilde{X}_n {}^t \tilde{Y}_n = A - \frac{1}{\langle \tilde{X}_n, \tilde{Y}_n \rangle} A \tilde{X}_n {}^t \tilde{Y}_n$$

et on ré-applique la méthode de la puissance à  $\tilde{B}$ . On obtient par exemple le code :

```
def puissance2(X0, A, k):
    X=puissance(X0, A, k)
    Y=puissance(X0, np.transpose(A), k)
    ps=scal(X, Y)
    B=A-(1/ps)*((A@X)@np.transpose(Y))
    return puissance(X0, B, k)
```

On peut tester les codes avec la matrice  $A$ . Cette dernière est de taille 2, donc la seconde valeur propre est aussi la plus petite :

```
>>> puissance2(X0, A, 20)
array([[ -0.85749285],
       [ -0.51449588]])
```

```
>>> puissance_min(X0, A, 20)
array([[ -0.85749288],
       [ -0.51449583]])
```

On obtient des valeurs proches.

### Problème

11. La fonction  $t \in \mathbb{R}^+ \mapsto P(t)Q(t)e^{-t}$  est continue sur  $\mathbb{R}^+$ . On a donc une intégrale généralisée en  $+\infty$ . Or :

- L'intégrale de Riemann  $\int_1^{+\infty} 1/t^2 dt$  est convergente;
- On a par les croissances comparées

$$P(t)Q(t)e^{-t} = o_{+\infty}\left(1/t^2\right);$$

- $1/t^2 \geq 0$  pour  $t \geq 1$ .

Par le critère de négligeabilité, l'intégrale

$$\int_0^{+\infty} P(t)Q(t)e^{-t} dt$$

est convergente.

12. — Symétrie

Soient  $P, Q \in E$ .

$$\begin{aligned} \langle P, Q \rangle &= \int_0^{+\infty} P(t)Q(t)e^{-t} dt \\ &= \int_0^{+\infty} Q(t)P(t)e^{-t} dt = \langle Q, P \rangle. \end{aligned}$$

→ Bilinearité

Justifions la linéarité à gauche. Soient  $\lambda \in \mathbb{R}$  et  $P, Q, R \in E$ , par linéarité de l'intégrale,

$$\begin{aligned} \langle \lambda P + Q, R \rangle &= \int_0^{+\infty} (\lambda P(t) + Q(t))R(t)e^{-t} dt \\ &= \lambda \int_0^{+\infty} P(t)R(t)e^{-t} dt + \int_0^{+\infty} Q(t)R(t)e^{-t} dt \\ &= \lambda \langle P, R \rangle + \langle Q, R \rangle. \end{aligned}$$

Par symétrie, on obtient la linéarité à droite et donc la bilinéarité.

→ Positivité

Pour  $P \in E$ . Par croissance de l'intégrale avec les bornes dans le bon sens

$$\langle P, P \rangle = \int_0^{+\infty} \underbrace{P(t)^2 e^{-t}}_{\geq 0} dt \geq 0.$$

→ Définie

De plus si  $\langle P, P \rangle = 0$ . C'est-à-dire

$$\int_0^{+\infty} P(t)^2 e^{-t} dt = 0.$$

On en déduit par positivité et continuité de l'intégrande

$$\forall t \in \mathbb{R}^+, \quad P(t)^2 e^{-t} = 0.$$

Le polynôme  $P$  admet une infinité de racines (tous les réels positifs) donc  $P$  est le polynôme nul.

**13.a)** Procéder à une intégration par parties.

**13.b)** La relation précédente donne

$$\forall i \in \llbracket 1; n \rrbracket, \quad \langle x^i, 1 \rangle = i \langle x^{i-1}, 1 \rangle.$$

Le résultat s'en déduit par récurrence ou en calculant le produit télescopique

$$\frac{\langle x^k, 1 \rangle}{\langle 1, 1 \rangle} = \prod_{i=1}^k \frac{\langle x^i, 1 \rangle}{\langle x^{i-1}, 1 \rangle} = \prod_{i=1}^k i = k!$$

**14.** On a

$$\alpha(1) = 0, \alpha(x) = 1 - x$$

et pour tout  $k \in \llbracket 2; n \rrbracket$

$$\alpha(x^k) = -kx^k + k^2x^{k-1}.$$

On en déduit que la matrice de  $\alpha$  dans la base canonique est

$$\begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 4 & \ddots & \vdots \\ \vdots & \ddots & -2 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & n^2 \\ 0 & \cdots & \cdots & 0 & -n \end{pmatrix}.$$

**15.** Cette matrice est triangulaire, le spectre se lit sur la diagonale

$$\text{Sp}(\alpha) = \{-k \mid k \in \llbracket 0; n \rrbracket\}.$$

Il y a  $\dim \mathbb{R}_n[x] = n + 1$  valeurs propres distinctes, on sait alors que l'endomorphisme est diagonalisable.

**16.a)** On sait que dans ce cas, les sous-espaces propres de  $\alpha$  sont de dimension 1.

**16.b)** — Existence

Soit  $Q_k$  un vecteur (non nul) de  $\ker(\alpha + k \text{id}_{\mathbb{R}_n[x]})$  et  $c_k$  son coefficient dominant (non nul). On peut donc poser

$$P_k = \frac{1}{c_k} Q_k \in \mathbb{R}_n[x]$$

qui est bien de coefficient dominant 1 et vecteur propre pour la valeur propre  $-k$ .

— Unicité

Soit  $R_k$  un polynôme vérifiant les mêmes propriétés. Comme le sous-espace propre est de dimension 1

$$R_k \in \ker(\alpha + k \text{id}_{\mathbb{R}_n[x]}) = \text{Vect}(P_k).$$

Il existe donc  $a \in \mathbb{R}$  tel que  $R_k = aP_k$ . Comme le coefficient dominant de  $R_k$  est égal à 1, on a nécessairement  $a = 1$  puis  $R_k = P_k$ .

On a donc bien existence et unicité d'un tel polynôme.

**17.** Soit  $d$  le degré de  $P_k$ . Il existe  $Q \in \mathbb{R}_{d-1}[x]$  tel que

$$P_k = x^d + Q(x)$$

puis par linéarité de  $\alpha$

$$\begin{aligned} \alpha(P_k) &= \alpha(x^d) + \alpha(Q) \\ &= (1-x)dx^{d-1} + xd(d-1)x^{d-1} + \alpha(Q) \\ -kP_k &= -dx^{d-1} + \underbrace{\cdots}_{\in \mathbb{R}_{d-1}[x]} \end{aligned}$$

En identifiant les coefficients dominants  $-k = -d$  donc  $d = k$ .

Pour  $k = 0$  ; on a  $\alpha(1) = 0$  et 1 est un polynôme de coefficient dominant 1 tel que  $\alpha(1) = -0 \times 1$  donc, par unicité,  $P_0 = 1$  qui est bien de degré 0.

**18.** D'après ce qui précède

$$P_0 = 1.$$

Soit  $P = x + a$  un polynôme de degré 1 et de coefficient dominant 1. On a  $\alpha(P) = 1 - x$  donc  $\alpha(P) = -P$  si et seulement si  $a = -1$ . Ainsi

$$P_1 = x - 1.$$

De même, on pose  $P = x^2 + bx + c$ . Dans ce cas, on vérifie que

$$\alpha(P) = -2x^2 + (4-b)x + b$$

donc  $\alpha(P) = -2P$  si et seulement si  $4-b = -2b$  et  $b = -2c$  d'où  $b = -4$  et  $c = 2$ . Finalement,

$$P_2 = x^2 - 4x + 2.$$

**19.a)** On a

$$\langle \alpha(P), Q \rangle = \int_0^{+\infty} (tP''(t) + (1-t)P'(t))Q(t)e^{-t} dt.$$

Soit  $A \in \mathbb{R}_*^+$  et calculons

$$\int_0^A (tP''(t) + (1-t)P'(t))Q(t)e^{-t} dt$$

via une intégration par parties avec les fonctions  $u$  et  $Q$  de classe  $\mathcal{C}^1$  où  $u$  est définie sur  $\mathbb{R}^+$  par

$$u(t) = tP'(t)e^{-t}.$$

Par croissance comparée,  $\lim_{t \rightarrow +\infty} u(t)Q(t) = 0$  et

$$\langle \alpha(P), Q \rangle = - \int_0^{+\infty} tP'(t)Q'(t)e^{-t} dt.$$

**19.b)** En reprenant l'expression précédent,  $P$  et  $Q$  ont un rôle symétrique dans l'expression de  $\langle \alpha(P), Q \rangle$ . Dès lors

$$\langle \alpha(P), Q \rangle = \langle \alpha(Q), P \rangle.$$

Puis par symétrie du produit scalaire

$$\langle \alpha(P), Q \rangle = \langle P, \alpha(Q) \rangle.$$

**20.** Soient  $k, \ell \in \mathbb{N}$  avec  $k \neq \ell$ . Comme  $P_k$  est vecteur propre

$$\langle \alpha(P_k), P_\ell \rangle = -k \langle P_k, P_\ell \rangle.$$

Mais en utilisant la symétrie et le fait que  $P_\ell$  soit vecteur propre, on a aussi

$$\langle \alpha(P_k), P_\ell \rangle = \langle \alpha(P_\ell), P_k \rangle = -\ell \langle P_k, P_\ell \rangle.$$

On a donc l'égalité

$$-k \langle P_k, P_\ell \rangle = -\ell \langle P_k, P_\ell \rangle,$$

et comme  $k \neq \ell$ , on a nécessairement

$$\langle P_k, P_\ell \rangle = 0.$$

Les polynômes  $P_k$  et  $P_\ell$  sont orthogonaux. La famille  $(P_0, \dots, P_n)$  est orthogonale.

En particulier, la famille (qui ne contient pas le vecteur nul) est aussi libre. Comme elle contient  $n + 1$  vecteurs, c'est une base de  $\mathbb{R}_n[x]$ .

**21.** Le système est équivalent à (question 13.b)

$$\forall i \in \llbracket 0; n \rrbracket, \quad \sum_{k=0}^n \lambda_k x_k^i = i! = \langle x^i, 1 \rangle.$$

→ Si ces équations sont vérifiées, on a pour tout  $P \in \mathbb{R}_n[x]$  avec  $P = \sum_{i=0}^n a_i x^i$

$$\begin{aligned} \sum_{k=1}^n \lambda_k P(x_k) &= \sum_{k=1}^n \lambda_k \sum_{i=0}^n a_i x_k^i \\ &= \sum_{i=0}^n a_i \left( \sum_{k=1}^n \lambda_k x_k^i \right) \\ &= \sum_{i=0}^n a_i \langle x^i, 1 \rangle \\ &= \left\langle \sum_{i=0}^n a_i x^i, 1 \right\rangle \\ &= \langle P(x), 1 \rangle \end{aligned}$$

$$\sum_{k=1}^n \lambda_k P(x_k) = \int_0^{+\infty} P(t) e^{-t} dt.$$

→ Réciproquement, si  $(\star)$  est vrai alors pour le choix du monôme  $P = x^i$ , on obtient bien le système car

$$\sum_{k=1}^n \lambda_k x_k^i = \int_0^{+\infty} t^i e^{-t} dt = \langle x^i, 1 \rangle = i!$$

**22.a)** Comme  $\dim \mathbb{R}_{n-1}[x] = \dim \mathbb{R}^n$ , il suffit de vérifier que l'application est injective. Or pour  $P \in \text{Ker}(\varphi)$ , on a

$$P(x_1) = P(x_2) = \dots = P(x_n) = 0.$$

Le polynôme  $P$  de degré au plus  $n-1$  a au moins  $n$  racines. C'est donc le polynôme nul. On en déduit que

$$\text{Ker } \varphi \subset \{0_{\mathbb{R}_{n-1}[x]}\}$$

puis  $\text{Ker } \varphi = \{0_{\mathbb{R}_{n-1}[x]}\}$

car le polynôme nul vérifie  $\varphi(0_{\mathbb{R}_{n-1}[x]}) = (0, \dots, 0)$ . Finalement,  $\varphi$  est un isomorphisme.

**22.b)** On a

$$A = \begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \dots & x_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^{n-1} \end{bmatrix}$$

Le système devient

$$AA = F \quad \text{où} \quad \begin{cases} A = [\lambda_1 \dots \lambda_n] \\ F = [0! \dots (n-1)!]. \end{cases}$$

Comme  $A$  est inversible, on a une unique solution donnée par  $\Lambda = FA^{-1}$ .

**23.** Soit  $P \in \mathbb{R}_{2n-1}[x]$ . Il existe un unique couple  $(S, R) \in \mathbb{R}[x]^2$  tel que

$$P = SQ + R \quad \text{avec} \quad \deg R < \deg(Q) = n.$$

Dans ce cas, on a par bilinéarité du produit scalaire

$$\langle P, 1 \rangle = \langle SQ, 1 \rangle + \langle R, 1 \rangle.$$

Or  $\deg P < n$ , donc la relation  $(\star)$  donne

$$\langle R, 1 \rangle = \sum_{k=1}^n \lambda_k R(x_k) = \sum_{k=1}^n \lambda_k P(x_k).$$

En effet,  $Q$  admet pour racines les réels  $x_i$  donc pour tout indice  $k$

$$P(x_k) = S(x_k) \underbrace{Q(x_k)}_{=0} + R(x_k) = R(x_k).$$

Notons de plus que d'après la formule définissant le produit scalaire

$$\langle SQ, 1 \rangle = \langle S, Q \rangle.$$

Précisons ensuite que  $Q = P_n$  car  $P_n$  et  $Q$  ont les mêmes  $n$  racines, le même degré et le même coefficient dominant. Comme  $S$  est de degré au plus  $n-1$ ,  $S$  est une combinaison linéaire de  $(P_0, \dots, P_{n-1})$  (on rappelle que la famille des  $n$  premiers polynômes  $P_i$  sont de degré échelonné  $(0, 1, 2, \dots, n-1)$ , ils constituent une base de  $\mathbb{R}_{n-1}[x]$ ). Ainsi la famille  $(P_0, \dots, P_n)$  étant orthogonale

$$\langle S, Q \rangle = \langle S, P_n \rangle = 0.$$

Finalement

$$\langle P, 1 \rangle = \underbrace{\langle SQ, 1 \rangle}_{=0} + \langle R, 1 \rangle = \sum_{k=1}^n \lambda_k P(x_k).$$

**24.** Posons  $P = Q^2 \in \mathbb{R}_{2n}[x]$  de sorte que

$$\sum_{i=1}^n \lambda_i P(x_i) = 0$$

et

$$\int_0^{+\infty} P(t) e^{-t} dt = \int_0^{+\infty} Q(t) Q(t) e^{-t} dt = \|Q\|^2 > 0.$$

Les deux quantités sont bien distinctes.

**25.**

```
def facto(n):
    F=np.ones(n)
    for i in range(1,n):
        F[i]=i*F[i-1]
    return F

>>> facto(5)
array([ 1.,  1.,  2.,  6., 24.] )
```

Noter que si on commence par 1!, on peut écrire simplement

```
np.cumprod(np.arange(1,n))
```

26.

```
# La transposée de A est la matrice de
# Vandermonde
def V(X):
    n=len(X)
    A=np.ones([n,n])
    for i in range(n):
        for j in range(1,n):
            A[i,j]=X[i]*A[i,j-1]
    return A

# Un test
X=np.arange(1,5)
print(V(X))

>>> [[ 1.  1.  1.  1.]
      [ 1.  2.  4.  8.]
      [ 1.  3.  9. 27.]
      [ 1.  4. 16. 64.]]
```

27.

```
import numpy.linalg as al

def Approx(f,n):
    X=racines(n)
    Lbda=facto(n)@(al.inv(V(X)))
    # On peut aussi la commande al.solve()
    # , plus précise
    s=0
    for i in range(n):
        s+=Lbda[i]*f(i)
    return s
```

---

### Exercice 2

---

28. L'application  $\varphi_u$  est une application de E dans lui-même. Soient  $\lambda \in \mathbb{R}$  et  $x, y \in E$

$$\begin{aligned} \varphi_u(x + \lambda y) &= 2 \frac{\langle x + \lambda y, u \rangle}{\langle u, u \rangle} u - (x + \lambda y) \\ &= 2 \frac{\langle x, u \rangle + \lambda \langle y, u \rangle}{\langle u, u \rangle} u - x - \lambda y \quad \text{bilinéarité de } \langle \cdot, \cdot \rangle \\ &= 2 \frac{\langle x, u \rangle}{\langle u, u \rangle} u - x + \lambda \left[ 2 \frac{\langle y, u \rangle}{\langle u, u \rangle} u - y \right] \\ \varphi_u(x + \lambda y) &= \varphi_u(x) + \lambda \varphi_u(y) \end{aligned}$$

L'application  $\varphi$  est bien un endomorphisme de E.

29. Vérifier que pour tout  $x \in E$ ,

$$\varphi_u \circ \varphi_u(x) = x.$$

Dit autrement,  $\varphi_u \circ \varphi_u = \text{id}_E$ .

D'après la caractérisation des applications bijectives,  $\varphi_u$  est bijective et

$$\varphi_u^{-1} = \varphi_u.$$

30.a) Soit  $x \in E$ . En utilisant la bilinéarité et la symétrie du produit scalaire,

$$\begin{aligned} \langle \varphi_u(x), \varphi_u(x) \rangle &= \left\langle 2 \frac{\langle x, u \rangle}{\langle u, u \rangle} u - x, 2 \frac{\langle x, u \rangle}{\langle u, u \rangle} u - x \right\rangle \\ &= 4 \frac{\langle x, u \rangle^2}{\langle u, u \rangle^2} \langle u, u \rangle - 4 \frac{\langle x, u \rangle}{\langle u, u \rangle} \langle u, x \rangle + \langle x, x \rangle \\ &= 4 \frac{\langle x, u \rangle^2}{\langle u, u \rangle} - 4 \frac{\langle x, u \rangle^2}{\langle u, u \rangle} + \langle x, x \rangle \\ \langle \varphi_u(x), \varphi_u(x) \rangle &= \langle x, x \rangle \end{aligned}$$

30.b) Soient  $x, y \in E$

$$\begin{aligned} \|x + y\|^2 - \|x - y\|^2 &= \|x\|^2 + 2\langle x, y \rangle + \|y\|^2 \\ &\quad - (\|x\|^2 - 2\langle x, y \rangle + \|y\|^2) \\ &= 4\langle x, y \rangle. \end{aligned}$$

30.c) Appliquons les deux résultats précédents :

$$\begin{aligned} \langle \varphi_u(x), \varphi_u(x) \rangle &= \frac{1}{4} \left( \|\varphi_u(x) + \varphi_u(y)\|^2 - \|\varphi_u(x) - \varphi_u(y)\|^2 \right) \\ &= \frac{1}{4} \left( \|\varphi_u(x + y)\|^2 - \|\varphi_u(x - y)\|^2 \right) \\ &= \frac{1}{4} \left( \|x + y\|^2 - \|x - y\|^2 \right) \\ \langle \varphi_u(x), \varphi_u(x) \rangle &= \langle x, y \rangle. \end{aligned}$$

31. On remarque que

$$\varphi_u(u) = u \quad \text{et} \quad u \neq 0_E.$$

Ainsi 1 est valeur propre de  $\varphi$  et

$$D_u = \text{Vect}(u) \subset E_1(\varphi).$$

De plus, pour  $v \in H$ ,  $v$  est orthogonal à  $u$  et

$$\varphi_u(v) = -v.$$

On en déduit que  $-1$  est valeur propre de  $\varphi$  et

$$H_u = \text{Vect}(u)^\perp \subset E_{-1}(\varphi).$$

or, dans un espace euclidien

$$D_u \oplus H_u = \text{Vect}(u) \oplus \text{Vect}(u)^\perp = E.$$

Comme les sous-espaces propres associés à des valeurs propres distinctes sont en somme directe, on en déduit que

$$E_1(\varphi) \oplus E_{-1}(\varphi) = E.$$

Cela justifie que  $\varphi$  est diagonalisable avec deux sous-espaces propres

$$E_{-1}(\varphi) = H_u \quad \text{et} \quad E_1(\varphi) = \text{Vect}(u).$$

👁 | Géométriquement,  $\varphi$  est une symétrie orthogonale.

32. Le sous-espace H est constitué des vecteurs  $X = (x, y, z)$  tels que  $x + y + z = 0$ . On reconnaît l'équation d'un plan de  $\mathbb{R}^3$  engendré par les vecteurs

$$v_1 = (1, -1, 0) \quad \text{et} \quad v_2 = (1, 1, -2).$$

On en déduit que

$$\dim(H^\perp) = \dim \mathbb{R}^3 - \dim H = 1.$$

On constate que  $v_3 = (1, 1, 1)$  est orthogonal à  $v_1$  et  $v_2$ . Le vecteur  $v_3$  est non nul et appartient à  $H^\perp$ . Il constitue une base de  $H^\perp$ .

Le vecteur  $v_3$  est un vecteur normal au plan. Ces coordonnées se retrouvent dans l'équation cartésienne de  $H$ .

On normalise les vecteurs pour obtenir une base orthonormée. Finalement, le sous-espace  $H^\perp$  est de dimension 1 et admet pour base orthonormale la famille formée d vecteur

$$w_3 = \left( \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}} \right)$$

et  $H$  admet pour base orthonormée

$$w_1 = \frac{\sqrt{2}}{2} \cdot (1, -1, 0) \quad \text{et} \quad w_2 = \frac{1}{\sqrt{5}}(1, 1, -2).$$

Nous avons fait le choix en amont de  $v_1$  et  $v_2$  orthogonaux. Si on n'avait pas ce choix, on aurait pu obtenir  $w_1$  et  $w_2$  par le procédé d'orthonormalisation de Schmidt.

**33.a)** Comme la famille est une base orthonormée de  $\mathbb{R}^3$ , on sait que pour tout  $X \in \mathbb{R}^3$

$$X = \underbrace{\langle X, w_1 \rangle w_1 + \langle X, w_2 \rangle w_2}_{\in H} + \underbrace{\langle X, w_3 \rangle w_3}_{\in H^\perp}.$$

Par définition du projecteur sur  $H^\perp$  parallèlement à  $H$

$$p(X) = \langle X, w_3 \rangle w_3.$$

**33.b)**

$$\begin{aligned} p(X) &= \langle X, w_3 \rangle w_3 \\ &= \left( \frac{x}{\sqrt{3}} + \frac{y}{\sqrt{3}} + \frac{z}{\sqrt{3}} \right) \begin{pmatrix} 1/\sqrt{3} & 1/\sqrt{3} & 1/\sqrt{3} \end{pmatrix} \\ &= \frac{1}{3} \begin{pmatrix} x+y+z & x+y+z & x+y+z \end{pmatrix}. \end{aligned}$$

On en déduit que la matrice est

$$B = \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}.$$

On a  $q = \text{id}_{\mathbb{R}^3} - p$ . D'où la matrice du projecteur associée est

$$\frac{1}{3} \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}.$$

**34.** Dans ce cas, avec  $v$  unitaire

$$\forall x \in E, \quad \varphi_v(x) = 2\langle x, v \rangle v - x.$$

Comme  $v \in H^\perp$ , il existe  $\lambda \in \mathbb{R}$  tel que

$$v = \lambda w_3.$$

Or  $v$  est unitaire,  $w_3$  aussi

$$1 = \|v\| = |\lambda| \cdot \|w_3\| = |\lambda| \text{ donc } \lambda = \pm 1$$

et

$$\begin{aligned} \varphi_v(x) &= 2\langle x, \lambda w_3 \rangle \lambda w_3 - x \\ &= 2\lambda^2 \langle x, w_3 \rangle w_3 - x \\ &= 2\langle x, w_3 \rangle w_3 - x. \end{aligned}$$

On constate que

$$\begin{aligned} \varphi_v &= 2p - \text{id}_E \\ &= 2p - (p + q) = p - q. \end{aligned}$$

La matrice de  $\varphi_v$  dans la base canonique est

$$\frac{1}{3} \begin{bmatrix} -1 & 2 & 2 \\ 2 & -1 & 2 \\ 2 & 2 & -1 \end{bmatrix}.$$

**35.** Soit  $x \in E$ , il existe  $y \in \Delta$  et  $z \in \Delta^\perp$  tels que  $x = y + z$ . Par définition de  $\psi$ , on a alors  $\psi(y) = y$  et  $\psi(z) = -z$  d'où

$$\begin{aligned} \psi(\psi(x)) &= \psi(\psi(y) + \psi(z)) \\ &= \psi(y - z) \\ &= \psi(y) - \psi(z) \\ &= y - (-z) \\ &= y + z \\ \psi(\psi(x)) &= x \end{aligned}$$

Ainsi, on a bien

$$\psi \circ \psi = \text{id}_E$$

Comme  $y \in \Delta$  et  $z \in \Delta^\perp$ , on a  $\langle y, z \rangle = 0$ . En réutilisant le fait que  $\psi(x) = y - z$  et la bilinéarité du produit scalaire, il vient

$$\langle \psi(x), \psi(x) \rangle = \langle y - z, y - z \rangle = \langle y, y \rangle - 2\langle z, y \rangle + \langle z, z \rangle = \langle y, y \rangle + \langle z, z \rangle$$

De même, on a

$$\langle x, x \rangle = \langle y + z, y + z \rangle = \langle y, y \rangle + 2\langle y, z \rangle + \langle z, z \rangle = \langle y, y \rangle + \langle z, z \rangle$$

Comme  $\langle \psi(x), \psi(x) \rangle = \langle x, x \rangle$ , on déduit que  $\|\psi(x)\| = \|x\|$  et que  $\psi$  conserve la norme. C'est donc une isométrie vectorielle, et l'endomorphisme  $\psi$  conserve le produit scalaire.

**36.** Soit  $u$  un vecteur unitaire appartenant à  $\Delta$ . Montrons que les applications  $\varphi_u$  et  $\psi$  coïncident sur  $\Delta$  et  $\Delta^\perp$ . Soit  $x \in \Delta$ , alors il existe  $\lambda \in \mathbb{R}$  tel que  $x = \lambda u$ . Par bilinéarité du produit scalaire,

$$\begin{aligned} \varphi_u(x) &= 2 \frac{\langle \lambda u, u \rangle}{\langle u, u \rangle} u - \lambda u \\ &= 2\lambda \frac{\langle u, u \rangle}{\langle u, u \rangle} u - \lambda u \\ &= 2\lambda u - \lambda u = \lambda u = x = \psi(x). \end{aligned}$$

Ainsi,  $\varphi_u$  et  $\psi$  coïncident sur  $\Delta$ . Considérons maintenant  $x \in \Delta^\perp$ . Comme  $u \in \Delta$ , on remarque que  $\langle x, u \rangle = 0$  et donc

$$\varphi_u(x) = 2 \frac{\langle x, u \rangle}{\langle u, u \rangle} u - x = -x = \psi(x).$$

De ce fait,  $\varphi_u$  et  $\psi$  coïncident sur  $\Delta^\perp$ . Comme  $\Delta$  et  $\Delta^\perp$  sont deux espaces supplémentaires, on conclut que  $\psi = \varphi_u$ . Ce qui conclut.



DS 4\* - solution

**Exercice 1**

Voir exercice 1 du sujet A.

**Problème**  
adapté de ESCP 1997

**10. Rédaction 1.**

Soit  $N$  un entier fixé  $N \geq C^2 + 1$ . Pour tout  $n \geq N$

$$0 \leq \frac{C^{2n}}{n!} \leq \underbrace{\frac{C^2 \times \dots \times C^2}{1 \times \dots \times N}}_{\text{Cste}} \times \underbrace{\frac{C^2}{N+1} \times \dots \times \frac{C^2}{(n-1)}}_{\leq 1} \times \frac{C^2}{n}$$

$$\leq \text{Cste} \times \frac{C^2}{n}.$$

Par encadrement  $\frac{C^{2n}}{n!} \xrightarrow{n \rightarrow \infty} 0$ .

**Rédaction 2.**

La série  $\sum \frac{(C^2)^n}{n!}$  est une série exponentielle convergente, on sait alors que le terme général tend vers 0.

• D'une part

$$u_{2n} = \frac{C^{2n}}{[2n/2]!} = \frac{C^{2n}}{n!} \xrightarrow{n \rightarrow \infty} 0.$$

D'autre part

$$u_{2n+1} = \frac{C^{2n+1}}{[n + \frac{1}{2}]!} = \frac{C^{2n+1}}{n!} = C \cdot \frac{C^{2n}}{n!} \xrightarrow{n \rightarrow \infty} 0.$$

On en déduit que les suites extraites  $(u_{2n}), (u_{2n+1})$  convergent vers une limite commune. D'après le cours, la suite  $(u_n)_{n \in \mathbb{N}}$  converge aussi vers cette limite

$$u_n \xrightarrow{n \rightarrow \infty} 0.$$

**11. Soit  $k \in \mathbb{N}$**

$$0 \leq u_{2k} + u_{2k+1} = \frac{C^{2k}}{k!} + \frac{C^{2k+1}}{k!}$$

$$= (1 + C) \cdot \frac{(C^2)^k}{k!}$$

À un terme multiplicatif près, on reconnaît le terme général de la série exponentielle de paramètre  $C^2$ . Il y a donc convergence et

$$\sum_{k=0}^{+\infty} (u_{2k} + u_{2k+1}) = (1 + C)e^{C^2}.$$

**12. Soit  $N \in \mathbb{N}$ . Posons  $S_N = \sum_{k=0}^N u_k$  de sorte que**

$$S_{2N} = \sum_{k=0}^{2N} u_k = \sum_{k=0}^{2N-1} u_k + u_{2N}$$

$$= \sum_{k=0}^{N-1} (u_{2k} + u_{2k+1}) + u_{2N}$$

En reprenant les questions précédentes

$$S_{2N} \xrightarrow{N \rightarrow +\infty} \sum_{k=0}^{+\infty} (u_{2k} + u_{2k+1}) + 0 = (1 + C)e^{C^2}$$

De plus

$$S_{2N+1} = S_{2N} + u_{2N+1} \xrightarrow{N \rightarrow +\infty} (1 + C)e^{C^2} + 0.$$

En distinguant termes d'indice pair et impair, on en déduit

$$S_N \xrightarrow{N \rightarrow +\infty} (1 + C)e^{C^2}.$$

Par définition, la série converge et

$$\sum_{k=0}^{+\infty} u_k = (1 + C)e^{C^2}.$$

**13. Soit  $\lambda \in [0; a]$ . Par inégalité triangulaire**

$$\left| \int_0^\lambda \frac{(\lambda - t)^n}{n!} g^{(n+1)}(t) dt \right| \leq \int_0^\lambda \frac{(\lambda - t)^n}{n!} |g^{(n+1)}(t)| dt$$

$$\leq \frac{K^{n+1}}{[\frac{n+1}{2}]!} \int_0^\lambda (\lambda - t)^n dt$$

En effectuant le changement de variable affine  $u = \lambda - t$

$$\int_0^\lambda (\lambda - t)^n dt = \int_0^\lambda u^n du = \frac{\lambda^{n+1}}{n+1}.$$

Par conséquent

$$\left| \int_0^\lambda \frac{(\lambda - t)^n}{n!} g^{(n+1)}(t) dt \right| \leq \frac{(K\lambda)^{n+1}}{(n+1) \cdot [\frac{n+1}{2}]!} \xrightarrow{n \rightarrow \infty} 0.$$

Par encadrement, on a bien

$$\int_0^\lambda \frac{(\lambda - t)^n}{n!} g^{(n+1)}(t) dt \xrightarrow{n \rightarrow \infty} 0.$$

Le raisonnement est identique pour  $\lambda \in [-a; 0]$ .

**14. D'après la formule de Taylor avec reste intégral avec  $g$  de classe  $\mathcal{C}^{n+1}$  ( $n$  est fixé). Pour tout  $\lambda \in [-a; a]$**

$$g(\lambda) = \sum_{k=0}^n \frac{g^{(k)}(0)}{k!} \lambda^k + \int_0^\lambda \frac{(\lambda - t)^n}{n!} g^{(n+1)}(t) dt.$$

Le résultat s'en déduit à l'aide de la question précédente et par passage à la limite lorsque  $n \rightarrow +\infty$ .

15. La fonction  $t \in \mathbb{R} \mapsto P(t)\varphi(t)$  est continue sur  $\mathbb{R}$ , on a donc une intégrale généralisée en  $\pm\infty$ . Par les croissances comparées

$$P(t) = o_{\pm\infty} \left( \frac{1}{t^2} \right).$$

Or les intégrales de Riemann

$$\int_1^{+\infty} (1/t^2) dt \quad \text{et} \quad \int_{-\infty}^{-1} (1/t^2) dt$$

sont convergentes et d'intégrandes positives. D'après le critère de négligeabilité, l'intégrale

$$\int_{-\infty}^{+\infty} P(t)e^{-t^2/2} dt$$

converge.

16. Voir cours.

17. Adaptons le procédé d'orthonormalisation de Schmidt.

→ *Étape 1*

Le polynôme constant 1 est de coefficient dominant 1 donc

$$P_0 = 1.$$

Notons que  $P_0$  est aussi de norme 1.

→ *Étape 2*

Par imparité de la fonction  $t \mapsto t\varphi(t)$  et sachant que l'intégrale est convergente, on a

$$\langle 1, x \rangle = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} 1 \times t\varphi(t) dt = 0.$$

Les vecteurs sont donc déjà orthogonaux. De plus

$$\|x\|^2 = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} t^2 e^{-t^2/2} dt = 1$$

en utilisant le moment d'ordre 2 d'une loi normale centrée réduite ou le résultat admis en début de problème. On pose donc

$$P_1 = x.$$

Précisons que  $P_1$  est de norme 1.

→ *Étape 3*

Déterminons un vecteur :  $P_2 \in \mathbb{R}_2[x]$  orthogonal à  $P_0$  et  $P_1$ . On cherche  $P_2$ , sous la forme

$$P_2(x) = x^2 + \lambda_1 P_1 + \lambda_2.$$

Les conditions  $\langle P_2, x \rangle = 0$  et  $\langle P_2, 1 \rangle = 0$  imposent

$$P_2(x) = x^2 - \langle x^2, x \rangle x - \langle x^2, 1 \rangle 1$$

Or  $\langle x^2, x \rangle = I_3 = \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}} t^3 \varphi(t) dt = 0$

par imparité de l'intégrande et  $\langle x^2, 1 \rangle = I_2 = 1$ . Finalement

$$P_2(x) = x^2 - 1.$$

18. Soit  $t \in \mathbb{R}$ .

$$\varphi^{(0)}(t) = e^{-t^2/2}, \quad \varphi^{(1)}(t) = -te^{-t^2/2}$$

$$\varphi^{(2)}(t) = (t^2 - 1)e^{-t^2/2}, \quad \varphi^{(3)}(t) = (3t - t^3)e^{-t^2/2}.$$

On en déduit que

$$\begin{aligned} H_0(x) &= 1, & H_1(x) &= x \\ H_2(x) &= x^2 - 1, & H_3(x) &= x^3 - 3x. \end{aligned}$$

19. Posons pour tout  $t \in \mathbb{R}$ ,  $Q(t) = -t$  de sorte que

$$Q'(t) = -1, \quad Q^{(k)}(t) = 0 \text{ pour } k \geq 2$$

et  $\varphi'(t) = Q(t)\varphi(t)$ .

D'après la formule de Leibniz avec  $n \in \mathbb{N}^*$

$$\varphi^{(n+1)}(t) = (\varphi')^{(n)}(t) = \sum_{k=0}^n \binom{n}{k} Q^{(k)}(t) \varphi^{(n-k)}(t)$$

$$= \sum_{k=0}^1 \binom{n}{k} Q^{(k)}(t) \varphi^{(n-k)}(t)$$

$$\varphi^{(k+1)}(t) = -t\varphi^{(k)}(t) - n\varphi^{(k-1)}(t).$$

D'où

$$H_{n+1}(t) = (-1)^{n+1} e^{t^2/2} \varphi^{(n+1)}(t)$$

$$= t(-1)^n e^{t^2/2} \varphi^{(n)}(t) - n(-1)^{n-1} e^{t^2/2} \varphi^{(n-1)}(t)$$

$$= tH_n(t) - nH_{n-1}(t).$$

La relation  $(\bullet)$  est prouvée.

• D'après la formule de dérivation d'un produit

$$H'_n(t) = (-1)^n \left( te^{t^2/2} \varphi^{(n)}(t) + e^{t^2/2} \varphi^{(n+1)}(t) \right)$$

$$= tH_n(t) - H_{n+1}(t)$$

$$H'_n(t) = nH_{n-1}(t)$$

en utilisant la relation  $(\bullet)$ .

20. Prouvons par récurrence que la propriété

$$\mathcal{P}(n) : \begin{cases} H_n \in \mathbb{R}[x] \\ H_n \text{ est de degré } n \\ \text{et de coefficient dominant } 1 \\ H_n(-x) = (-1)^n H_n(x) \text{ pour tout } x \in \mathbb{R} \end{cases}$$

est vraie pour tout  $n \in \mathbb{N}$ . Précisons que la dernière assertion signifie que le polynôme  $H_n$  est pair si  $n$  est un entier pair et  $H_n$  est impair si  $n$  est impair.

• *Rédaction 1.* Faire une récurrence double en utilisant la relation  $(\bullet)$ .

• *Rédaction 2.* Récurrence simple en utilisant  $(\bullet\bullet)$ . Écrivons cette seconde récurrence.

→ *Initialisation.* Comme  $H_0 = 1$ ,  $\mathcal{P}(0)$  est claire.

→ *Hérédité.* Soit  $n \in \mathbb{N}$ . Supposons  $\mathcal{P}(n)$  vraie. À partir de  $H'_{n+1} = (n+1)H_n$ , on a pour tout  $x \in \mathbb{R}$

$$H_{n+1}(x) = H_{n+1}(0) + \int_0^x (n+1)H_n(t) dt.$$

En intégrant un polynôme  $(n+1)H_n$  de degré  $n$ , on obtient bien un polynôme de degré  $n+1$ . Si on note  $\alpha_{n+1}$  le coefficient dominant de  $H_{n+1}(x)$

$$H_{n+1}(x) = \alpha_{n+1}x^{n+1} + \underbrace{Q_{n+1}(x)}_{\in \mathbb{R}_n[x]}$$

$$H_{n+1}(x) = \alpha_{n+1}(n+1)x^n + \underbrace{Q'_{n+1}(x)}_{\in \mathbb{R}_{n-1}[x]}$$

Comme  $H_{n+1} = (n+1)H_n$  et que  $H_n$  est unitaire

$$\alpha_{n+1}(n+1) = (n+1) \quad \text{puis} \quad \alpha_{n+1} = 1.$$

$H_{n+1}$  est bien de coefficient dominant 1. Posons pour tout  $x \in \mathbb{R}$

$$g(x) = H_{n+1}(-x) - (-1)^{n+1}H_{n+1}(x).$$

La fonction  $g$  est dérivable sur  $\mathbb{R}$  car polynomial et

$$\begin{aligned} g'(x) &= -H'_{n+1}(-x) - (-1)^{n+1}H'_{n+1}(x) \\ &= -(n+1)(H_n(-x) - (-1)^n H_n(x)) \\ &= 0. \end{aligned}$$

La fonction  $g$  est constante sur l'intervalle  $\mathbb{R}$ . Or

$$g(0) = H_{n+1}(0) - (-1)^{n+1}H_{n+1}(0)$$

- Si  $n$  est impair, on a directement  $g(0) = 0$ .

- Si  $n$  est pair, on montre question 21 que  $H_{n+1}(0) = 0$ . On a encore  $g(0) = 0$ .

Dès lors

$$\forall x \in \mathbb{R}, \quad g(x) = 0.$$

C'est-à-dire

$$\forall x \in \mathbb{R}, \quad H_{n+1}(-x) = (-1)^{n+1}H_n(x).$$

→ *Conclusion.* La propriété est vraie pour tout  $n \in \mathbb{N}$ .

**21.** En reprenant (•), on a pour tout  $n \in \mathbb{N}^*$

$$H_{n+1}(0) = 0 \cdot H_n(0) - nH_{n-1}(0)$$

c'est-à-dire  $h_{n+1} = -nh_{n-1}$ .

Notons que

$$h_0 = 1 \quad \text{et} \quad h_1 = 0.$$

Par conséquent pour  $n$  impair  $h_n = 0$  et pour  $n = 2p$  avec  $p \in \mathbb{N}$  pair, on utilise la relation de récurrence

$$h_{2(p+1)} = h_{2p+2} = -(2p+1)h_{2(p-1)}.$$

Un code possible est alors

```
def Hzero(n):
    if n%2!=0:
        return 0
    # On peut aussi tester la parité avec 2*np
    # .floor(n/2)!=n

    h=1
    p=int(n/2)
    for k in range(p):
        h*=- (2*k+1)
    return h
```

**22.** Soit  $Q = \sum_{i=0}^{n+1} b_i x^i$ , on a  $b_0 = 0$  et pour tout  $i \in \llbracket 1; n+1 \rrbracket$

$$b_i = \frac{a_{i-1}}{i}.$$

On en déduit le code :

```
def inte(MP):
    n=len(MP)
    # le degré est d=n-1
    MQ=np.zeros(n+1)
    # Q gagne un degré
    for i in range(1,n+1):
        MQ[i]=MP[i-1]/i
    return MQ

# test
MP=np.array([1,2,3])
print(inte(MP))
>>> [0. 1. 1. 1.]
```

**23.**

```
def hermite(n):
    H=np.array([1])
    for i in range(n):
        H=inte(H)
        H=(i+1)*H
        H[0]=Hzero(i+1)
    return H
```

**24.**

```
def trace(n):
    x=np.linspace(-3,3,100)
    H=hermite(n)
    y=np.zeros(100)
    for i in range(n+1):
        y+=H[i]*x**i
    plt.plot(x,y)
    plt.show()
```

• On conjecture que le polynôme  $H_n$  de degré  $n$  a exactement  $n$  racines distinctes.

**25.** La relation (•) donne un lien entre les coefficients de  $H_{n+1}$ ,  $H_n$  et  $H_{n-1}$ . On en déduit le code

```
def Hmatrice(m):
    H=np.zeros([m,m])
    H[0,0]=1
    H[1,1]=1
    for n in range(1,m-1):
        for k in range(m):
            H[n+1,k]=H[n,k-1]-(n)*H[n-1,k]
    return H
```

**26.** Soit  $x \in \mathbb{R}$

$$P(x)\varphi^{(n-1)}(x) = \underbrace{(-1)^{n-1}P(x)H_{n-1}(H)}_{\in \mathbb{R}[x]} e^{-t^2/2}.$$

On conclut sur la limite à l'aide des croissances comparées.

27. Soient  $A, B \in \mathbb{R}$  avec  $A < B$ . Pour  $n \in \mathbb{N}^*$

$$\begin{aligned} \int_A^B H_n(t)\varphi(t) dt &= \int_A^B (-1)^n e^{t^2/2} \varphi^{(n)}(t) e^{-t^2/2} dt \\ &= \int_A^B (-1)^n \varphi^{(n)}(t) dt \\ \int_A^B H_h(t)\varphi(t) dt &= \left[ (-1)^n \varphi^{(n-1)}(t) \right]_A^B \underset{\substack{A \rightarrow -\infty \\ B \rightarrow +\infty}}{\rightarrow} 0. \end{aligned}$$

On en déduit que

$$\langle H_n, H_0 \rangle = 0.$$

Pour  $n = 0$ , on a directement  $\langle H_0, H_0 \rangle = 1$ .

28. On a

$$\begin{aligned} \langle H_n, H_m \rangle &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} H_n(t) e^{-t^2/2} \cdot H_m(t) dt \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} (-1)^n \varphi^{(n)}(t) \cdot H_m(t) dt. \end{aligned}$$

Intégrons par parties sachant que les fonctions considérées sont de classe  $\mathcal{C}^1$ . Pour  $A, B \in \mathbb{R}$  avec  $A < B$

$$\begin{aligned} \int_A^B H_m(t)\varphi^{(n)}(t) dt &= \left[ H_m(t)\varphi^{(n-1)}(t) \right]_A^B \\ &\quad - \int_A^B H'_m(t)\varphi^{(n-1)}(t) dt. \end{aligned}$$

En utilisant la question 26

$$\left[ H_m(t)\varphi^{(n-1)}(t) \right]_A^B \underset{\substack{A \rightarrow +\infty \\ B \rightarrow +\infty}}{\rightarrow} 0$$

et par la relation (\*)

$$\int_A^B H'_m(t)\varphi^{(n-1)}(t) dt = m \int_A^B H_{m-1}(t)\varphi^{(n-1)}(t) dt.$$

On en déduit que

$$\int_{-\infty}^{+\infty} H_m(t)\varphi^{(n)}(t) dt = -m \int_{-\infty}^{+\infty} H_{m-1}(t)\varphi^{(n-1)}(t) dt$$

Le résultat s'en déduit

$$\begin{aligned} &\frac{(-1)^n}{\sqrt{2\pi}} \left( -m \int_{-\infty}^{+\infty} H_{m-1}(t)\varphi^{(n-1)}(t) dt \right) \\ &= \frac{m(-1)^{n-1}}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} H_{m-1}(t)\varphi^{(n-1)}(t) dt \\ &= m \langle H_{n-1}, H_{m-1} \rangle. \end{aligned}$$

29. On a déjà vu que pour

$$m = 0, \quad \forall n \in \mathbb{N}^* \langle H_m, H_n \rangle = 0 \\ \langle H_0, H_0 \rangle = 1.$$

Soient  $m, n \in \mathbb{N}^*$  avec  $m \neq n$ . D'après le résultat précédent

$$\begin{aligned} m \langle H_{m-1}, H_{n-1} \rangle &= \langle H_m, H_n \rangle \\ &= \langle H_n, H_m \rangle = n \langle H_{n-1}, H_{m-1} \rangle \end{aligned}$$

Comme  $m \neq n$ , on a nécessairement  $\langle H_{n-1}, H_{m-1} \rangle = 0$  puis

$$\langle H_n, H_m \rangle = 0.$$

On aurait aussi pu procéder par récurrence en distinguant  $n < m$  et  $m > n$ .

• Pour  $n \in \mathbb{N}^*$ , on a  $\langle H_n, H_n \rangle = n \langle H_{n-1}, H_{n-1} \rangle$ . Comme  $H_i \neq 0_{\mathbb{R}[x]}$ , les produits scalaires  $\langle H_i, H_i \rangle$  sont non nuls et on peut utiliser un produit télescopique (afin d'éviter une récurrence) et

$$\begin{aligned} \langle H_n, H_n \rangle &= \frac{\langle H_n, H_n \rangle}{\langle H_0, H_0 \rangle} = \prod_{i=1}^n \frac{\langle H_i, H_i \rangle}{\langle H_{i-1}, H_{i-1} \rangle} \\ &= \prod_{i=1}^n i = n! \end{aligned}$$

30. D'après ce qui précède, la famille de  $\mathbb{R}_n[x]$

$$\left( H_0, \frac{H_1}{\sqrt{1!}}, \frac{H_2}{\sqrt{2!}}, \dots, \frac{H_n}{\sqrt{n!}} \right)$$

est orthonormée. Comme elle contient  $\dim \mathbb{R}_n[x]$  vecteurs, c'est même une base orthonormée de  $\mathbb{R}_n[x]$  (on rappelle qu'une famille orthonormée est automatiquement libre). Le résultat s'en déduit en explicitant le polynôme  $P$  dans cette base orthonormée.

31. Soit  $x \in \mathbb{R}$ . Soit  $K$  obtenu pour  $b = c = x$ . Soit  $\lambda \in \mathbb{R}$

$$\left| \lambda^n \frac{H_n(x)}{n!} \right| \leq \frac{(K\lambda)^n}{\left[ \frac{n}{2} \right]!}.$$

En reprenant le résultat 1, on a la convergence de la série

$$\sum \frac{(K\lambda)^n}{\left[ \frac{n}{2} \right]!}$$

Par le critère de majoration des séries à termes positifs, on prouve la convergence (absolue) de

$$\sum \frac{H_n(x)}{n!} \lambda^n.$$

32.a) On a pour tout  $\lambda \in \mathbb{R}$

$$g_x(\lambda) = \varphi(\lambda - x).$$

Par composition,  $g_x$  est  $\mathcal{C}^\infty$  sur  $\mathbb{R}$  avec pour tout  $n \in \mathbb{N}$

$$\forall \lambda \in \mathbb{R}, \quad g_x^{(n)}(\lambda) = 1 \times \rho^{(n)}(\lambda - x) = \varphi^{(n)}(\lambda - x).$$

32.b) On a donc pour  $\lambda \in [-a; a]$  où  $a \in \mathbb{R}$  est fixé

$$\left| g_x^{(n)}(\lambda) \right| \leq |H_n(\lambda - x)|.$$

Or pour  $\lambda \in [-a; a]$ ,  $\lambda - x \in [-a - x, a - x]$  et avec les choix  $b = -a - x, c = a - x$ , il existe un réel  $K$  (dépendant uniquement de  $a$  et  $x$ ) tel que

$$\left| \frac{H_n(\lambda - x)}{n!} \right| \leq \frac{K}{\left[ \frac{n}{2} \right]!}.$$

On a donc bien

$$\max_{t \in [-a, a]} \left| g^{(n)}(t) \right| \leq \frac{K}{\left[ \frac{n}{2} \right]!}$$

et avec le résultat de la question 14

$$\begin{aligned} g_x(\lambda) &= \sum_{k=0}^{+\infty} \frac{g^{(k)}(0)}{k!} \lambda^k \\ &= \sum_{k=0}^{+\infty} \frac{(-1)^k H_k(-x)}{k!} e^{-x^2/2} \lambda^k. \end{aligned}$$

Or par symétrie de  $H_k$  (question 20), on a

$$(-1)^k H_k(-x) = H_k(x)$$

et

$$g_x(\lambda) = e^{-\frac{\lambda^2}{2} + x\lambda - \frac{x^2}{2}} = e^{\lambda(x - \frac{\lambda}{2})} \cdot e^{-\frac{x^2}{2}}.$$

D'où le résultat après simplification.

**33.** L'intégrande est continue, on a donc une intégrale généralisée en  $\pm\infty$ . Par les croissances comparées

$$e^t H_n(t) e^{-t^2/2} = 0 \left( H_n(t) e^{-t^2/4} \right) = o_{\pm\infty} \left( \frac{1}{t^2} \right).$$

On conclut sur la convergence par les critères de Riemann et de négligeabilité. De plus, on vérifie par une intégration par parties que

$$\langle \exp, H_n \rangle = \langle \exp, H_{n-1} \rangle.$$

La suite  $(\langle \exp, H_n \rangle)_{n \in \mathbb{N}}$  est donc constante, la constante est donnée par

$$\begin{aligned} \langle \exp, H_0 \rangle &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} e^{-\frac{t^2}{2}} \cdot e^{-t} dt \\ &= \frac{1}{\sqrt{2\pi}} e^{1/2} \int_{-\infty}^{+\infty} e^{-\frac{1}{2}(t-1)^2} dt. \end{aligned}$$

À l'aide du changement de variable affine  $u = t - 1$

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} e^{-\frac{1}{2}(t-1)^2} dt = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} e^{-\frac{u^2}{2}} du = 1.$$

Résumons, pour tout  $n \in \mathbb{N}$

$$\langle \exp, H_n \rangle = e^{1/2}.$$

En reprenant l'égalité de la question 32. b) avec  $\lambda = 1$

$$e^{x - \frac{1}{2}} = \sum_{k=0}^{+\infty} \frac{H_k(x)}{k!}$$

puis

$$e^x = \sum_{k=0}^{+\infty} e^{1/2} \frac{H_k(x)}{k!} = \sum_{k=0}^{+\infty} \langle \exp, H_k \rangle \frac{H_k(x)}{k!}.$$

**34.** Le système est équivalent à (question 13.b))

$$\forall i \in \llbracket 0; n \rrbracket, \quad \sum_{k=0}^n \lambda_k x_k^i = J_i = \sqrt{2\pi} \langle x^i, 1 \rangle.$$

→ Si ces équations sont vérifiées, on a pour tout  $P \in \mathbb{R}_n[x]$

avec  $P = \sum_{i=0}^n a_i x^i$

$$\begin{aligned} \sum_{k=1}^n \lambda_k P(x_k) &= \sum_{k=1}^n \lambda_k \sum_{i=0}^n a_i x_k^i \\ &= \sum_{i=0}^n a_i \left( \sum_{k=1}^n \lambda_k x_k^i \right) \\ &= \sum_{i=0}^n a_i \sqrt{2\pi} \langle x^i, 1 \rangle \\ &= \sqrt{2\pi} \left\langle \sum_{i=0}^n a_i x^i, 1 \right\rangle \\ &= \sqrt{2\pi} \langle P(x), 1 \rangle \\ \sum_{k=1}^n \lambda_k P(x_k) &= \int_{-\infty}^{+\infty} P(t) e^{-t^2/2} dt. \end{aligned}$$

→ Réciproquement, si  $(\star)$  est vrai alors pour le choix du monôme  $P = x^i$ , on obtient bien le système car

$$\sum_{k=1}^n \lambda_k x_k^i = \int_0^{+\infty} t^i e^{-t} dt = J_i.$$

**35.a)** Comme  $\dim \mathbb{R}_{n-1}[x] = \dim \mathbb{R}^n$ , il suffit de vérifier que l'application est injective. Or pour  $P \in \text{Ker}(\varphi)$ , on a

$$P(x_1) = P(x_2) = \dots = P(x_n) = 0.$$

Le polynôme  $P$  de degré au plus  $n-1$  a au moins  $n$  racines. C'est donc le polynôme nul. On en déduit que

$$\text{Ker} \varphi \subset \{0_{\mathbb{R}_{n-1}[x]}\}$$

puis

$$\text{Ker} \varphi = \{0_{\mathbb{R}_{n-1}[x]}\}$$

car le polynôme nul vérifie  $\varphi(0_{\mathbb{R}_{n-1}[x]}) = (0, \dots, 0)$ . Finalement,  $\varphi$  est un isomorphisme.

**35.b)** On a

$$A = \begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \dots & x_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^{n-1} \end{bmatrix}$$

Le système devient

$${}^t A \Lambda = F$$

où

$$\Lambda = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} \quad \text{et} \quad F = \begin{bmatrix} J_0 \\ J_1 \\ \vdots \\ J_{n-1} \end{bmatrix}.$$

Comme  $A$  est inversible, sa transposée aussi et on a une unique solution donnée par

$$\Lambda = {}^t A^{-1} F.$$

**36.** Soit  $P \in \mathbb{R}_{2n-1}[x]$ . Il existe un unique couple  $(S, R) \in \mathbb{R}[x]^2$  tel que

$$P = SQ + R \quad \text{avec} \quad \deg R < \deg(Q) = n.$$

Dans ce cas, on a par bilinéarité du produit scalaire

$$\langle P, 1 \rangle = \langle SQ, 1 \rangle + \langle R, 1 \rangle.$$

Or  $\deg P < n$ , donc la relation  $(\star)$  donne

$$\langle R, 1 \rangle = \sum_{k=1}^n \lambda_k R(x_k) = \sum_{k=1}^n \lambda_k P(x_k).$$

En effet,  $Q$  admet pour racines les réels  $x_i$  donc pour tout indice  $k$

$$P(x_k) = S(x_k) \underbrace{Q(x_k)}_{=0} + R(x_k) = R(x_k).$$

Notons de plus que d'après la formule définissant le produit scalaire

$$\langle SQ, 1 \rangle = \langle S, Q \rangle.$$

Précisons ensuite que  $Q = H_n$  car  $H_n$  et  $Q$  ont les mêmes  $n$  racines, le même degré et le même coefficient dominant. Comme  $S$  est de degré au plus  $n-1$ ,  $S$  est une combinaison linéaire de  $(H_0, \dots, H_{n-1})$  (on rappelle que la famille des  $n$  premiers polynômes  $H_i$  sont de degré échelonné  $(0, 1, 2, \dots, n-1)$ , ils constituent une base de  $\mathbb{R}_{n-1}[x]$ ). Ainsi la famille  $(H_0, \dots, H_n)$  étant orthogonale

$$\langle S, Q \rangle = \langle S, H_n \rangle = 0.$$

Finalement

$$\langle P, 1 \rangle = \underbrace{\langle SQ, 1 \rangle}_{=0} + \langle R, 1 \rangle = \sum_{k=1}^n \lambda_k P(x_k).$$

37. Posons  $P = Q^2 \in \mathbb{R}_{2n}[x]$  de sorte que

$$\sum_{i=1}^n \lambda_i P(x_i) = 0$$

et

$$\int_{-\infty}^{+\infty} P(t)e^{-t^2} dt = \int_{-\infty}^{+\infty} Q(t)Q(t)e^{-t^2/2} dt = \sqrt{2\pi}\|Q\|^2 > 0.$$

Les deux quantités sont bien distinctes.

38. Par imparité de l'intégrande  $J_n = 0$  si  $n$  est impair et par le résultat admis au préliminaire

$$J_n = \sqrt{2\pi} \frac{(2n)!}{2^n \cdot n!}.$$

```
def matriceJ(n):
    facto=np.cumprod(np.arange(1,2*n))
    print(facto)
    p=int((n-1)/2)
    J=np.zeros([n,1])
```

**Bonus** C'est Henri Poincaré. Raymond Poincaré est son cousin, président du conseil lors de la première guerre mondiale.

Le mémoire dont parle Charles Hermite a valu à Henri Poincaré le prestigieux prix du roi de Suède. Or ce dernier contenait une erreur! Elle fut corrigée par la suite par Poincaré et devient une des erreurs les plus fécondes de l'histoire des mathématiques.



Henri Poincaré 1854-1912



Raymond Poincaré 1860-1934

```
# pour une matrice colonne
J[0]=1
for i in range(1,p+1):
    J[2*i]=facto[2*i+1]/(2**(i+1)*
    facto[i])
return (2*np.pi)**(1/2)*J
```

En pratique, il est plus judicieux d'utiliser une formule de récurrence entre  $J_{n+2}$  et  $J_n$ .

39.

```
# A est la matrice de Vandermonde
def V(X):
    n=len(X)
    A=np.ones([n,n])
    for j in range(n):
        for i in range(1,n):
            A[i,j]=X[i]*A[i,j-1]
    return A
```

```
# Un test
X=np.arange(1,5)
print(V(X))
```

```
>>>
[[ 1.  1.  1.  1.]
 [ 2.  4.  8. 16.]
 [ 3.  9. 27. 81.]
 [ 4. 16. 64. 256.]]
```

40.

```
import numpy.linalg as al

def Approx(f,n):
    X=racines(n)
    Lbda=al.solve(np.transpose(V(X)),
    matriceJ(n))
    s=0
    for i in range(n):
        s+=Lbda[i]*f(i)
    return s
```