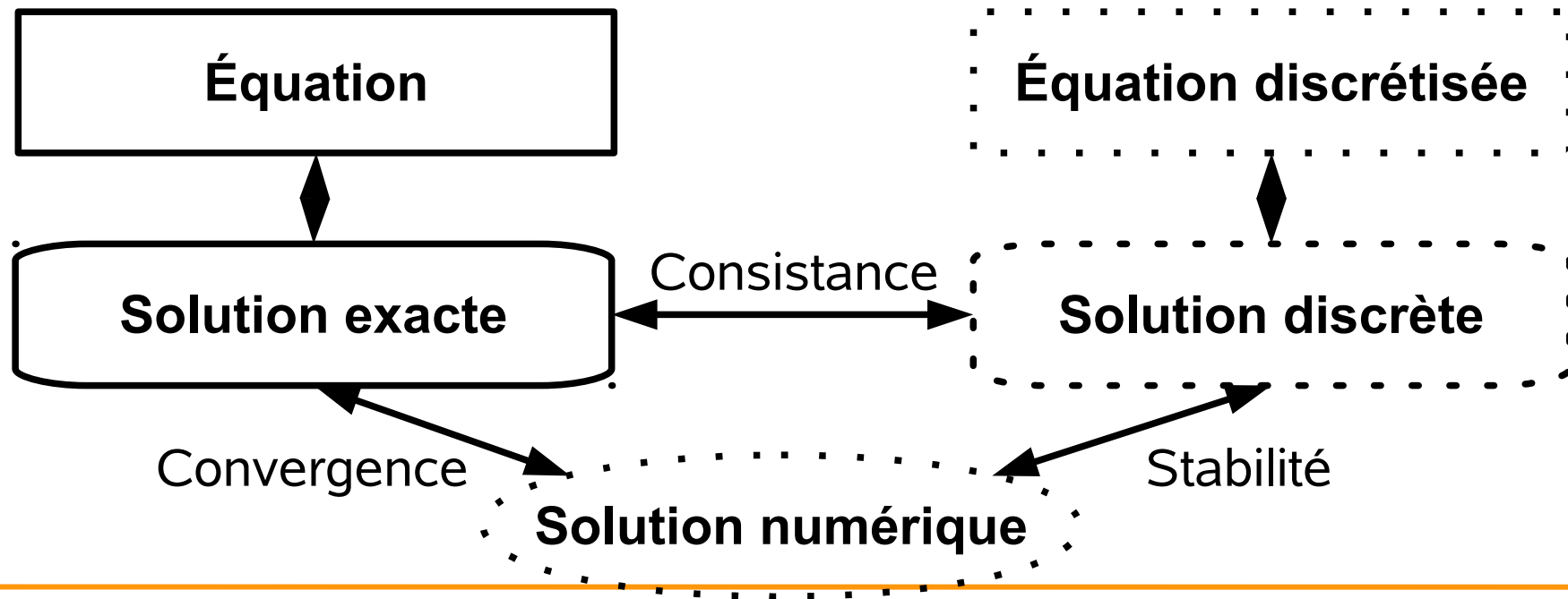


Retour sur le cours + SVD - PCA

Rappel : Convergence, Consistance et Stabilité

- Convergence : la solution numérique tend vers la solution exacte
- Consistance : solution discrète tend vers la solution exacte avec la réduction de la discrétisation
- Stabilité : différence entre solution discrète et numérique est bornée
- Relations



Rappel : Conditionnement

- Robustesses aux perturbations
 - Données d (perturbées $d + \delta d$)
 - Bruit de mesure
 - Conversion en numérique
 - Résultats x (perturbées $x + \delta x$)
 - Erreur numérique

- Conditionnement relatif

$$K_{\text{rel}}(\mathbf{d}) = \lim_{\epsilon \rightarrow 0} \sup_{|\delta \mathbf{d}| < \epsilon} \left\{ \frac{|\delta \mathbf{x}| / |\mathbf{x}|}{|\delta \mathbf{d}| / |\mathbf{d}|} \right\}$$

- Conditionnement absolu $K_{\text{abs}}(\mathbf{d}) = \lim_{\epsilon \rightarrow 0} \sup_{|\delta \mathbf{d}| < \epsilon} \left\{ \frac{|\delta \mathbf{x}|}{|\delta \mathbf{d}|} \right\}$

Rappel : système linéaire

$$\mathbf{A} \mathbf{x} = \mathbf{b}$$

- Si \mathbf{A} est une matrice inversible, le conditionnement relatif du système est

$$K_{\text{rel}}(\mathbf{b}) \leq K(\mathbf{A}) = \|\mathbf{A}^{-1}\| \|\mathbf{A}\|$$

$K(\mathbf{A})$ = conditionnement de la matrice

- Matrice symétrique définie positive

– Conditionnement = Rapport des valeurs propres

$$K(\mathbf{A}) = \frac{\lambda_{\max}}{\lambda_{\min}}$$

Rappel : résoudre un système linéaire

- Méthodes directes
 - Si A est diagonale : $O(n)$
 - Si A est triangulaire : $O(n^2)$
 - Sinon : triangulation ou factorisation
 - LU : matrice quelconque – $O(2/3 n^3)$
 - Cholesky : matrice symétrique définie positive – $O(1/3 n^3)$
 - Householder et QR :
 - Limites : Grand conditionnement \rightarrow solution peu précise
- Méthodes itératives : convergence vers une solution
 - Jacobi : A à diagonale dominante stricte par ligne
 - Gauss-Seidel : A à diagonale dominante stricte par ligne et symétrique définie positive
 - Gradient conjugué

Ne jamais résoudre avec
l'inverse de la matrice A

Rappel : optimisation

- Approcher des données par une fonction analytique $\mathbf{f}_v(\mathbf{x})$

- Donnée : ($\mathbf{x}_m=1..M$, $\mathbf{y}_m=1..M$)

- \mathbf{v} : paramètres de la fonction

- \mathbf{v} minimise une erreur : exemple
ou maximise un qualité

$$\min_{\mathbf{v}} \sum_{m=1}^M w_m \left\| \mathbf{y}_m - \mathbf{f}_v(\mathbf{x}_m) \right\|$$

- Sous des contraintes

- Optimisation linéaire

- \mathbf{f}_v est une combinaison linéaire de fonctions

$$\mathbf{f}_v(\mathbf{x}) = \sum_{k=1}^K v_k \mathbf{f}_k(\mathbf{x})$$

- L'erreur est quadratique et les contraintes sont linéaires

- Optimisation non-linéaire

- Tous les autres cas

Rappel : moindres carrés

- L'erreur est une norme euclidienne $E = \sum_{m=1}^M w_m \left\| \mathbf{y}_m - \mathbf{f}_v(\mathbf{x}_m) \right\|_2^2$
 - Un minimum atteint →
gradient de l'erreur est nul $\nabla_v E = 0$
 - Ajout de contrainte par des multiplicateurs (ex : Lagrange)
- Optimisation linéaire
 - Résoudre une système linéaire sous contraintes linéaires
 - Solution unique
- Optimisation non-linéaire
 - Solution unique seulement si l'erreur est convexe
 - En général : minimum local
 - Processus itératifs convergeant vers un minimum
 - Méthodes de descente de gradient
 - Levenberg-Marquard

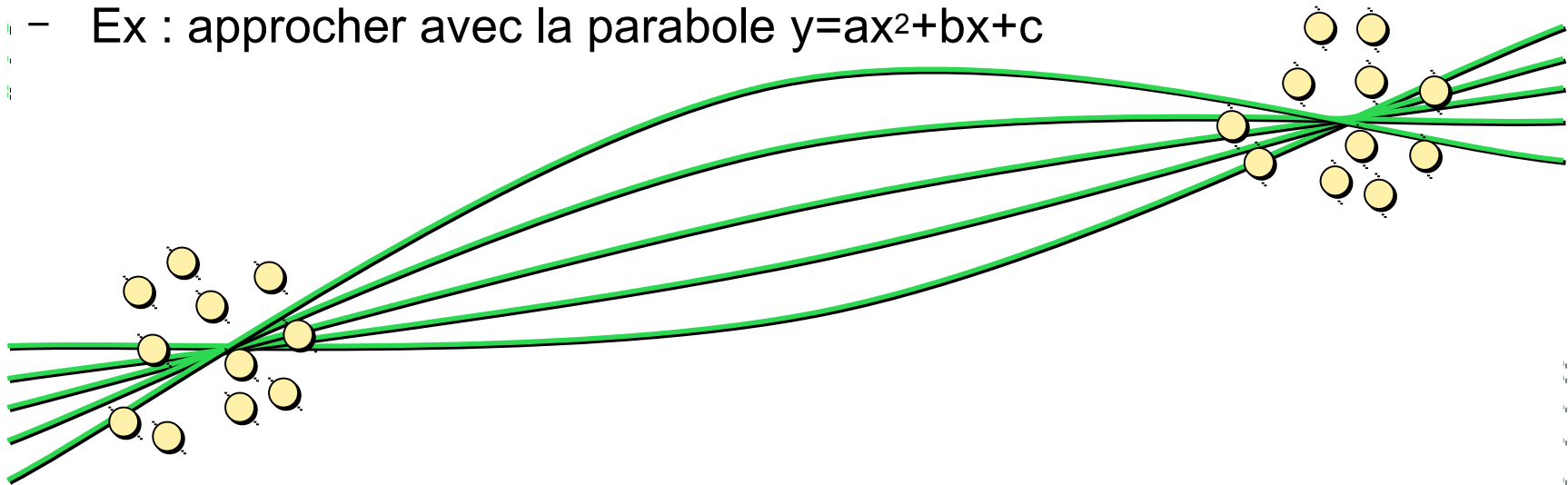
Valeurs propres

SVD

PCA

Moindres carrés sous-constraint

- Moins de mesures que de variables
 - Impossible théoriquement d'utiliser les moindres carrés
 - Le système à résoudre a la forme $A^T A \mathbf{x} = A^T \mathbf{b}$
 - A a plus de colonnes que de lignes
 - $A^T A$ est donc singulière
- Les mesures sont de mauvaises contraintes
 - $A^T A$ est presque singulière : mauvais conditionnement
 - Ex : approcher avec la parabole $y = ax^2 + bx + c$



Décomposition singulière SVD

- Soit une matrice A de dimension $M \times N$
 - Il existe 2 matrices orthogonales U ($M \times M$) et V ($N \times N$) et une matrice diagonale Σ de dimension $P = \min(M, N)$ telle que

$$A = U \Sigma V^T \Leftrightarrow \Sigma = U^T A V$$

– Avec $\Sigma = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \sigma_P \end{bmatrix} \quad \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_P \geq 0$

- Les σ_i sont les valeurs singulières
- Les colonnes de U et V sont les vecteurs singuliers de gauche et de droite

Interprétations

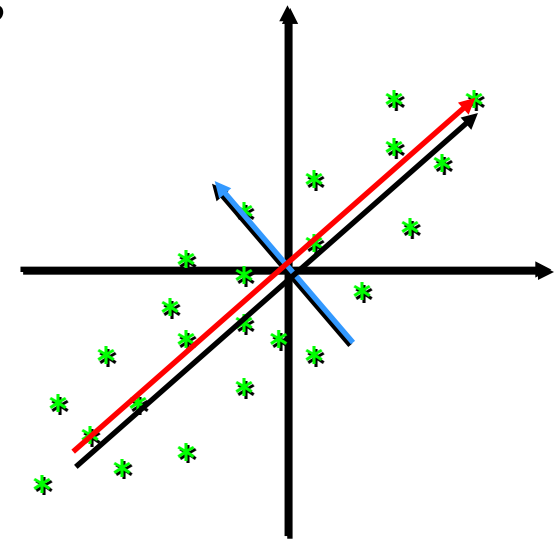
- Lien avec les valeurs propres

$$\Sigma^T \Sigma = (U^T A V)^T U^T A V = V^T A^T U U^T A V = V^T A^T A V$$

– Donc :

- Les σ_i^2 sont les valeurs propres de $A^T A$ et $A A^T$
 - Les colonnes de V sont les vecteurs propres
- Approximation des données
 - Demi-longueurs des axes de l'hyperellipse

$$\mathbf{x}^T A^T A \mathbf{x} - 1 = 0$$



- Conditionnement

$$K(A) = \frac{\sigma_{max}}{\sigma_{min}} \quad K(A) = \frac{\sigma_1}{\sigma_p}$$

Calcul de pseudo inverse

- On définit la matrice

$$A^+ = V \Sigma^+ U^T$$

$$\Sigma^+ = \begin{bmatrix} \frac{1}{\sigma_1} & 0 & \dots & \dots & \dots & \dots & 0 \\ 0 & \ddots & \ddots & & & & \vdots \\ \vdots & \ddots & \frac{1}{\sigma_R} & \ddots & & & \vdots \\ \vdots & & \ddots & 0 & \ddots & & \vdots \\ \vdots & & & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & \dots & 0 & 0 & 0 \end{bmatrix}$$

- Principes
 - On annule les valeurs inverses trop grandes
 - On laisse libre un certain nombre de paramètres

Applications

- Avantage : méthode numérique stable
 - Coût : $O(n^3)$
 - Plus rapide si l'on ne cherche que Σ , U et Σ , ou Σ et V
- Calcul du rang R

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_R > \sigma_{R+1} = \dots = \sigma_P = 0$$

$$A = \sum_{i=1}^P \sigma_i \mathbf{u}_i \mathbf{v}_i^T$$

- En matlab

```
s=svd(X);
```

```
tol=max(size(X))*s(1)*eps;
```

```
rang=sum(s > tol);
```

- Approximation par une somme de matrice de rang K

$$A_K = \sum_{i=1}^K \sigma_i \mathbf{u}_i \mathbf{v}_i^T \quad \left\| A - A_K \right\|_2 = \sigma_{K+1}$$

Analyse en composantes principales (PCA)

- Une méthode de réduction de dimension des données
 - Projection des données au sens des moindres carrés
 - Capturer les principales variations

- Une première approximation $E_0 = \min_a \sum_{m=1}^M \|\mathbf{y}_m - \mathbf{a}\|_2^2$
 - Meilleur représentant au sens des moindres carrés

$$\mathbf{a} = \frac{1}{M} \sum_{m=1}^M \mathbf{y}_m$$

- Quelles sont les ou les directions principales ?
- Représenter les données par une ligne passant par la moyenne

$$\mathbf{y}_m = \mathbf{a} + \alpha_m \mathbf{e}$$

$$E_1 = \min_{\alpha_1, \dots, \alpha_M} \sum_{m=1}^M \|\mathbf{y}_m - \mathbf{a} - \alpha_m \mathbf{e}\|_2^2$$

PCA – Intuitions (suite)

- Calculer les α_m par différentiation partielle $\partial_{\alpha_m} E_1 = 0 \Leftrightarrow \alpha_m = \mathbf{e}^T (\mathbf{y}_m - \mathbf{a})$
- Nouvelle erreur pour calculer la direction principale

$$\begin{aligned} E_1 &= \sum_{m=1}^M \|\mathbf{y}_m - \mathbf{a} - \alpha_m \mathbf{e}\|_2^2 = \sum_{m=1}^M \alpha_m^2 - 2 \sum_{m=1}^M \alpha_m \mathbf{e}^T (\mathbf{y}_m - \mathbf{a}) + \sum_{m=1}^M \|\mathbf{y}_m - \mathbf{a}\|^2 \\ &= \sum_{m=1}^M \|\mathbf{y}_m - \mathbf{a}\|^2 - 2 \sum_{m=1}^M \mathbf{e}^T (\mathbf{y}_m - \mathbf{a}) (\mathbf{y}_m - \mathbf{a})^T \mathbf{e} + \sum_{m=1}^M \|\mathbf{y}_m - \mathbf{a}\|^2 \\ &= -2 \sum_{m=1}^M \mathbf{e}^T (\mathbf{y}_m - \mathbf{a}) (\mathbf{y}_m - \mathbf{a})^T \mathbf{e} + 2 \sum_{m=1}^M \|\mathbf{y}_m - \mathbf{a}\|^2 \\ &= -2 \mathbf{e}^T \mathbf{S} \mathbf{e} + 2 \sum_{m=1}^M \|\mathbf{y}_m - \mathbf{a}\|^2 \end{aligned}$$

– S matrice de covariance (*scatter matrix*)

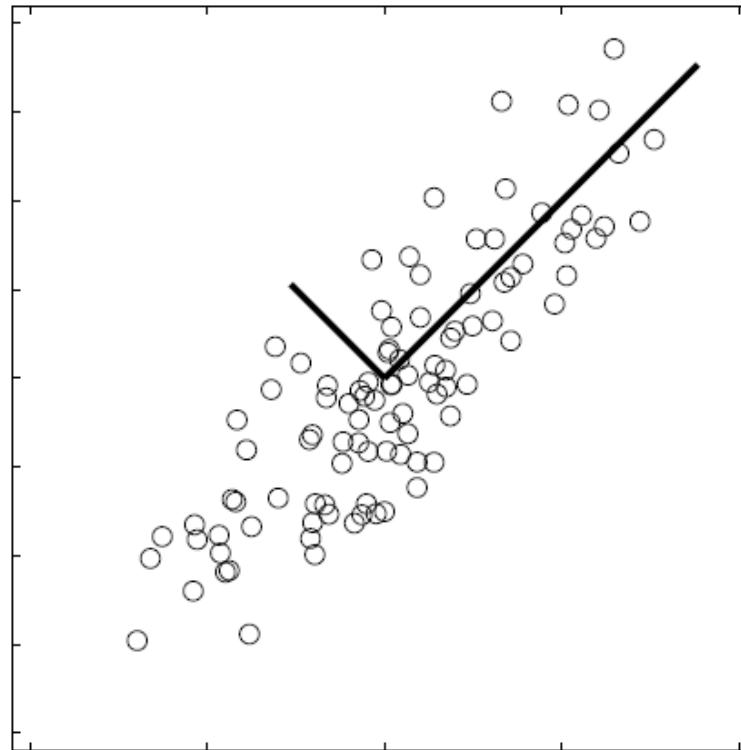
$$\mathbf{S} = \sum_{m=1}^M (\mathbf{y}_m - \mathbf{a})(\mathbf{y}_m - \mathbf{a})^T$$

- Minimisation sous contrainte

$$\begin{aligned} \min_{\mathbf{e}} E_1 &\Leftrightarrow \max_{\mathbf{e}} \mathbf{e}^T \mathbf{S} \mathbf{e} \\ &\|\mathbf{e}\|^2 = 1 \end{aligned}$$

PCA – Intuitions (fin)

- Utilisation des multiplicateurs de Lagrange $E_3 = \mathbf{e}^T \mathbf{S} \mathbf{e} - \lambda (\mathbf{e}^T \mathbf{e} - 1)$
- Par dérivation $\nabla_{\mathbf{e}} E_3 = 0 \Leftrightarrow \mathbf{S} \mathbf{e} = \lambda \mathbf{e}$
 - \mathbf{e} est un vecteur propre de la matrice \mathbf{S}
- Peut-être étendue à M directions principales
 - Calcul d'une SVD
 - Amplitude de variation = valeur propre
 - Direction de variation = vecteur propre



Rappel - suite

Rappel – autre optimisation

- Programmation linéaire
 - Minimiser une norme infinie

$$\begin{aligned} \min_{\mathbf{v}} \mathbf{d}^T \mathbf{v} \\ \text{subject to } \mathbf{c}_m^T \mathbf{v} \leq b_m \end{aligned}$$

- Maximiser un objectif

$$\begin{aligned} \max_{\mathbf{v}} \mathbf{d}^T \mathbf{v} \\ \text{subject to } \begin{cases} \mathbf{c}_m^T \mathbf{v} + e_m = b_m & \forall m \\ v_k \geq 0 & \forall k \\ e_m \geq 0 & \forall m \end{cases} \end{aligned}$$

$$\text{with } \begin{cases} \mathbf{v} = (v_1 \quad \dots \quad v_k)^T \\ \mathbf{d} = (d_1 \quad \dots \quad d_k)^T \\ \mathbf{c}_m = (c_{m1} \quad \dots \quad c_{mk})^T \end{cases}$$

- Programmation quadratique
 - Norme euclidienne
 - Contraintes linéaires d'inégalité

$$\begin{aligned} \min_{\mathbf{v}} \mathbf{v}^T \mathbf{Q}^T \mathbf{v} + \mathbf{d}^T \mathbf{v} \\ \text{subject to } \begin{cases} \mathbf{c}_j^T \mathbf{v} = b_j \\ \mathbf{c}_m^T \mathbf{v} \leq b_m \end{cases} \end{aligned}$$

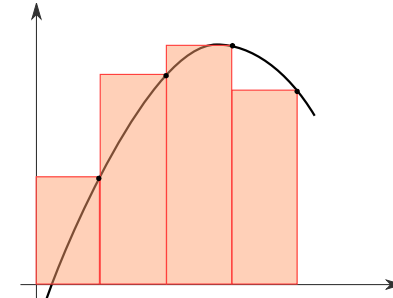
Rappel - intégration numérique 1D

- Intervalles uniformes

Convergence

- Méthode des rectangles

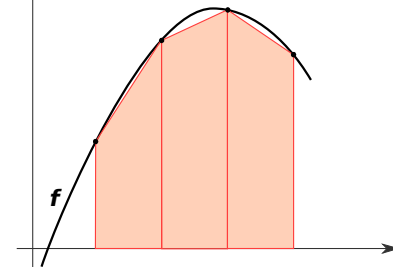
$$I_n = \frac{b-a}{n} \sum_{k=1}^n f(x_k)$$



$O(n^{-1})$

- Méthode des trapèzes

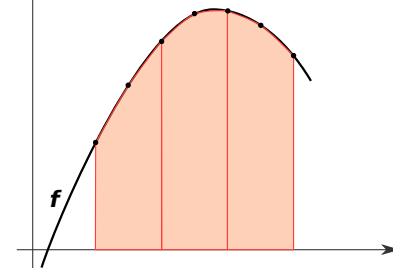
$$I_n = \frac{b-a}{2n} \sum_{k=1}^n (f(x_{k-1}) + f(x_k))$$



$O(n^{-2})$

- Formule de Simpson

$$I_n = \frac{b-a}{6n} \sum_{k=1}^n (f(x_{k-1}) + 4f(x_{k-1/2}) + f(x_k))$$



$O(n^{-4})$

Rappel - quadrature de Gauss en 1D

$$I = \int_{-1}^1 f(x) dx \simeq \sum_{m=1}^M W_m f(\varepsilon_m)$$

Poids

Positions d'intégration

- Les poids et les positions **maximisent la précision**
 - M positions pour l'espace des polynômes de degré $2M-1$
 - Le calcul des intégrales est exact sur cet espace

$$\forall p \in P_{2M-1}, \int_{-1}^1 p(x) dx = \sum_{m=1}^M W_m p(\varepsilon_m)$$

- Les positions sont les racines du polynôme de Legendre
 - Base de polynômes orthogonale sur $[-1,1]$

$$P_M(x) = \frac{1}{2^M M!} \frac{d^M}{dx^M} \left((x^2 - 1)^M \right)$$

Rappel – intégration de Monte Carlo 1D

- Estimateur

$$I_n = \frac{1}{n} \sum_{i=1}^n \frac{1}{\text{pdf}(\mathbf{x}_i)} f(\mathbf{x}_i)$$

- Densité de probabilité

$$\text{pdf}(x) \geq 0$$

- Probabilité associée $P[X_i \in [x, x+dx]] = \text{pdf}(x) dx$

- Propriété

$$\int_{-\infty}^{\infty} \text{pdf}(x) dx = 1$$

- Convergence

- Variance $V[I_n] = \frac{1}{n} V[I_1]$

- Écart-type $\sigma[I_n] = \frac{1}{\sqrt{n}} \sigma[I_1]$

- Meilleur choix de pdf

$$\text{pdf}(x) = \frac{1}{I} f(x) \Rightarrow V[I_n] = 0$$

Rappel – intégration Monte Carlo

- Biais

- Différence valeur attendue vs cherchée

$$\text{biais} = E[I_n] - I$$

- Estimateur sans biais

$$\text{biais} = 0$$

Rappel : convergence et dimension

- En 1D
 - Convergence en $O(n^{-(c+1)})$
 - c = continuité de la fonction
- En dimension plus élevée d
 - Pour n valeurs / mailles
 - Taille intervalle 1D $n^{1/d}$
 - Convergence en $O(n^{-c/d})$
 - Monte Carlo en $O(n^{-1/2})$

Rappel : espace de fonctions

- Produit scalaire

- Norme $\|f\|_{[a,b]}^2 = \int_a^b f^2(x) dx$

$$\langle f, g \rangle_{[a,b]} = \int_a^b f(x)g(x) dx$$

- Orthogonalité $\langle f, g \rangle_{[a,b]} = 0$

- Matrice de Gram

- Det = 0 \rightarrow pas une base

- Diagonale \rightarrow orthogonale

- Vecteurs propres avec valeurs propres non-nulles \rightarrow base orthogonale.

$$M = \begin{bmatrix} |f_1|^2 & \langle f_1, f_2 \rangle & \cdots & \langle f_1, f_K \rangle \\ \langle f_2, f_1 \rangle & |f_2|^2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \langle f_{K-1}, f_K \rangle \\ \langle f_K, f_1 \rangle & \cdots & \langle f_K, f_{K-1} \rangle & |f_K|^2 \end{bmatrix}$$

- Partition de l'unité

- $\sum_i f_i = 1$

- Exercice : base orthogonale

- x^n sur $[a,b]$

Rappel : bases de fonctions

- 1D – polynomiales

- Tchebychev 1 : orthogonale sur $[-1,1]$

$$T_n(\cos(\theta)) = \cos(n\theta)$$

$$\langle f, g \rangle = \int_{-1}^1 \frac{f(x)g(x)}{\sqrt{1-x^2}} dx$$

- Bernstein : orthogonale sur $[0,1]$

$$b_{k,N}(t) = \binom{N}{k} t^k (1-t)^{N-k} \quad \text{avec } k=0 \dots N$$

- 1D – polynomiales par morceaux

- Support compact

- Gram à diagonales

- B-Splines de degré P

$$B_{p,N}(t) = \sum_{i=p}^{p+N+1} \left[\prod_{j=p, j \neq i}^{p+N+1} \frac{1}{t_j - t_i} \right] \max^N(t - t_i, 0)$$

- 2D – sur un disque

- Zernike : orthonormale

- 2D – directions

- Harmonique sphérique : orthonormale

Rappel – résolution EDP - EDO

- Différences finies : calculer des valeurs en des positions
 - Approximation des dérivées d'ordre k

- Avant

$$\partial_{x^n}^n \mathbf{u}_i = \frac{1}{\Delta x^n} \sum_{j \geq i} u_j + o(\Delta x^k)$$

- Arrière

$$\partial_{x^n}^n \mathbf{u}_i = \frac{1}{\Delta x^n} \sum_{j \leq i} u_j + o(\Delta x^k)$$

- Centrée

$$\partial_{x^n}^n \mathbf{u}_i = \frac{1}{\Delta x^n} \sum_{j=-p}^p u_{i+p} + o(\Delta x^k)$$

- Résolution d'un système linéaire

- Explicite : pour chaque itération k

$$\mathbf{u}^{(k+1)} - \mathbf{f}(\mathbf{u}^{(k)}, \dots, \mathbf{u}^{(0)}) = 0$$

- Implicite : pour chaque itération k

$$\mathbf{f}(\mathbf{u}^{(k+1)}, \mathbf{u}^{(k)}, \dots, \mathbf{u}^{(0)}) = 0$$

- En général : stable

Rappel – résolution EDP - EDO

- Éléments finies

- Coefficients dans un base de fonctions

$$u(\mathbf{x}) \simeq \sum_{k=1}^K v_k \mathbf{b}_k(\mathbf{x})$$

- Étapes

1. Forme faible

$$\forall b, \int b \operatorname{res}(u) = 0$$

2. Intégration par partie

3. Discrétisation

$$\forall b_i, \int b_i \operatorname{res} \left(\sum_{k=1}^K v_k \mathbf{b}_k(\mathbf{x}) \right) = 0$$