Apprentissage automatique - partie 2

Mohamed Amine Remita

23 mars 2016

Adapté des cours de Mohamed Bouguessa Ph.D (DIC9370)
et Ahmed Halioui (BIF7101)

1. Rappel
2. Apprentissage automatique
   - Apprentissage non supervisé
3. Évaluation de l'apprentissage
   - Approches pour évaluer la performance
   - Métriques pour mesurer la qualité d'un classificateur
   - Courbe ROC
4. Apprentissage automatique et bioinformatique
5. Atelier
6. Lecture

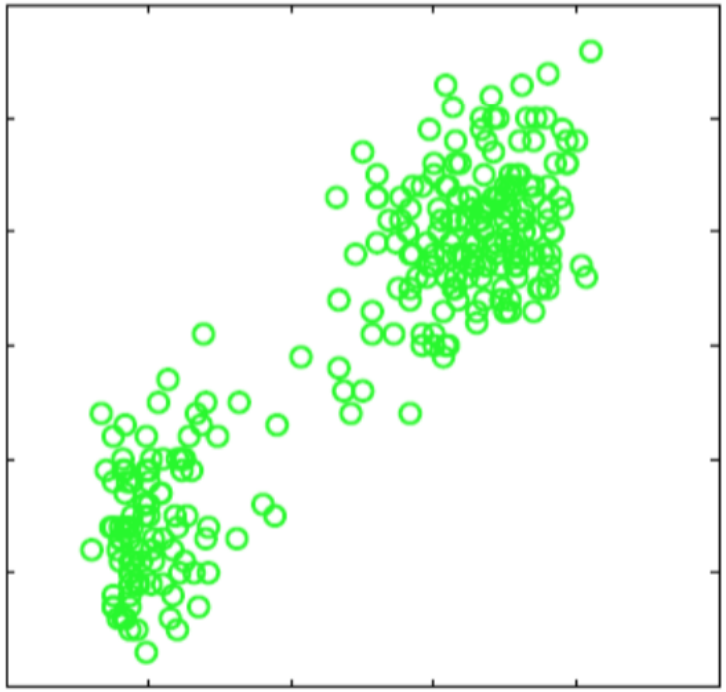
Rappel - Apprentissage automatique

- Apprentissage naturel et apprentissage automatique
- Types des données
- Prétraitement des données
- Apprentissage supervisé

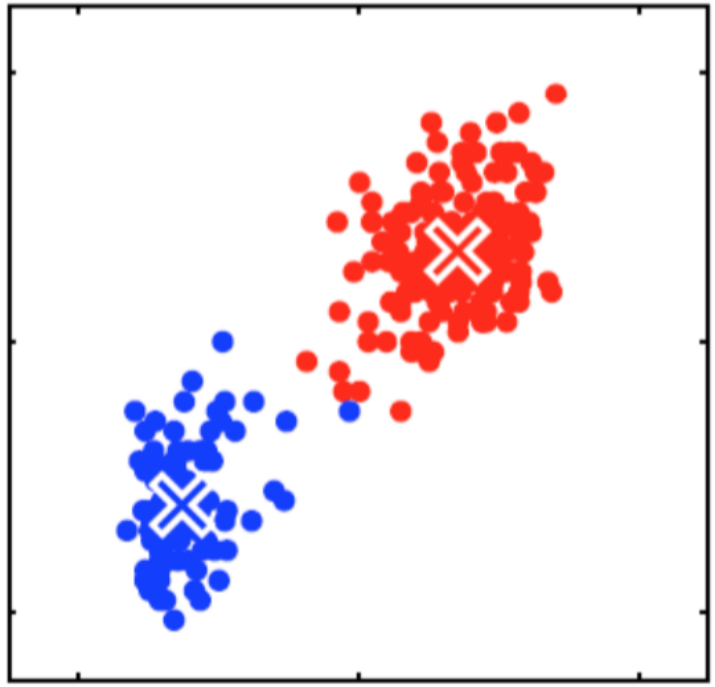
Classification vs clustering

- Le but principal du clustering est la découverte automatique des structures similaires dans l'espace d'objets
- La classification supervisée consiste à l'assignation d'un objet à une classe spécifique parmi un certain nombre de classe prédéfinies

Clustering



Entrée: ensemble de données (non étiquetés)



Sortie : clusters identifiés

Apprentissage non supervisé (clustering)

- Le processus du clustering vise à construire des groupes (clusters) d'objets similaires à partir d'un ensemble hétérogène d'objets
- Chaque cluster issu de ce processus doit vérifier les deux propriétés suivantes :
  1. La cohésion interne (les objets appartenant à ce cluster soient les plus similaires possibles)
  2. L'isolation externe (les objets appartenant aux autres clusters soient les plus distinctes possibles)
- Le processus de clustering repose sur une mesure précise de la similarité des objets que l'on veut regrouper. Cette mesure est appelée distance ou métrique

Stratégies

- Partitionnement (K-means)
- Clustering hiérarchique
- Clustering basé sur la densité (DBSCAN)

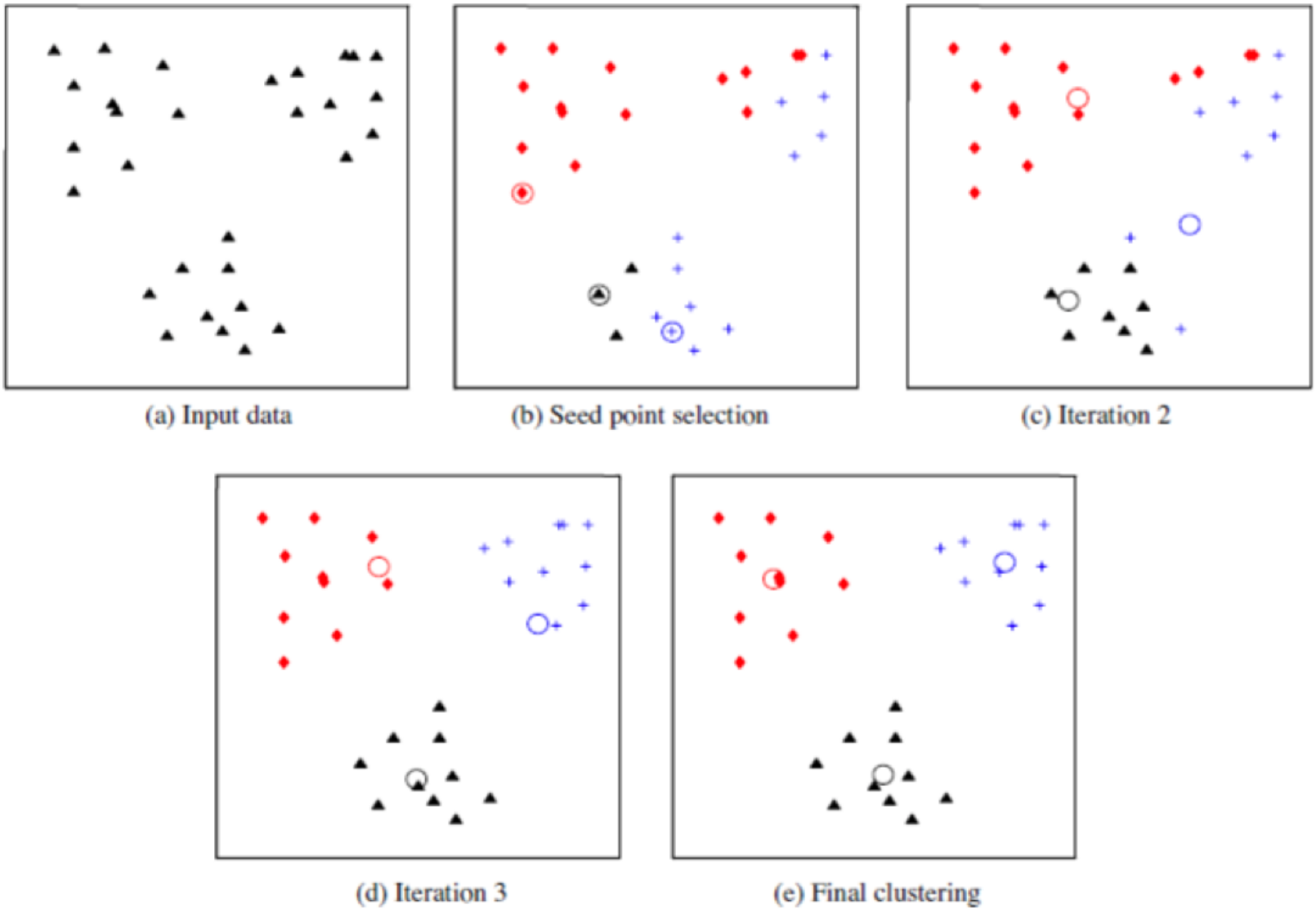
Algorithme K-means

- L'algorithme partitionne l'ensemble des données à un certain nombre de clusters K (K est fourni par l'utilisateur)
- Chaque cluster est représenté par son centre
- On commence avec K clusters et on raffine les clusters itérativement
- K-means génère une partition Hard (chaque objet appartient à un seul cluster seulement)

Algorithme K-means

1. Sélectionner aléatoirement un ensemble de K objets comme centres initiaux
2. Répéter :
   - Former K clusters et ce en assignant chaque point au centre le plus proche
   - Recalculer les centres de clusters
3. Jusqu'à stabilité de la partition (les centres ne changent pas)

Algorithme K-means



(a) Input data (b) Seed point selection (c) Iteration 2

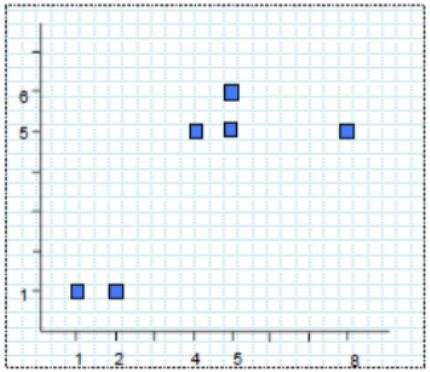
(d) Iteration 3 (e) Final clustering

Caractéristiques de K-means

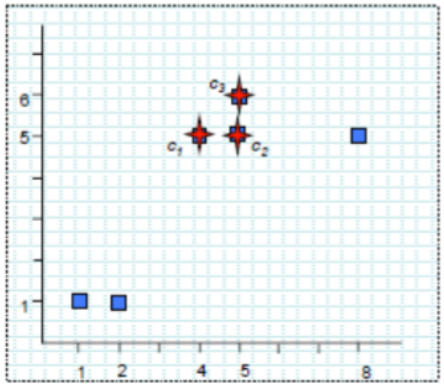
1. Avantages
   - Relativement efficace (rapide)
   - Converge souvent
2. Faiblesses
   - Besoin de spécifier K
   - Ne gère pas le bruit
   - Sensibles à la sélection initiale des centres de clusters

Algorithme K-means

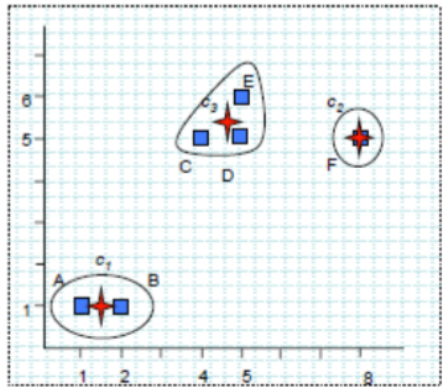
- **Initialisation 1**



Début

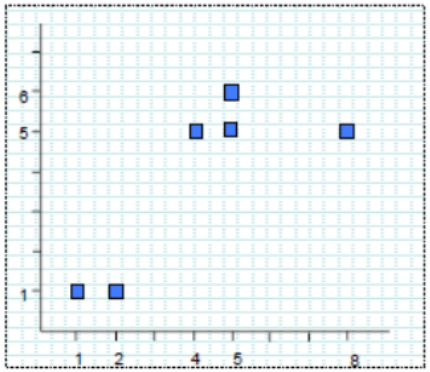


Initialisation

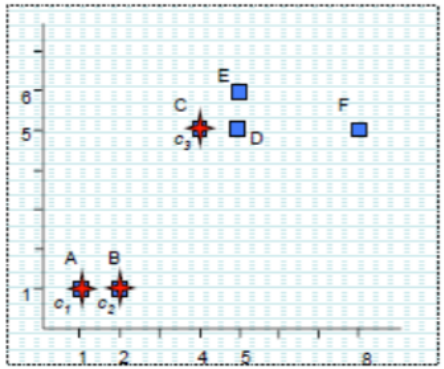


Fin

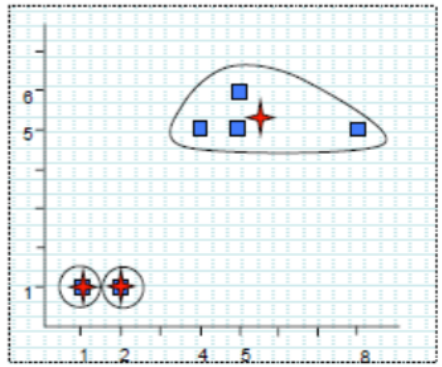
- **Initialisation 2**



Début



Initialisation



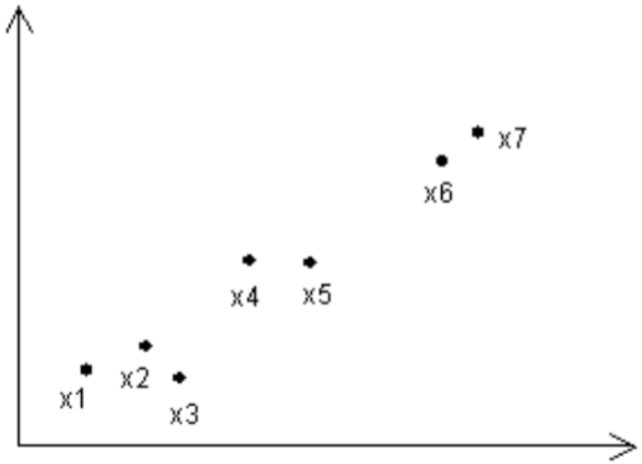
Fin

Clustering hiérarchique

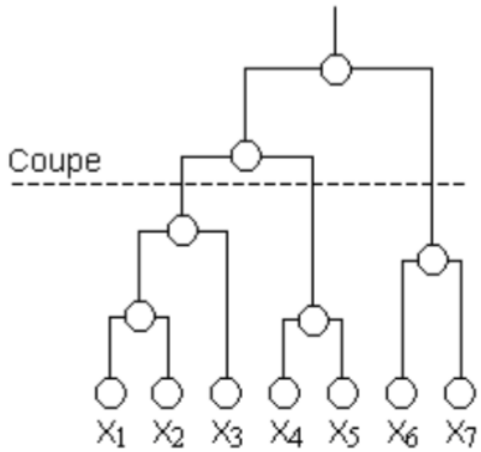
- Un algorithme de clustering hiérarchique ne produit pas une seule partition mais une hiérarchie de partition emboîtées
- Un cluster est défini comme un noeud d'arbre, auquel est associé l'ensemble des objets qui le composent
- Il existe deux catégories d'algorithmes hiérarchiques :
  1. Méthodes ascendantes ou agglomératives
  2. Méthodes descendantes

Méthodes ascendantes ou agglomératives

- La partition initiale contient autant de clusters que d'objets ($K = n$)
- À chaque étape, on cherche un couple (C_i, C_j) de clusters candidats à la fusion qui maximise une certaine mesure de similarité
- On réitère ce processus jusqu'à l'obtention d'un seul cluster contenant tous les éléments
- Afin de déterminer le nombre de clusters, on coupe la hiérarchie à un certain niveau



Un ensemble d'objets à classer



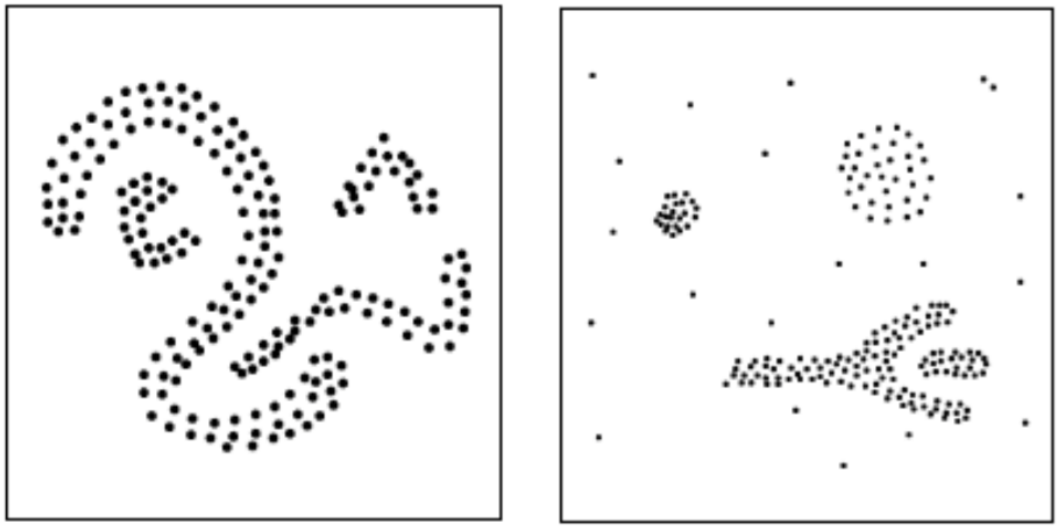
Dendogramme de la partition

Méthodes descendantes

- Commencer avec un cluster contenant touts objets
- Séparer les groupes en plus petits groupes jusqu'à ce que chauqe groupe ne contient qu'un seul objet
- Dans cette approche, on a besoin de décider qu'elle est le cluster qu'on doit le diviser, à quelle étape et comment faire la division

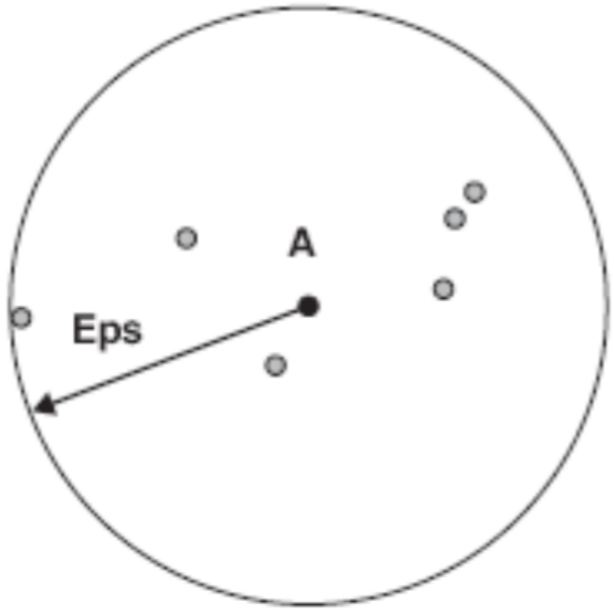
Clustering basé sur la densité

- Les techniques de clustering vu précédemment ne permettent pas l'identification de clusters de forme : étirée, linéaire, allongée, etc.
- DBSCAN (Density-Based Spatial Clustering of Applications with Noise) est capable d'identifier ce type de clusters



Clustering basé sur la densité

- Un cluster est une région de grande densité entourée par des points avec une densité relativement faible
- Un bruit appartient à une région de très faible dendité
- On dit un objet appartient à une région de forte densité si la cardinalité de son voisinage dépasse un certiain seuil



Évaluation de l'apprentissage

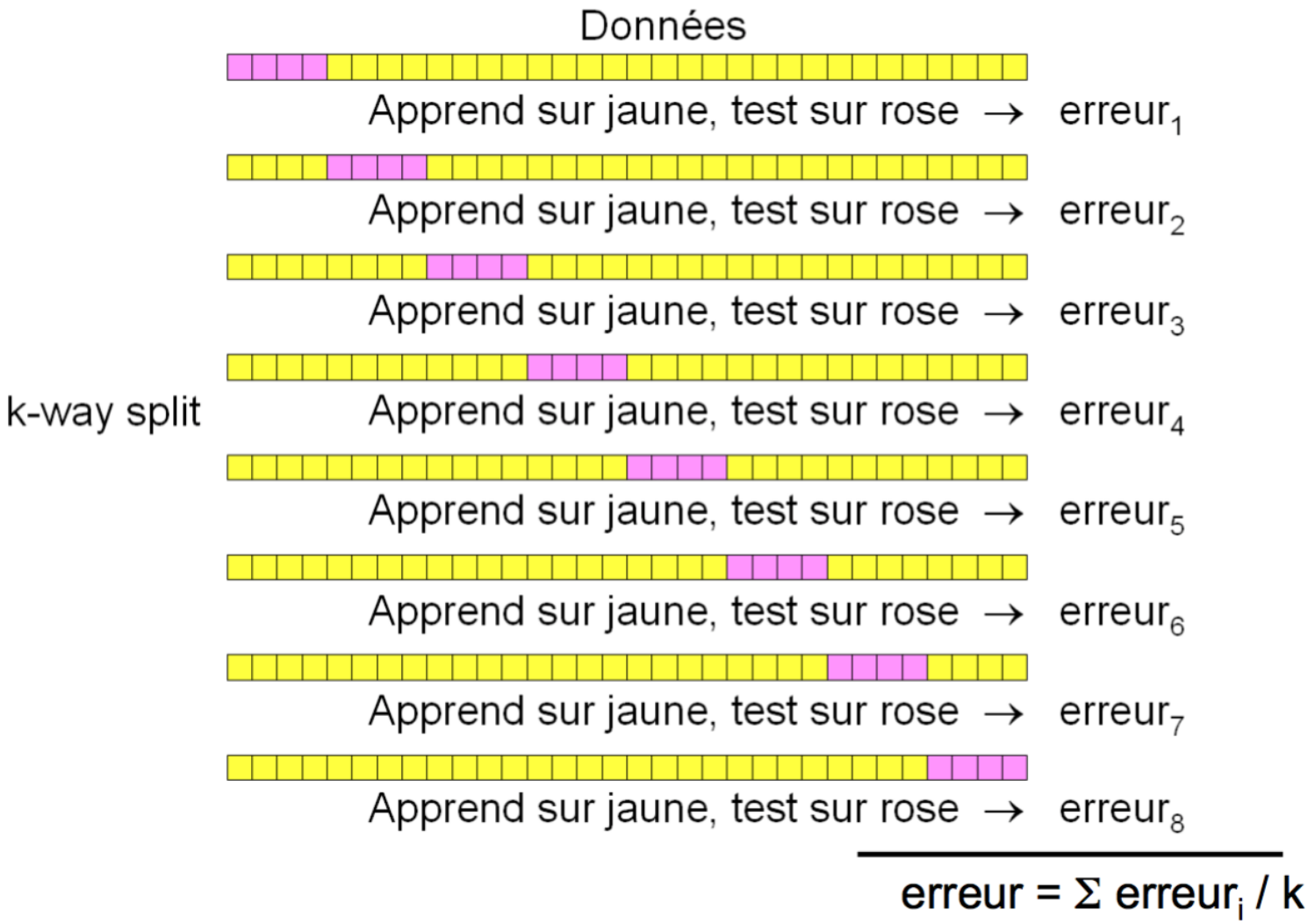
Utilisation d'un échantillon de test

- La méthode la plus simple pour estimer la qualité d'un algorithme d'apprentissage est de diviser l'ensemble des exemples en deux ensembles indépendants : le premier, noté A, est utilisé pour l'apprentissage, le second, noté T, sert à mesurer sa qualité.
- T est l'échantillon de test tel que : $S = A \cup T$ et $A \cap T = \Phi$
- La mesure des erreurs commises par l'algorithme d'apprentissage sur l'ensemble de test T est une estimation de sa qualité

La validation croisée

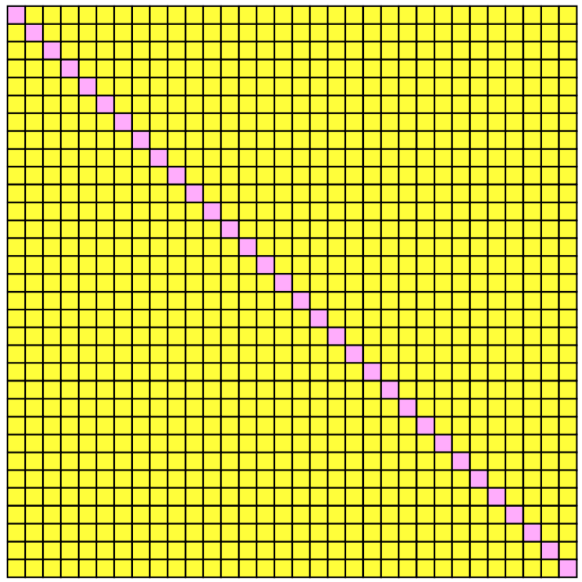
- Diviser les données d'appentissage S en k sous-échantillons de tailles égales
- Retenir l'un de ces échantillons (i). Rouler l'algorithme C sur l'ensemble $S - i$
- Mesurer le taux d'erreur $R_i(C)$ sur l'ensemble de test i
- Recommencer le processus décrit ci-dessus pour chaque échantillon i
- L'erreur estimée finale est donnée par la moyenne des erreurs mesurées

La validation croisée



Leave-one-out

- Lorsque les données disponibles sont très peu nombreuses, il est possible de pousser à l'extrême la méthode de validation croisée en prenant k le nombre total d'exemple diponible ($k = n$). Dans ce cas, on ne retient à chaque fois qu'un seul exemple pour le test, et on répète l'apprentissage k fois pour tous les autres exemples d'apprentissage.



Matrice de confusion

- Mesurer la qualité de généralisation du classificateur

		classe prédite : Oui	classe prédite : Non
classe réelle	Oui	Vrais positifs (TP)	Faux négatifs (FN)
	Non	Faux positifs (FP)	Vrais négatifs (TN)

Calcul des mesures

- Précision $= \frac{TP}{TP+FP}$ (par rapport aux instances prédites)
- Rappel $= \frac{TP}{TP+FN}$ (par rapport aux instances réelles)
- F-mesure $= \frac{2(Precision \times Rappel)}{Precision+Rappel)}$

		classe prédite: Oui	classe prédite: Non
classe réelle	Oui	Vrais positifs (TP)	Faux négatifs (FN)
	Non	Faux positifs (FP)	Vrais négatifs (TN)

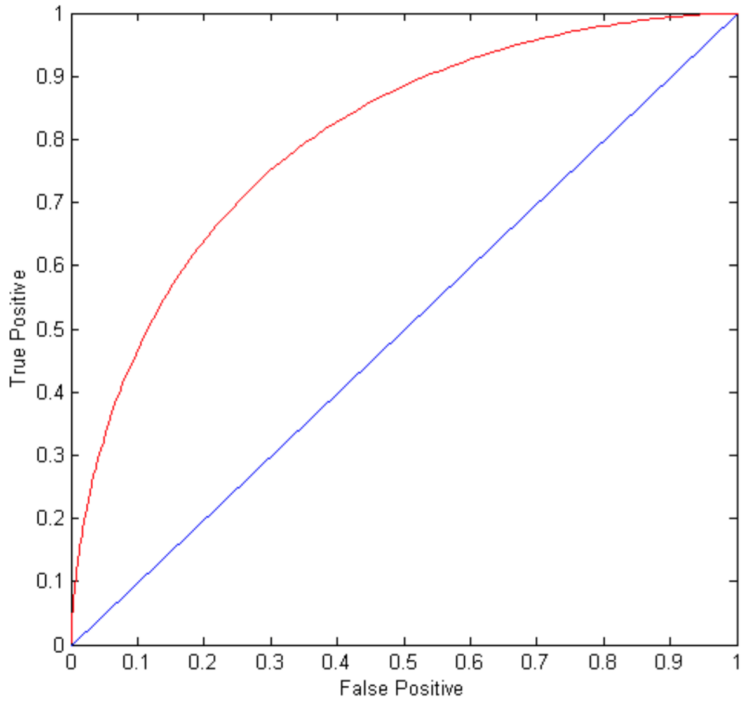
Calcul des mesures

- True Positive Rate
  TPR (ou sensibilité) $= \frac{TP}{TP+FN}$
- True Negative Rate
  TNR (ou Spécificité) $= \frac{TN}{FP+TN}$
- False Positive Rate
  FPR (ou 1-Spécificité) $= \frac{FP}{FP+TN}$
- False Negative Rate FNR $= \frac{FN}{TP+FN}$

		classe prédite	
		Oui	Non
classe réelle	Oui	Vrais positifs (TP)	Faux négatifs (FN)
	Non	Faux positifs (FP)	Vrais négatifs (TN)

Courbe ROC

- ROC : Receiver Operating Characteristic
- Elle met en ralation dans un graphique les taux de faux positifs (en abscisse) et les taux de vraix postifs (en ordonnée)
- Elle est définie pour les problémes de deux classes

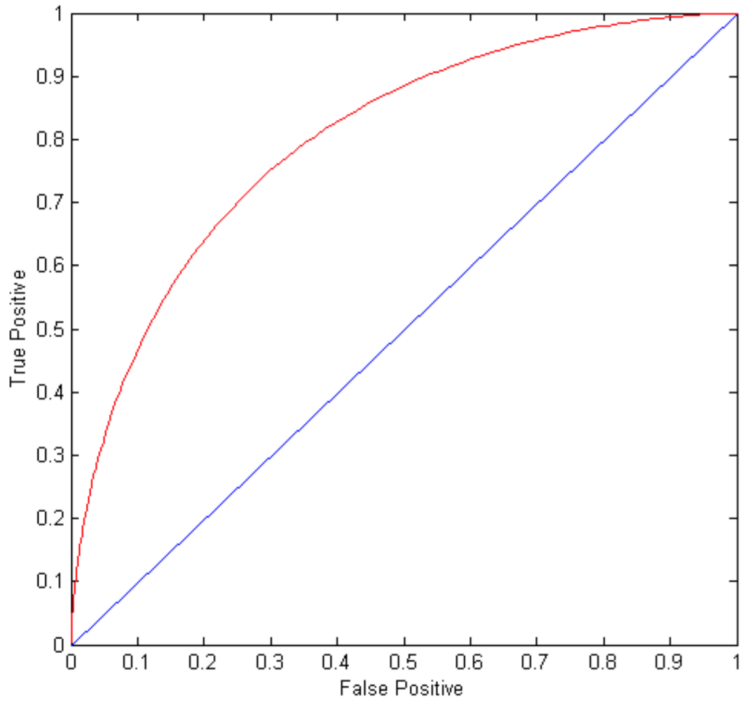


Courbe ROC

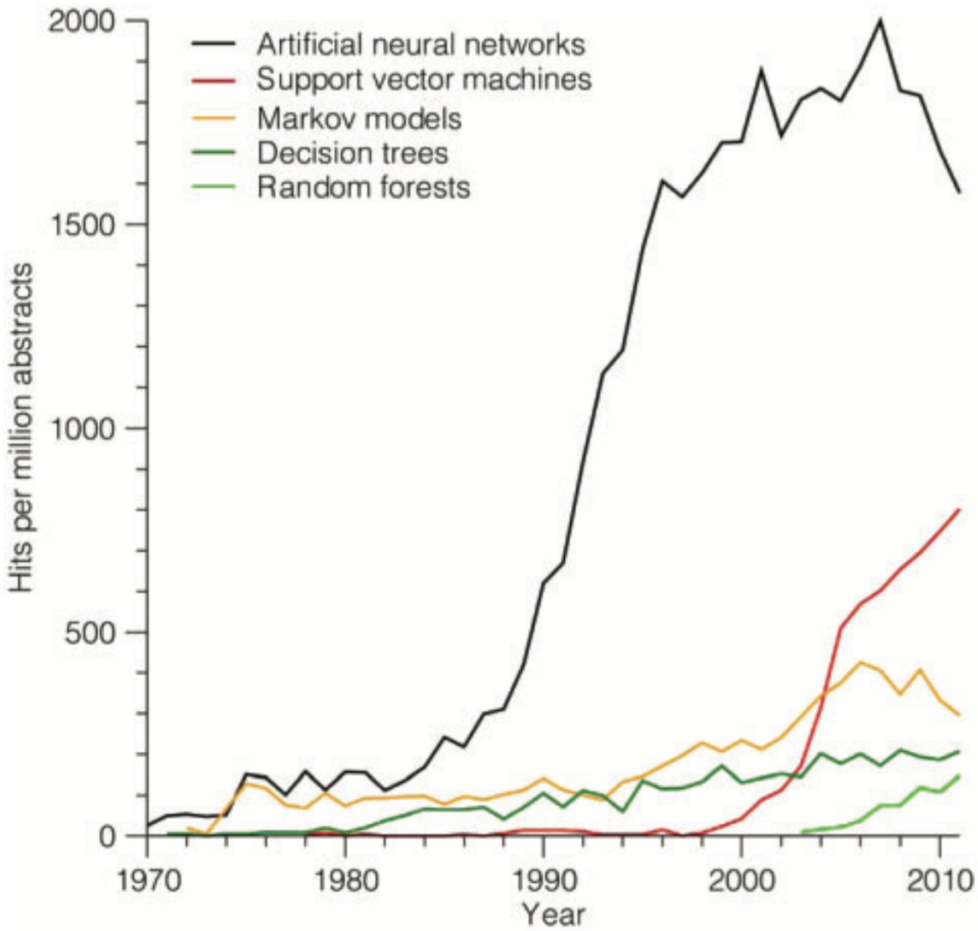
- Les classificateurs discrets (arbres de décision) renvoient seulement une classe de décision et donc une seule matrice de confusion $--> $ un seul point de l'espace ROC
- La courbe ROC est désignée pour les classificateurs à score (Réseaux de neurones, SVM, Réseaux bayésiens) qui renvoient avec une classe de décision un score de probabilité qui représente le degré d'appartenance d'un example à une classe spécifique

Courbe ROC

- Quelques points importants dans la courbe :
- (FPR, TPR) :
- (0, 0) prédit toujours négatif
- (1, 1) prédit toujours positif
- (0, 1) classification idéale
- Ligne diagonale (ligne de hasard) : classification aléatoire

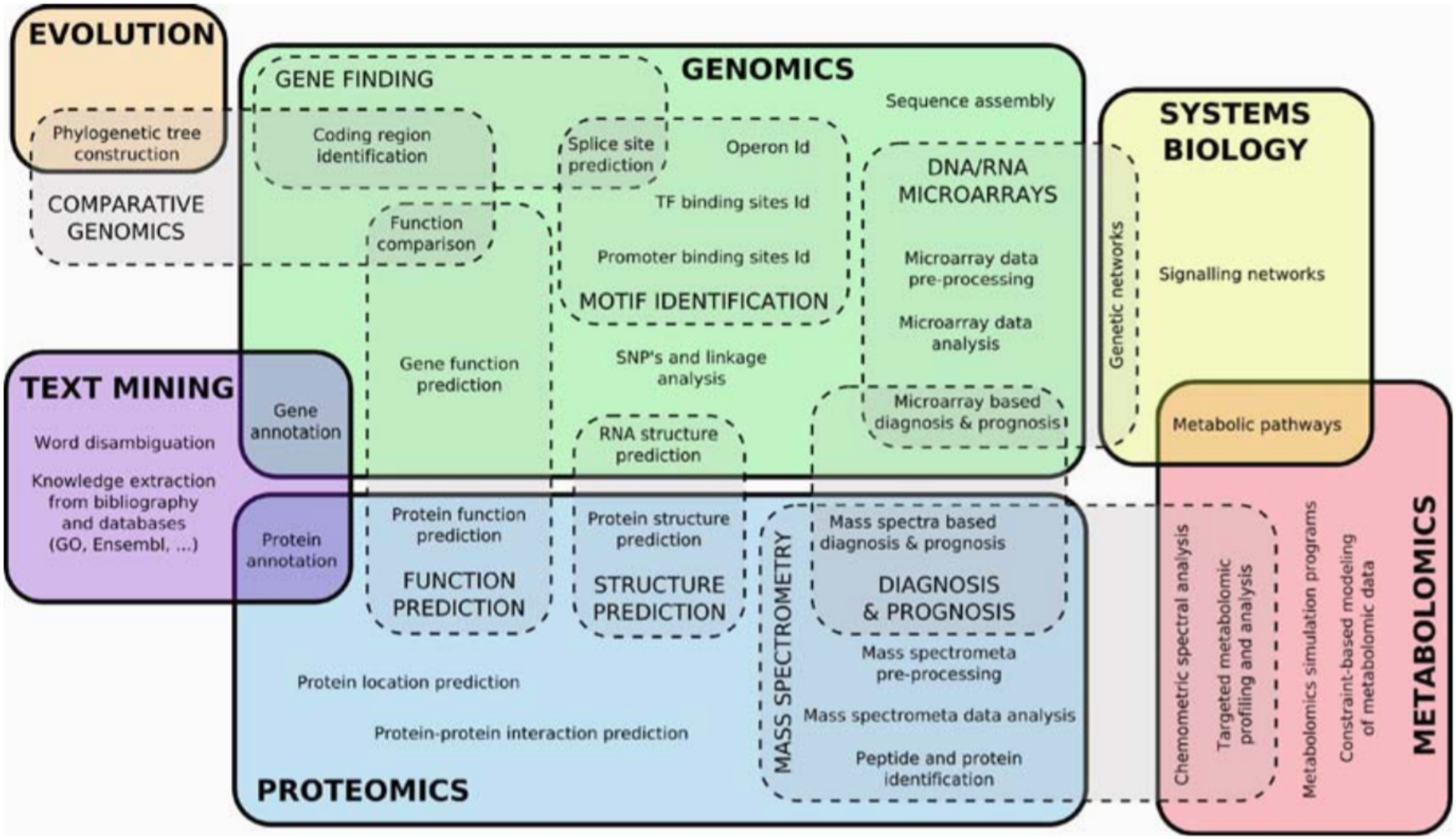


Utilisation de l'apprentissage supervisé



The growth of supervised machine learning methods in PubMed.
Jensen et Bateman 2011

Applications en bioinformatique



General scheme of the current applications of machine learning techniques in bioinformatics. Inza et al. 2010

Atelier : Prédiction des précurseurs des miARNs (Triplet-SVM)

- Téléchargez de miRBase le fichier Fasta des précurseurs
- Avec le programme **0_select_espece_fasta.pl**, sélectionnez les séquences de l'homme (hsa)
- Comment Triplet-SVM construit-il ses jeux de données négatifs ?
- Utilisez les deux métodes offertes par **1_generer_ensemble_negatif.pl** pour créer deux jeux de données négatifs
- Décrivez chaque méthode
- Construisez un troisième jeux de données négatif contenant des séquences nucléotidiques vraies (cDNA, EST etc)

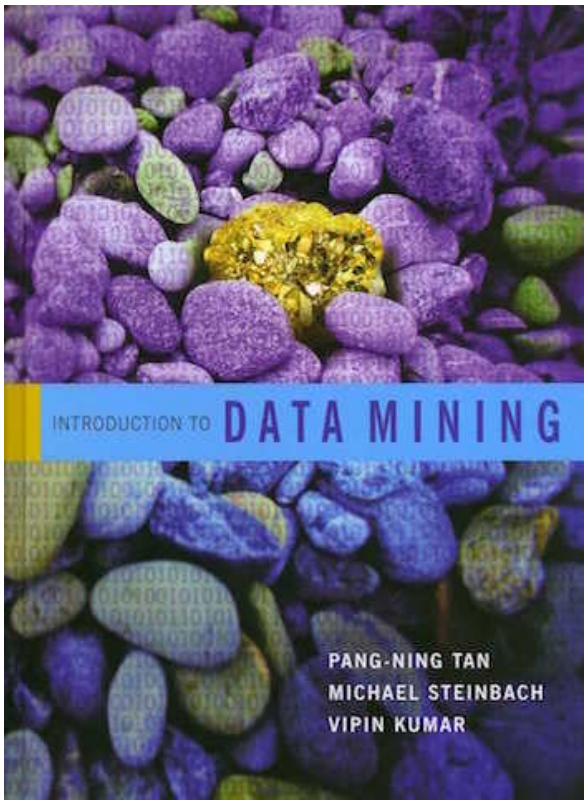
Atelier : Prédiction des précurseurs des miARNs (Triplet-SVM)

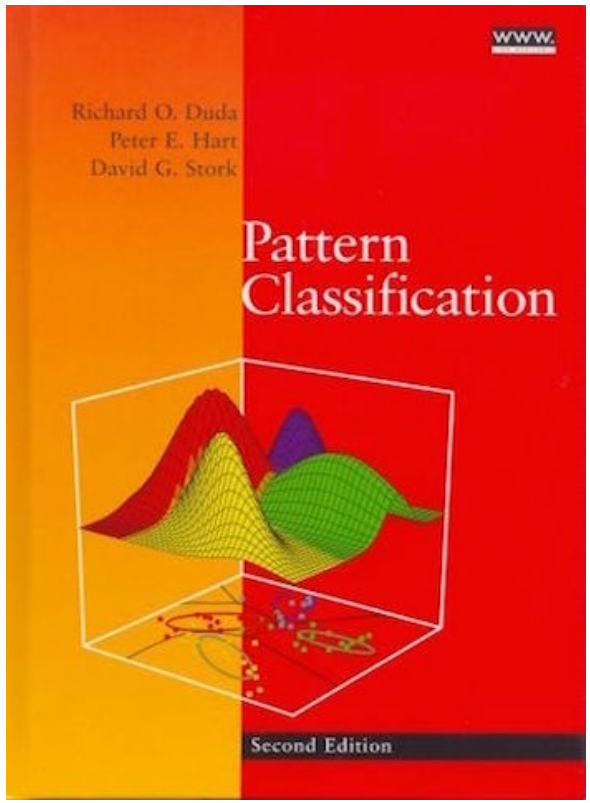
- Partitionnez les données (vrais et synthétiques) en deux sous-ensembles (entrainement et test) avec **2_generer_train_test_set_fasta.pl**
- Repliez toutes ces séquenes
- Calculez le nombres d'occurrences des triplets de ces séquences avec **3_calculer_Xu_triplets.pl**
- Créez des jeux de données finaux en fusionnant données positives et négatives
- Utilisez WEKA pour l'apprentissage de differents algorithmes en utilisant une validation croisée ou les jeux de données de test
- Comparez vos résultats avec les résultats de Triplet-SVM

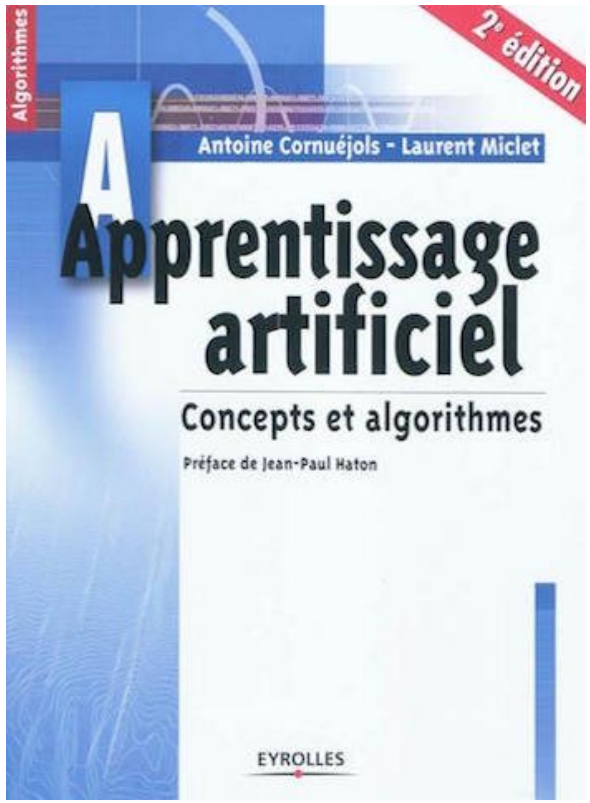
Atelier : Comparaison des précurseurs des animaux et des plantes

- À partir du fichier des précurseurs, générez un fichier contenant des précurseurs de quelques animaux (homme, souris, etc) et un autre des plantes (Arabidopsis, riz, mais etc)
- Créez ensuite un jeux de données pour l'entrainement et un autre pour le test à partir de chaque fichier
- Repliez les séquences
- Calculez le nombre d'occurences des triplets
- Fusionnez les résultats animaux/plantes, jeux d'entrainement et de test séparément
- Utilisez WEKA pour entrainer des algorithmes supervisés et non supervisés
- Est ce que les **Triplets** peuvent discriminer entre les précurseurs des animaux et des plantes ?

Lecture


INTRODUCTION TO DATA MINING
PANG-NING TAN
MICHAEL STEINBACH
VIPIN KUMAR


www.
Richard O. Duda
Peter E. Hart
David G. Stork
Pattern Classification
Second Edition


Algorithmes
2e édition
Antoine Cornuéjols - Laurent Miclet
Apprentissage artificiel
Concepts et algorithmes
Préface de Jean-Paul Haton
EYROLLES