
Machine Learning

N. Hascoët – Prof. F. Chinesta

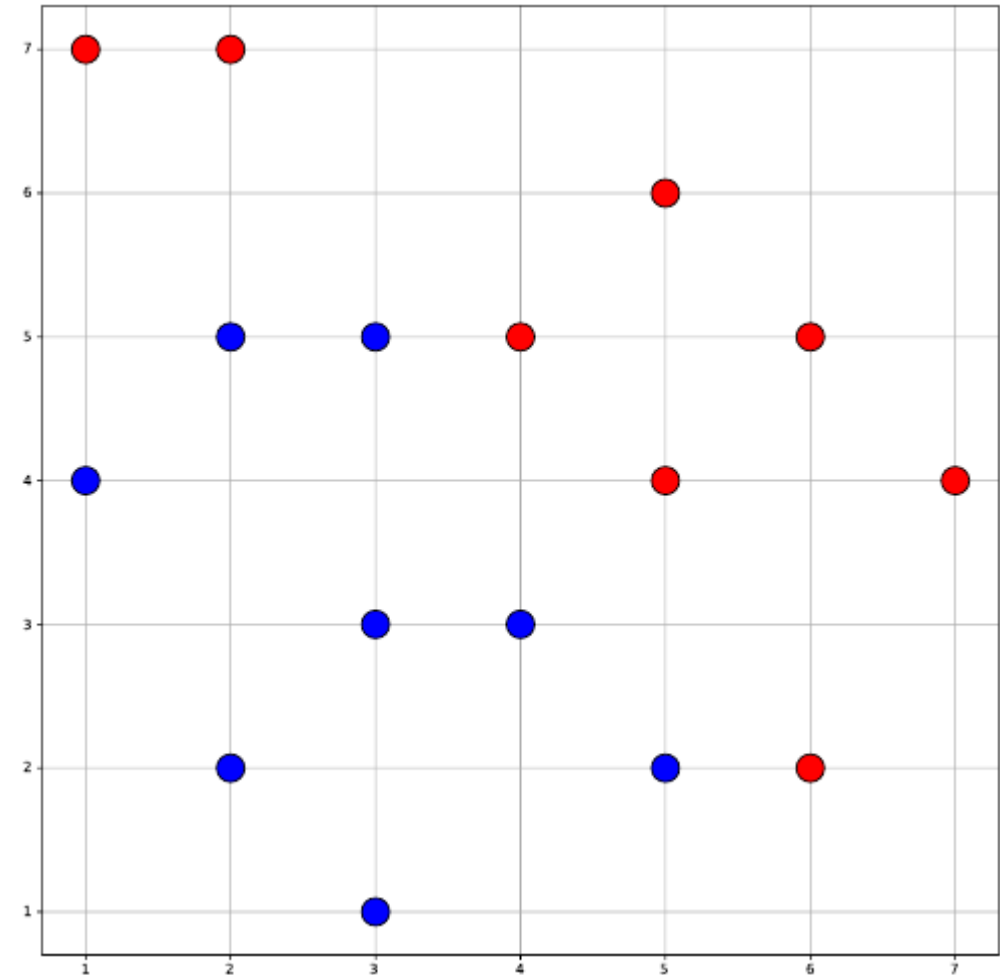
4 juillet 2019

Classification

- Apprentissage **SUPERVISÉ**
- Une partie des données est étiquetée/**LABELLISÉ**
- Première étape : **ENTRAÎNEMENT** (offline)
- Deuxième étape : **PRÉDICTION**/test (online)

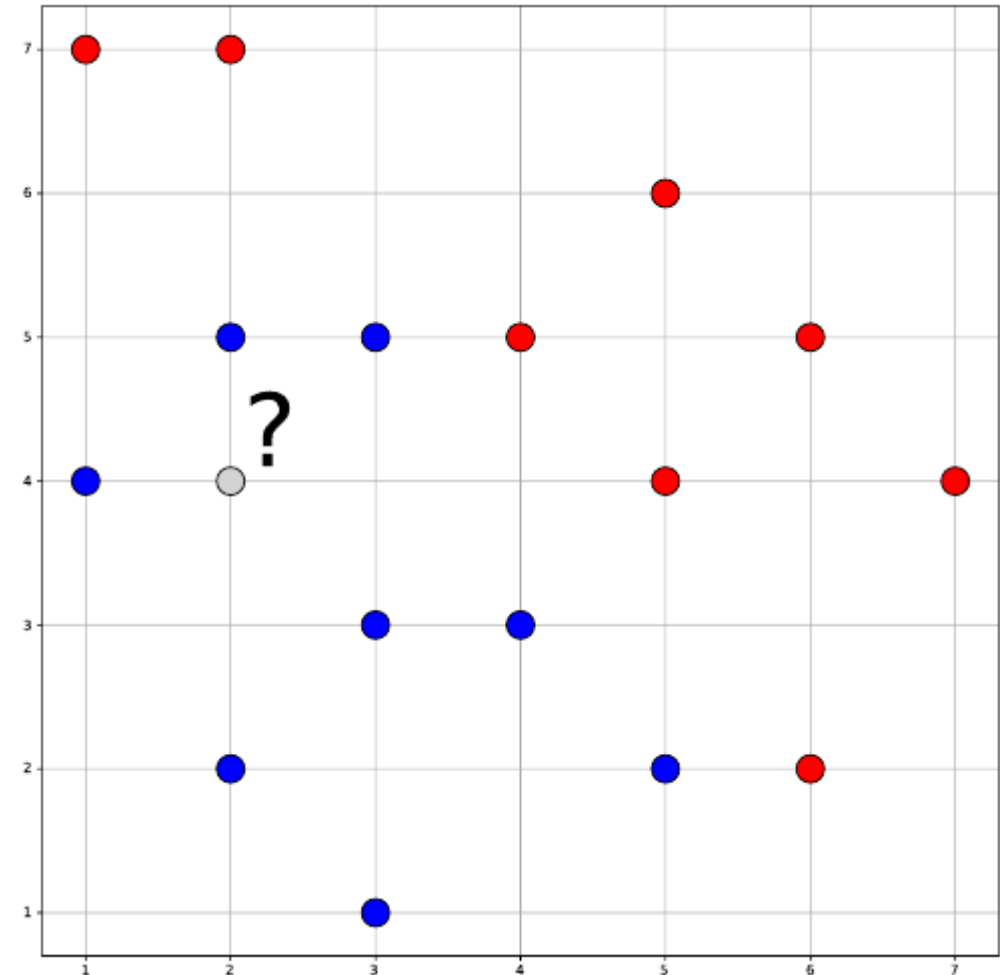
Classification

- Données d'entraînement **CONNUES** (étiquetées sur deux classes)



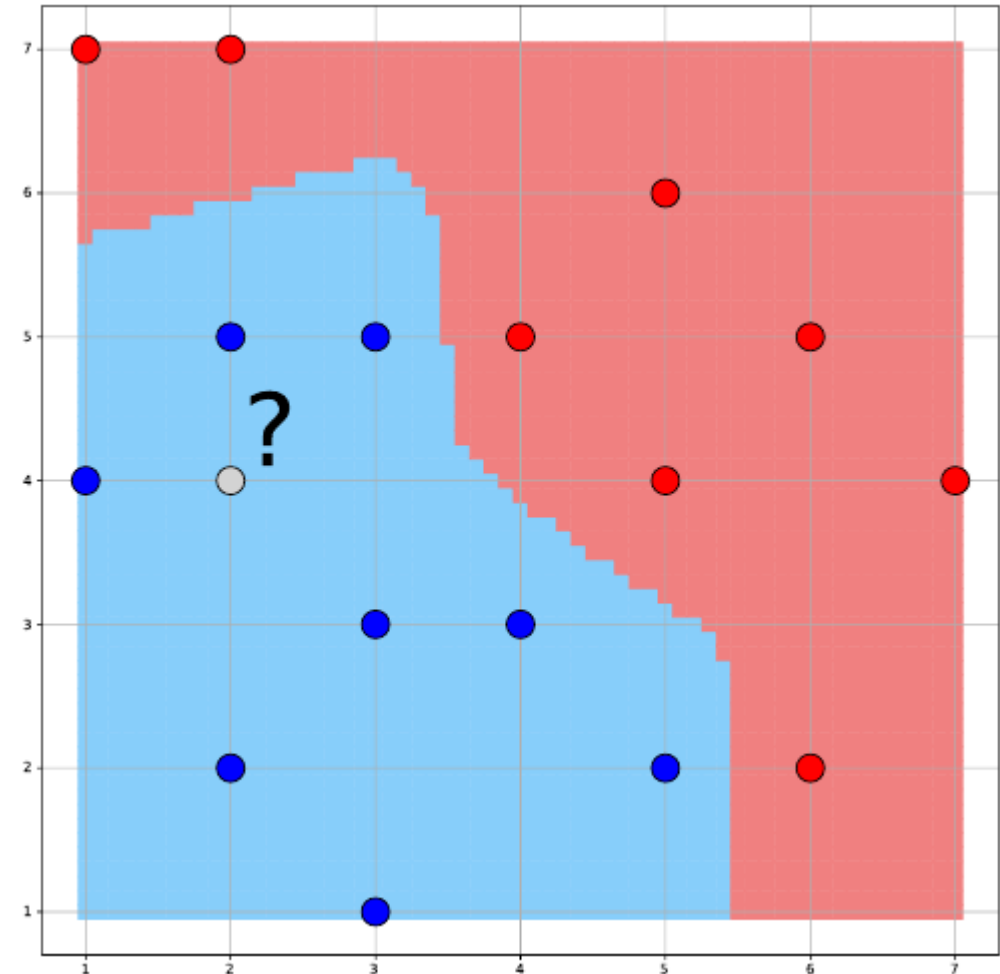
Classification

- Données d'entraînement **CONNUES** (étiquetées sur deux classes)
- Une nouvelle donnée **INCONNUE**



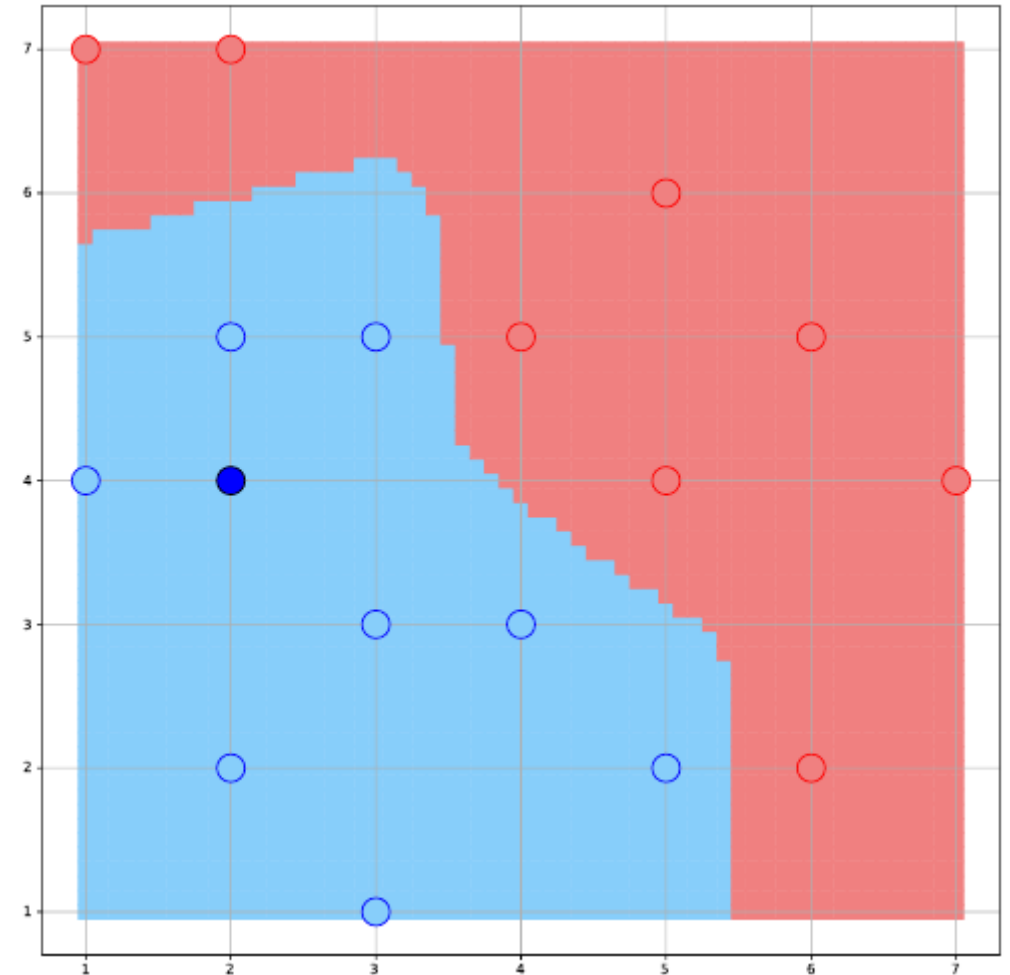
Classification

- Données d'entraînement **CONNUES** (étiquetées sur deux classes)
- Une nouvelle donnée **INCONNUE**
- Génération d'un modèle à partir des données d'entraînement ⇒ **FRONTIÈRE DE DÉCISION**



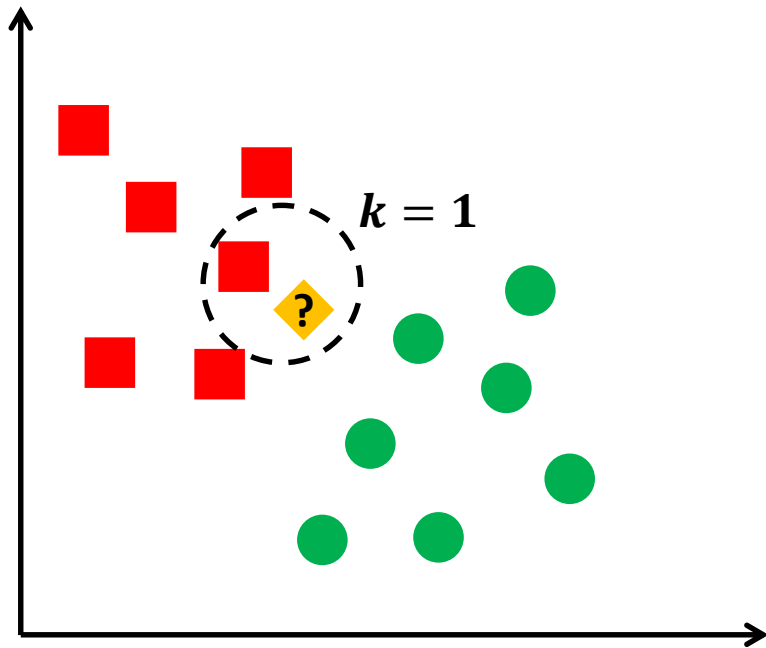
Classification

- Données d'entraînement **CONNUES** (étiquetées sur deux classes)
- Une nouvelle donnée **INCONNUE**
- Génération d'un modèle à partir des données d'entraînement \Rightarrow **FRONTIÈRE DE DÉCISION**
- **PRÉDICTION** de la donnée inconnue dans l'une des deux classes

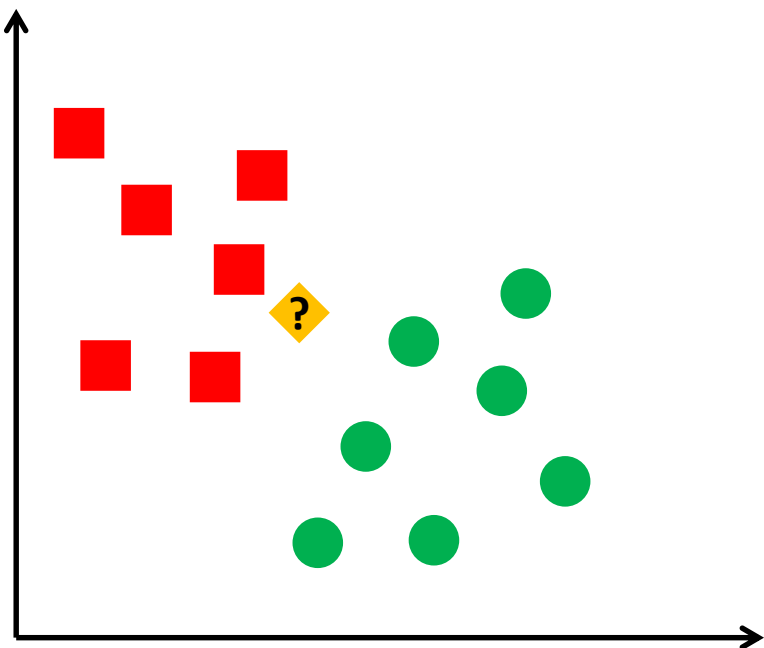


Méthode des k plus proches voisins

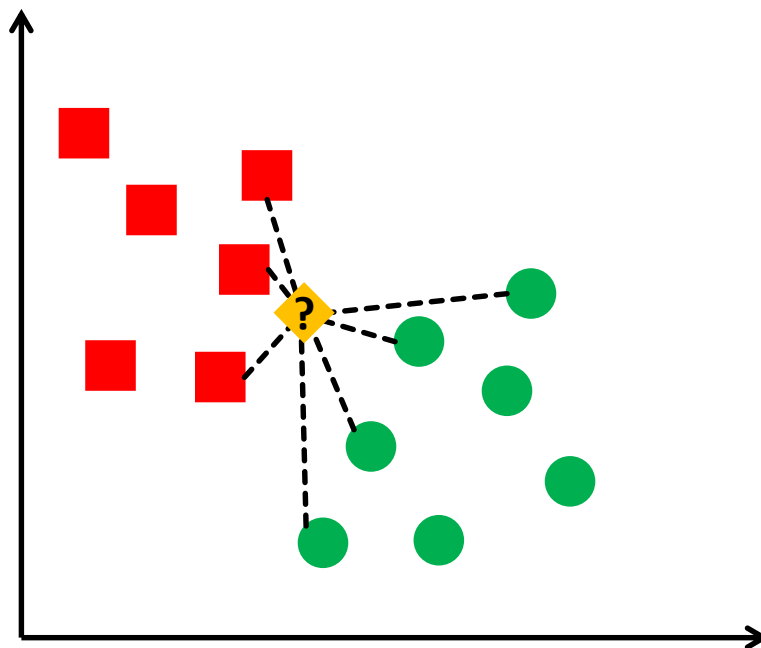
- En paramètre : k , le nombre de plus proches voisins utilisés pour déterminer la classe



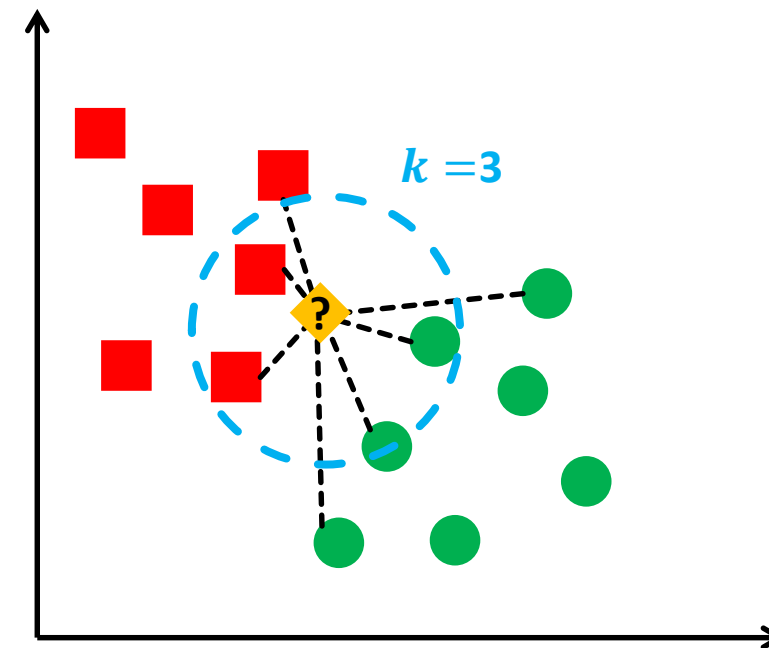
Méthode des k plus proches voisins



État initial



Calcul des distances



Recherche des voisins
Vote et choix

Méthode des k plus proches voisins

- Efficace à FAIBLE DIMENSION
- Risque d'OVERFITTING à plus haute dimension (nécessite plus de données)
- Si k est petit \Rightarrow forte influence du BRUIT
- Si k est grand \Rightarrow augmentation du TEMPS DE CALCUL

Classification naïve bayésienne

- Théorème de Bayes

$$P(y|x_1, \dots, x_n) = \frac{P(y)P(x_1, \dots, x_n|y)}{P(x_1, \dots, x_n)} \quad (1)$$

- Prédiction par le calcul de la probabilité d'un évènement y donné
 - Étape 1 : Calcul des probabilités *a priori* des classes ($P(y), P(x_1, \dots, x_n)$)
 - Étape 2 : Calcul des probabilités conditionnelles *a posteriori* ($P(x_1, \dots, x_n|y)$)
 - Étape 3 : Calculer la probabilité conditionnelle de l'évènement y avec (1)
 - Étape 4 : Sélectionner la classe avec la plus haute probabilité

Classification naïve bayésienne

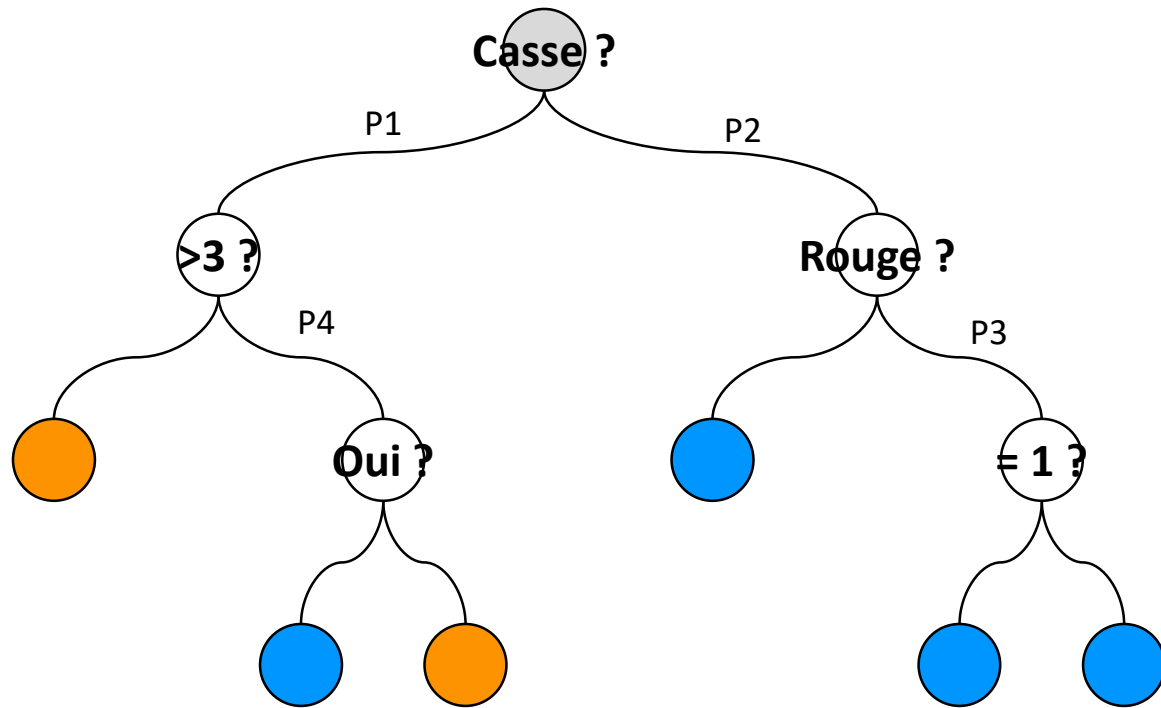
- Faible **COÛT DE CALCUL**
- Adapté pour des valeurs **DISCONTINUES** (classification)
- Présuppose l'**INDÉPENDANCE** des différentes paramètres dans les données d'entrée
- Nécessite un jeu de donnée **COMPLET**
(Problématique de la « zéro probabilité »)

Arbre de décision

- Représentation sous forme d'arbre
 - Branche : paramètre d'entrée
 - Nœud : test sur le paramètre
 - Feuille : résultat dans une des classes

- Partitionnement binaire récursif

Arbre de décision



- Plusieurs choix de paramètre de partitionnement \Rightarrow FONCTION DE COÛT
- Pour un arbre le plus ÉQUILIBRÉ possible
- ALGORITHME GLOUTON

Arbre de décision

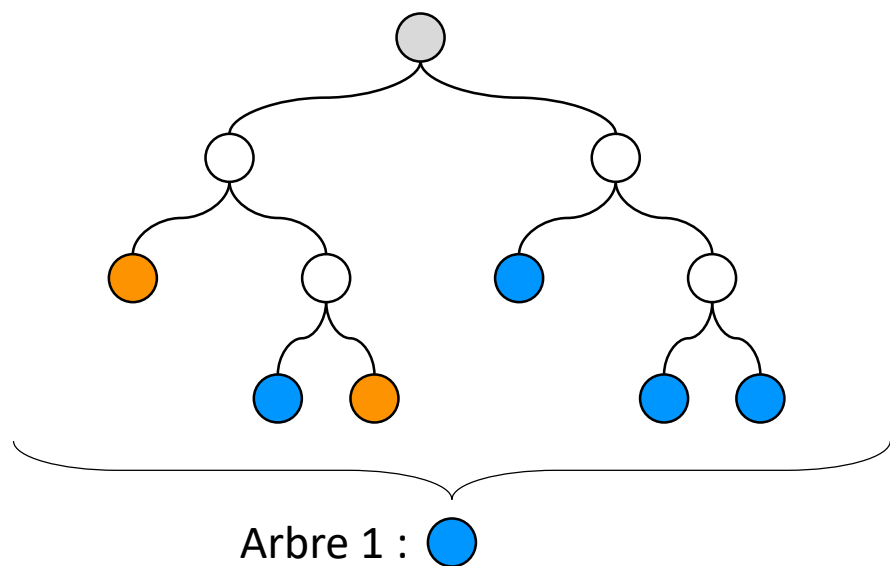
- Donne implicitement l'importance de chaque paramètre
- Gère des données numériques et catégorielles
- Visualisation simple

- Peut créer des arbres difficilement généralisables
- Algorithme glouton
- Risque de minimum local

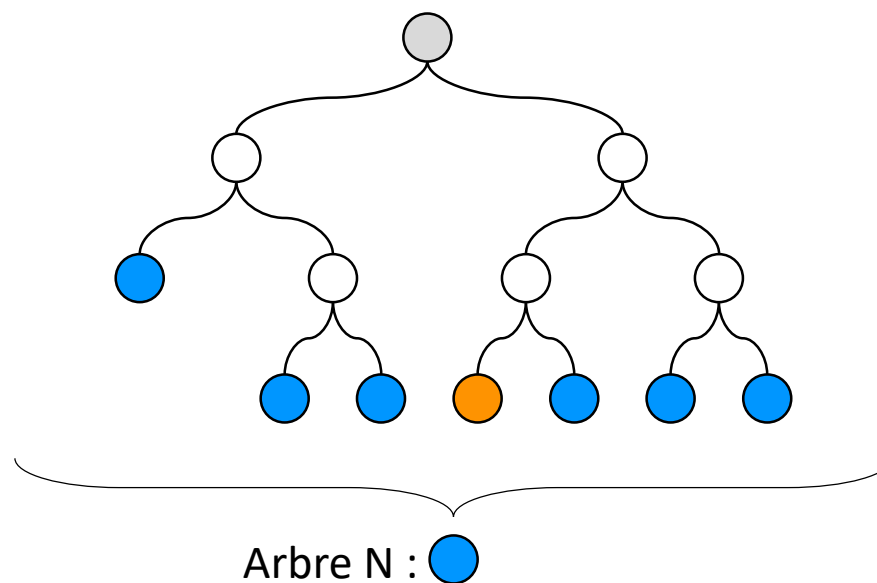
Forêt d'arbres décisionnels

- Basé sur plusieurs arbres de décision : une **FORÊT**
- Créer des arbres les plus décorrélés possibles **ALÉATOIREMENT**
- Prédiction par **VOTE**
- La **DÉCISION COLLECTIVE** protège de possibles **ERREURS INDIVIDUELLES**

Forêt d'arbres décisionnels



...



Classification Support Vector Machine

- FONCTION DE DÉCISION (dans le cas d'une séparation linéaire) :

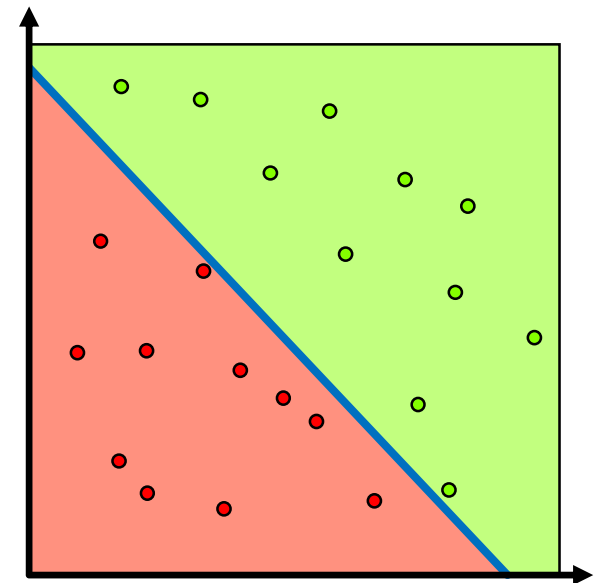
$$D(x) = \text{signe}(f(x))$$

- a et b satisfont l'équation pour les données d'ENTRAÎNEMENT (x_i, y_i) :

$$D(x_i) = y_i$$

- Calcul d'une frontière de séparation :

$$\Delta(a, b) = \{x \in \mathbb{R} \mid ax + b = 0\}$$



Classification Support Vector Machine

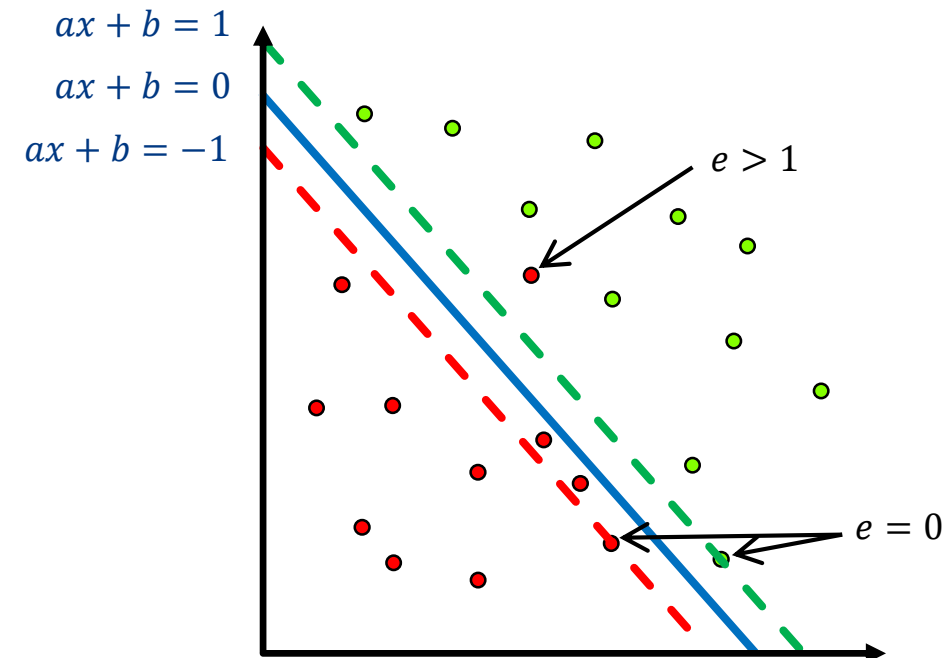
- Entraînement : optimisation $\min_a \|a\|^2$ tel que $y_i(ax_i + b) \geq 1$
- $e_i \geq 0$: distance du point i à la FRONTIÈRE
- Frontière « souple » :

$$\min_a \|a\|^2 + C \sum_{i=1}^N e_i$$

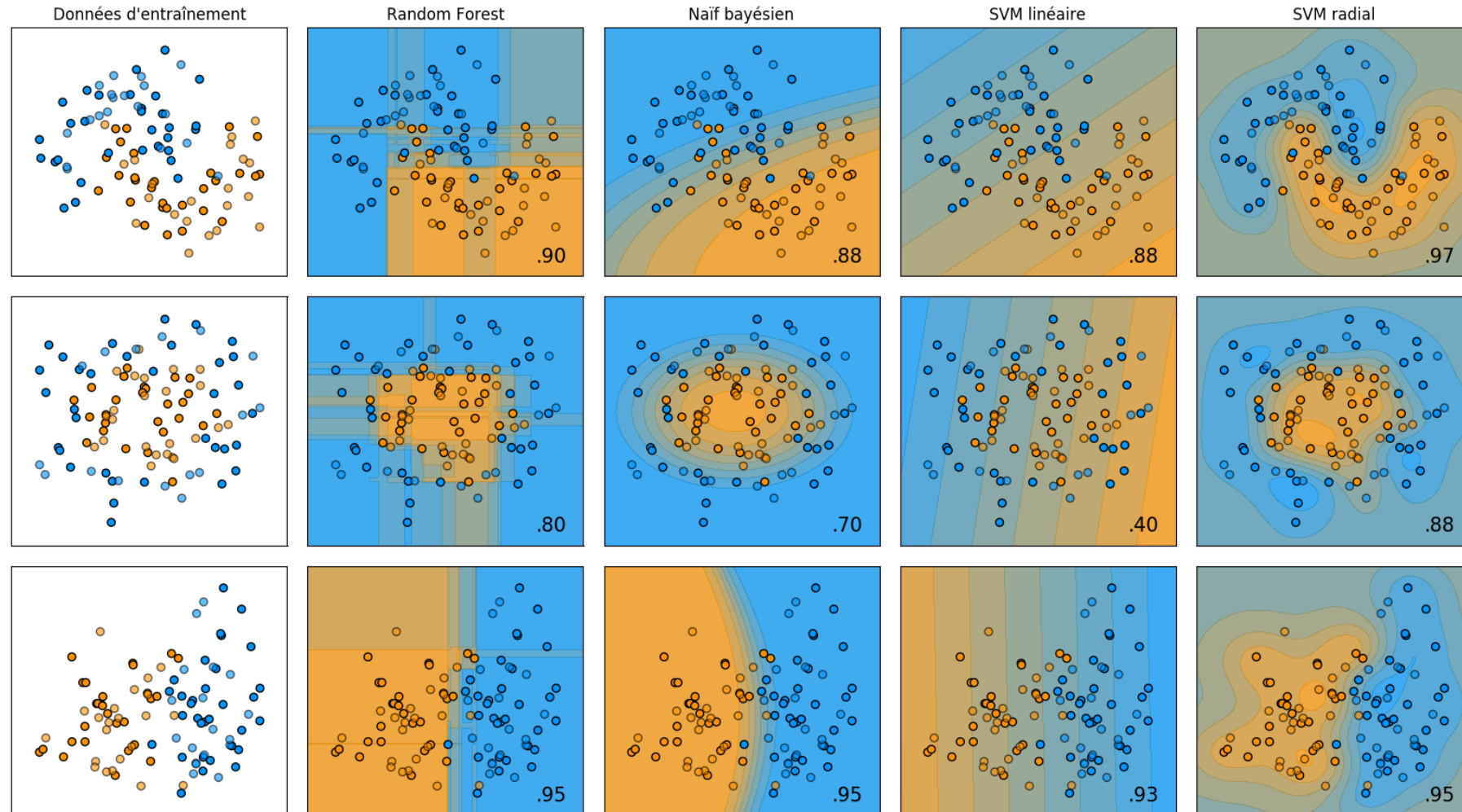
- Tel que $y_i(ax_i + b) \geq 1 - e_i$
- Equations des NOYAUX SVM :
 - Linéaire : $k(u, v) = u' \times v$
 - Polynomial : $k(u, v) = (\gamma u' \times v + r)^\delta$
 - Radial : $k(u, v) = \exp(-\gamma |u - v|^2)$

avec C le paramètre de PÉNALISATION

avec γ le paramètre d'INFLUENCE

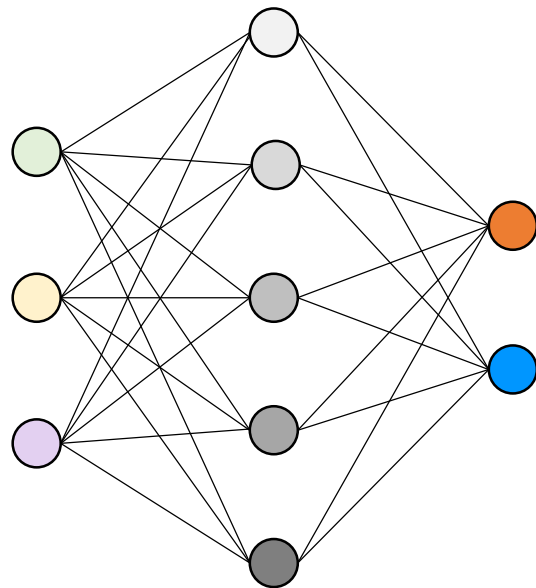


Classification Support Vector Machine



Réseau de neurones

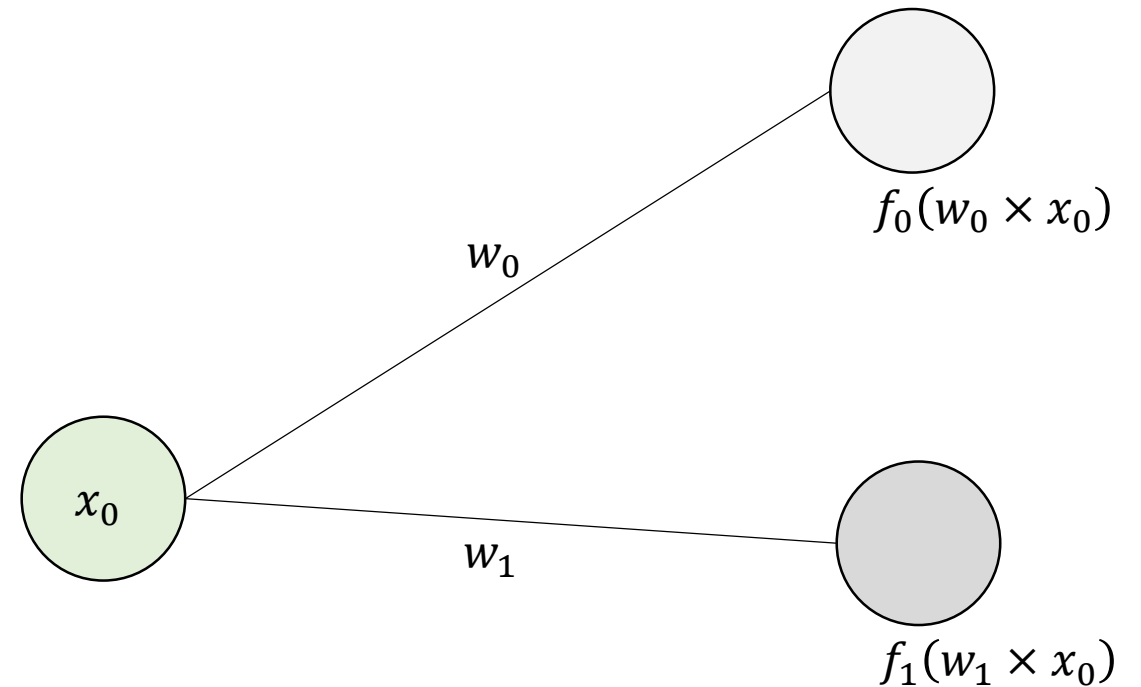
Entrées Neurones Classes



Valeur

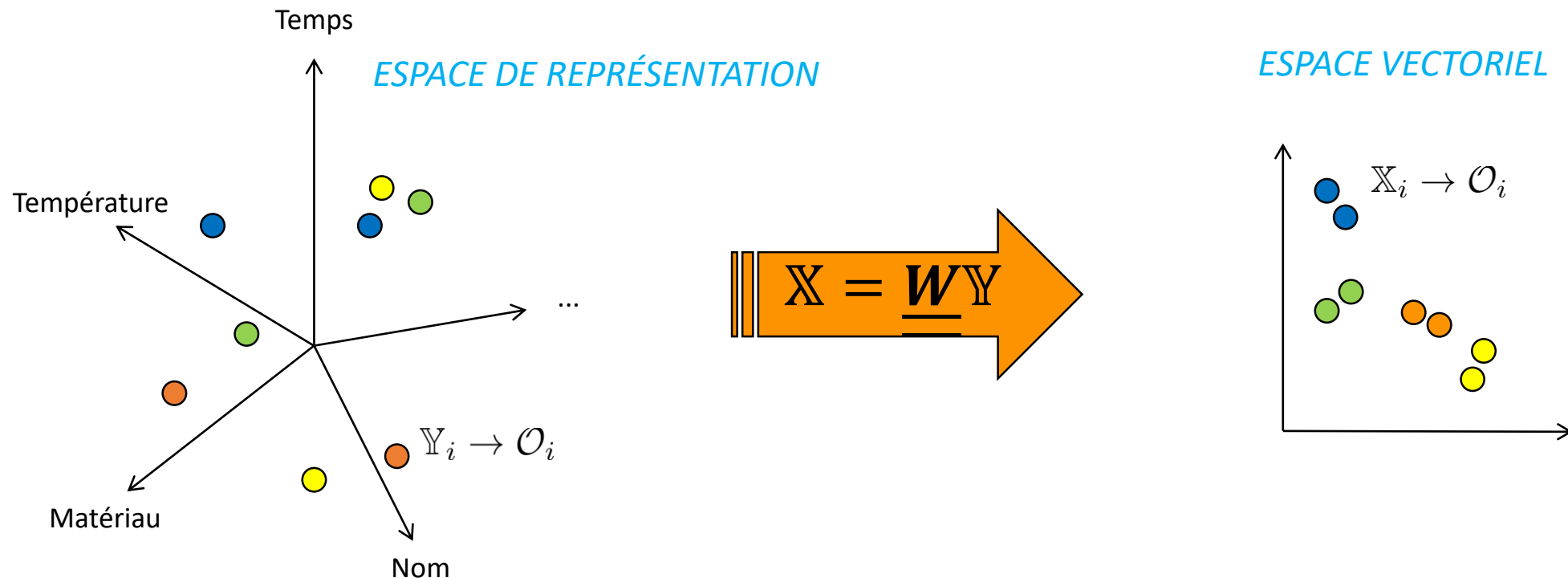
Poids

Fonction d'activation



Code2Vect

- Espace de représentation hétérogène
- Corrélation entre paramètres et résultat



Code2Vect

$$\underline{\hat{X}}_i = \underline{\hat{W}} \underline{Y}_i$$

- L'ordre dans l'espace vectoriel des \underline{X}_i est proportionnel aux données résultat :

$$\|\underline{X}_i - \underline{X}_j\|^2 = |o_i - o_j|^2$$

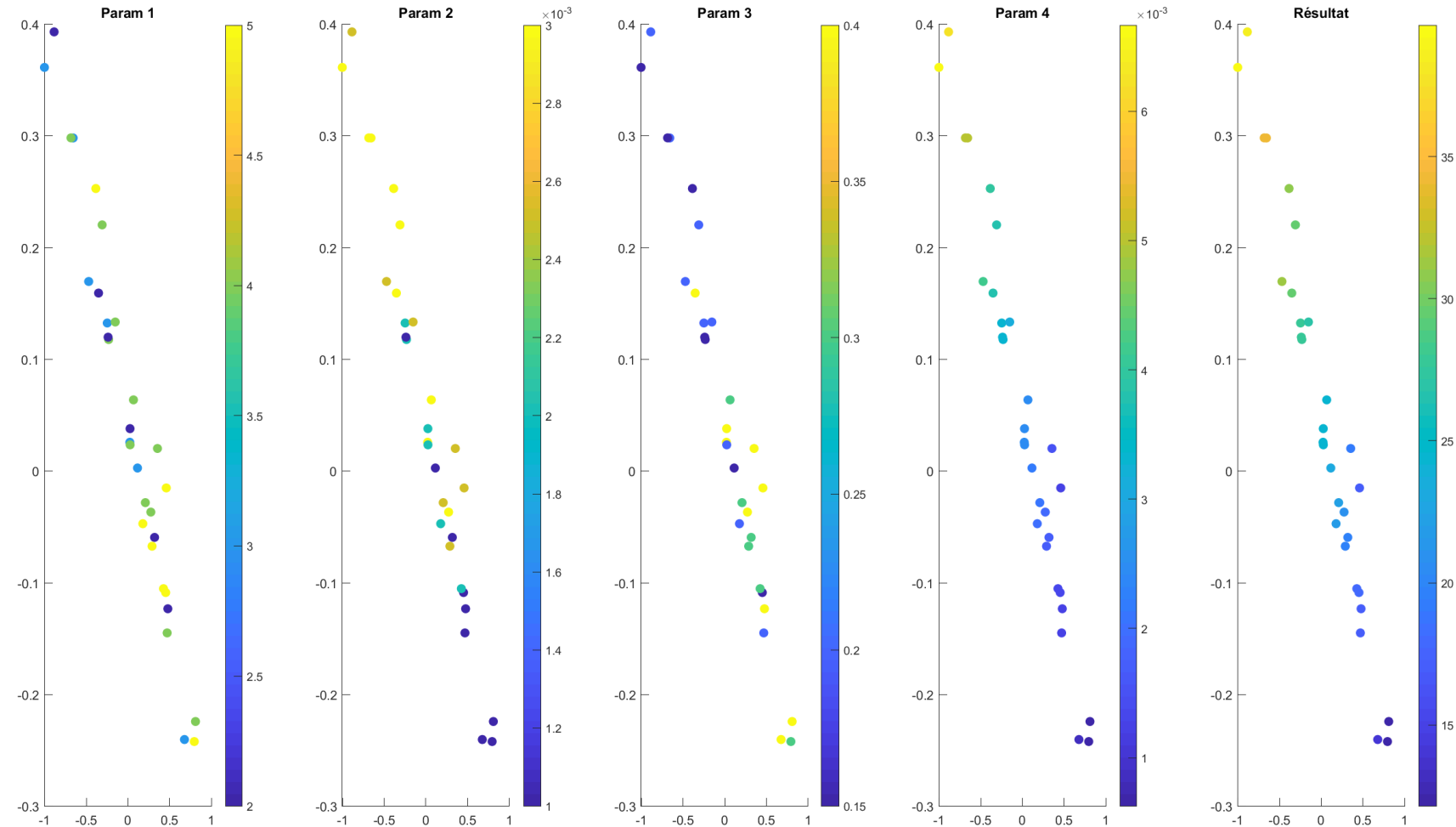
$$(\underline{X}_i - \underline{X}_j)(\underline{X}_i - \underline{X}_j) = (o_i - o_j)^2$$

$$\left(\underline{W}(\underline{Y}_i - \underline{Y}_j)\right)\left(\underline{W}(\underline{Y}_i - \underline{Y}_j)\right) = (o_i - o_j)^2$$

$$(\underline{Y}_i - \underline{Y}_j)\underline{W}^T \underline{W}(\underline{Y}_i - \underline{Y}_j) = (o_i - o_j)^2$$

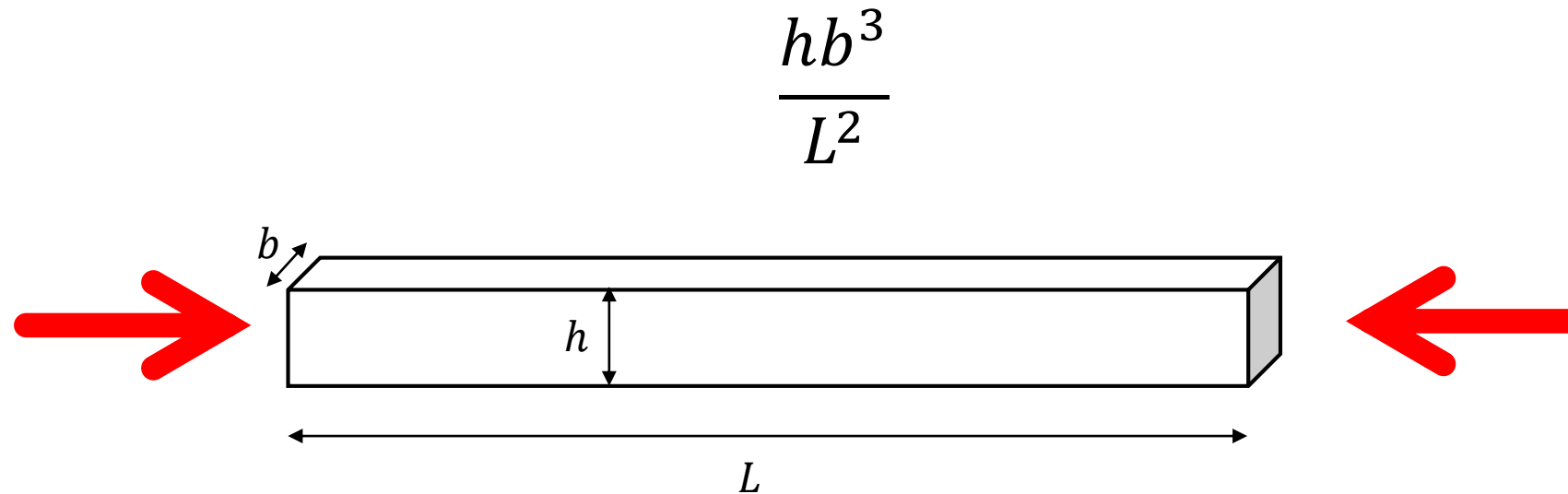
Code2Vect

- PRÉDICTION d'un résultat (classification, régression, etc.)
- CORRÉLATIONS entre paramètres et résultats



Code2Vect

- Création de paramètres « composites »
- Exemple : Équation de flambage



Décomposition en Modes Dynamiques

- Input \underline{I}_i et Output \underline{O}_i
- Modèle $\underline{\underline{M}}$

$$\underline{\underline{M}}_{N \times N} \underline{O}_{N \times 1} = \underline{I}_{N \times 1}$$

- Fonction de transfert $\underline{\underline{T}}$ ($\underline{\underline{T}} = \underline{\underline{M}}^{-1}$)
$$\underline{\underline{T}} \underline{I} = \underline{O}$$

- Décomposition de $\underline{\underline{M}}$ et enrichissement si nécessaire

Tensor Mode Decomposition

- Équation de Fokker-Planck (convection et diffusion)

$$\frac{\partial \Psi(p)}{\partial t} + \frac{\partial}{\partial p} (\dot{p} \Psi) = \frac{\partial}{\partial p} \left(D_r \frac{\partial \Psi}{\partial p} \right)$$

- Applicable a un modèle enrichi par des données ?
 - Problème Markovien
 - Classification de modèles