

PRÉPARATION AUX ORAUX – EXERCICES CENTRALE 2

Solution de 1 : Centrale 2025 – Théorème de Liouville

1. (a)

```

1 def fibonacci(n):
2     a, b = 1, 1
3     for _ in range(n):
4         a, b = b, a + b
5     return a

```

Python

Python gère nativement les entiers de taille arbitraire. L'appel `fibonacci(200)` s'évalue instantanément et renvoie 453973694165307953197296969697410619233826.

(b)

```

1 def test_approximation(x, p, q, c, mu):
2     return abs(x - p/q) <= c / (q**mu)

```

Python

(c)

```

1 phi = (1 + 5 ** .5) / 2
2
3 for n in range(1, 13):
4     F_2n = fibonacci(2 * n)
5     F_2nm1 = fibonacci(2 * n - 1)
6     print(f"n={n}, Vérifié : {test_approximation(phi, F_2n, F_2nm1, 5 **
7         (-.5), 2)}")

```

Python

(d)

```

1 def meilleure_approximation(x, n):
2     q_min = 10 ** (n - 1) if n > 1 else 1
3     q_max = 10 ** n - 1
4
5     meilleur_p, meilleur_q = 0, 1
6     erreur_min = float('inf')
7
8     for q in range(q_min, q_max + 1):
9         p = round(q * x)
10        erreur = abs(x - p/q)
11        if erreur < erreur_min:
12            erreur_min = erreur
13            meilleur_p, meilleur_q = p, q
14
15    return (meilleur_p, meilleur_q)

```

Python

2. Soit $q \in \mathbb{N}^*$. On pose $p = \lfloor qx \rfloor$. On a alors $p \leq qx < p + 1$. On en déduit $-1 < qx - p < 1$ puis $|qx - p| \leq 1$.

En divisant par q , on obtient $\left| x - \frac{p}{q} \right| \leq \frac{1}{q}$.

Si x est irrationnel, l'écart est strictement positif et tend vers 0.

La suite des fractions $\frac{p}{q}$ prend donc une infinité de valeurs distinctes.

La définition est vérifiée avec $c = 1$ et $\mu = 1$. Donc $\mu(x) \geq 1$.

3. (a) $\frac{a}{b} - \frac{p}{q} = \frac{aq - bp}{bq}$. Comme les rationnels sont distincts, le numérateur $aq - bp$ est un entier non nul. Sa valeur absolue est supérieure ou égale à 1. D'où le résultat.

(b) Soit $x = \frac{a}{b}$. La question précédente assure que pour toute autre fraction $\frac{p}{q}$, on a

$\left| x - \frac{p}{q} \right| \geq \frac{1}{bq}$. L'exposant μ ne peut donc pas dépasser 1, sinon l'inégalité de la définition serait violée pour q grand.

Considérons la suite $p_n = na + 1$ et $q_n = nb$. On a $\left| x - \frac{p_n}{q_n} \right| = \frac{1}{nb} = \frac{1}{q_n}$. Ces fractions sont distinctes et valident la définition avec $c = 1$ et $\mu = 1$. Donc $\mu(x) = 1$.

4. L'ensemble $\mathbb{Z}_n[X]$ est en bijection avec \mathbb{Z}^{n+1} . Il est donc dénombrable. L'ensemble $\mathbb{Z}[X]$ est la réunion dénombrable des $\mathbb{Z}_n[X]$, il est dénombrable.

Chaque polynôme non nul admet un nombre fini de racines. L'ensemble des nombres algébriques est donc une réunion dénombrable d'ensembles finis. Il est au plus dénombrable.

\mathbb{R} n'est pas dénombrable. Le complémentaire de l'ensemble des nombres algébriques n'est donc pas vide.

5. (a) Soit z une racine de P avec $|z| > 1$. On a $a_n z^n = -\sum_{k=0}^{n-1} a_k z^k$.

$$|a_n| |z|^n \leq \sum_{k=0}^{n-1} |a_k| |z|^k \leq |a_n| M \sum_{k=0}^{n-1} |z|^k = |a_n| M \frac{|z|^n - 1}{|z| - 1}.$$

En simplifiant par $|a_n| |z|^n$ (qui est non nul) et en majorant $\frac{|z|^n - 1}{|z|^n}$ par 1, il vient

$$1 \leq \frac{M}{|z| - 1}. \text{ On obtient } |z| \leq 1 + M.$$

(b) $P(x) = 0$. Soit $\frac{p}{q}$ un rationnel. On suppose $\left| x - \frac{p}{q} \right| \leq 1$. L'écart est trivialement minoré dans le cas contraire.

Le polynôme P est de degré n et annule x algébrique d'ordre n .

Il est donc irréductible sur \mathbb{Q} et n'admet aucune racine rationnelle.

La valeur $P\left(\frac{p}{q}\right)$ est un rationnel non nul de dénominateur q^n . On a donc $\left| P\left(\frac{p}{q}\right) \right| \geq \frac{1}{q^n}$.

Dans $\mathbb{C}[X]$, le polynôme se factorise sous la forme $P(X) = a_n(X - x) \prod_{i=2}^n (X - \alpha_i)$ où les α_i sont les autres racines complexes.

L'évaluation en $\frac{p}{q}$ donne $\left| P\left(\frac{p}{q}\right) \right| = |a_n| \left| \frac{p}{q} - x \right| \prod_{i=2}^n \left| \frac{p}{q} - \alpha_i \right|$.

D'après la question 5.a, on a $|\alpha_i| \leq 1 + M$. On majore la distance pour chaque racine avec l'inégalité triangulaire

$$\left| \frac{p}{q} - \alpha_i \right| \leq \left| \frac{p}{q} \right| + |\alpha_i| \leq (|x| + 1) + (1 + M) = |x| + M + 2.$$

On pose la constante $K = |a_n|(|x| + M + 2)^{n-1}$. On obtient $\left|P\left(\frac{p}{q}\right)\right| \leq K \left|x - \frac{p}{q}\right|$. En combinant les deux inégalités, on a $\frac{1}{q^n} \leq K \left|x - \frac{p}{q}\right|$. On en déduit $\left|x - \frac{p}{q}\right| \geq \frac{1/K}{q^n}$.

Cette minoration stricte en $\frac{1}{q^n}$ rend impossible l'existence d'une infinité de rationnels vérifiant $\left|x - \frac{p}{q}\right| \leq \frac{c}{q^\mu}$ pour un $\mu > n$. On conclut que $\mu(x) \leq n$.

6. Soit $N_q = \sum_{k=0}^q 10^{-k!}$. C'est un rationnel de dénominateur $10^{q!}$.

$$|N - N_q| = \sum_{k=q+1}^{+\infty} 10^{-k!} \leq 10^{-(q+1)!} \sum_{j=0}^{+\infty} 10^{-j} \leq \frac{2}{10^{(q+1)!}} = \frac{2}{(10^{q!})^{q+1}}.$$

La définition est vérifiée pour tout entier q avec $\mu = q + 1$. La borne supérieure est $+\infty$. Le nombre N (constante de Liouville) n'est donc pas algébrique.

Solution de 2 : Centrale 2025 – Décomposition polaire

1. Immédiat ce $O_n(\mathbb{R})$ et $S_n^+(\mathbb{R})$ sont des parties du groupes multiplicatif $GL_n(\mathbb{R})$.

2. (a)

```
1 import numpy as np
2
3 def generer_diagonale_positive(n):
4     return np.diag(np.random.uniform(0, 10, n))
```

Python

(b)

```
1 from scipy.stats import ortho_group
2
3 def generer_Snplusplus(n):
4     D = generer_diagonale_positive(n)
5     P = ortho_group.rvs(dim=n)
6     return P.dot(D).dot(P.T) # ou bien P @ D @ P.T
```

Python

(c)

```
1 import numpy as np
2
3 def test_Snplusplus(A):
4     n = len(A)
5     # Vérification de la symétrie
6     B = abs(A - A.T)
7     for i in range(n):
8         for j in range(n):
9             if B[i, j] > 1e-5:
10                return False
11
12     # Vérification des valeurs propres
13     valeurs_propres = np.linalg.eigvals(A)
14     for vp in valeurs_propres:
15         if vp < -1e-5:
16             return False
17
18     return True
```

Python

Et voici une solution plus expéditive.

```
1 def test_Snplusplus(A):
2     if not np.allclose(A, A.T, atol=1e-5):
3         return False
4     valeurs_propres = np.linalg.eigvals(A)
5     return np.all(valeurs_propres > -1e-5)
```

Python

(d)

```
1 import matplotlib.pyplot as plt
2
3 dets = []
4 for _ in range(10000):
5     O = ortho_group.rvs(dim=3)
6     S = generer_Snplusplus(3)
7     dets.append(np.linalg.det(O @ S))
8
9 dets.sort()
10 plt.plot(dets)
11 plt.show()
```

Python

3. L'application μ est la restriction de l'application bilinéaire $(A, B) \mapsto AB$ de $\mathcal{M}_n(\mathbb{R})^2$ dans $\mathcal{M}_n(\mathbb{R})$. En dimension finie, toute application bilinéaire est continue.

4. (a) La matrice $M^T M$ est symétrique réelle.

Pour tout vecteur colonne X non nul, $X^T M^T M X = \|MX\|^2$. Comme M est inversible, $MX \neq 0$ et la norme est strictement positive, donc $M^T M \in S_n^+(\mathbb{R})$.

Le théorème spectral assure l'existence d'une matrice orthogonale P et d'une matrice diagonale D à coefficients strictement positifs telles que $M^T M = P D P^T$. On pose $S = P \sqrt{D} P^T$ (où \sqrt{D} contient les racines des éléments diagonaux). $S \in S_n^+(\mathbb{R})$ et $S^2 = M^T M$.

(b) On pose $O = M S^{-1}$. On vérifie que $O^T O = (S^T)^{-1} M^T M S^{-1} = S^{-1} S^2 S^{-1} = I_n$. O est orthogo-

nale et $M = OS$.

5. (a) $OS = O'S'$ donc $(OS)^T OS = (O'S')^T O'S'$ puis $S^T O^T OS = S^T O'^T O'S'$.
Comme O et O' sont orthogonales, $S^2 = S'^2$.
- (b) La matrice S'^2 est symétrique réelle, donc diagonalisable. Soient $\lambda_1, \dots, \lambda_r$ ses valeurs propres distinctes, qui sont strictement positives. Les valeurs propres de S' sont les $\sqrt{\lambda_i}$.
Il existe un polynôme interpolateur de Lagrange P tel que $P(\lambda_i) = \sqrt{\lambda_i}$ pour tout i .
On a alors $S' = P(S'^2)$.
Comme $S^2 = S'^2$, leurs valeurs propres sont identiques. Le même polynôme s'applique. $S = P(S^2) = P(S'^2) = S'$.
On obtient donc $S = S'$.
- (c) L'égalité $S = S'$ et la relation $OS = O'S'$ impliquent immédiatement $O = O'$ (car S est inversible). L'application est injective.
6. (a) L'application $\phi : M \mapsto M^T M$ est continue. $O_n(\mathbb{R}) = \phi^{-1}(\{I_n\})$, c' est une partie fermée. De plus, pour toute matrice orthogonale O , $\|O\|^2 = \text{tr}(O^T O) = \text{tr}(I_n) = n$. L'ensemble est borné. En dimension finie, un fermé borné est compact.
- (b) L'adhérence de $S_n^{++}(\mathbb{R})$ est $S_n^+(\mathbb{R})$ (les matrices symétriques positives). Toute matrice de $S_n^+(\mathbb{R})$ est limite de la suite $M + \frac{1}{k} I_n \in S_n^{++}(\mathbb{R})$.
- (c) Soit (M_p) une suite de $GL_n(\mathbb{R})$ convergeant vers $M \in GL_n(\mathbb{R})$. Notons $(O_p, S_p) = \mu^{-1}(M_p)$ et $(O, S) = \mu^{-1}(M)$.
La suite (O_p) est à valeurs dans le groupe orthogonal $O_n(\mathbb{R})$. Ce groupe est compact d'après la question 6.a. Soit O' une valeur d'adhérence de cette suite. Il existe une extractrice φ telle que $O_{\varphi(p)} \xrightarrow{p \rightarrow +\infty} O'$.
On a $S_{\varphi(p)} = O_{\varphi(p)}^{-1} M_{\varphi(p)}$. La continuité du passage à l'inverse et du produit matriciel donne la convergence $S_{\varphi(p)} \xrightarrow{p \rightarrow +\infty} O'^{-1} M$. Posons $S' = O'^{-1} M$. La suite $(S_{\varphi(p)})$ est à valeurs dans $S_n^{++}(\mathbb{R})$. D'après 6.b, sa limite S' appartient à $S_n^+(\mathbb{R})$.
La matrice S' est le produit de deux matrices inversibles. Elle est donc inversible. Une matrice symétrique positive et inversible est définie positive. On obtient $S' \in S_n^{++}(\mathbb{R})$.
L'égalité $S' = O'^{-1} M$ s'écrit $M = O'S'$. Il s'agit d'une décomposition polaire de M . L'injectivité de μ garantie à la question 5.c impose $O' = O$ et $S' = S$.
La suite (O_p) évolue dans un compact et possède une unique valeur d'adhérence O . Elle converge donc vers O . Il s'ensuit que $S_p = O_p^{-1} M_p$ converge vers $O^{-1} M = S$.
On a bien $\mu^{-1}(M_p) \xrightarrow{p \rightarrow +\infty} \mu^{-1}(M)$.