

Introduction aux Méthodes d'Intelligence Artificielle pour la Santé

Pr Roch Giorgi

 roch.giorgi@univ-amu.fr

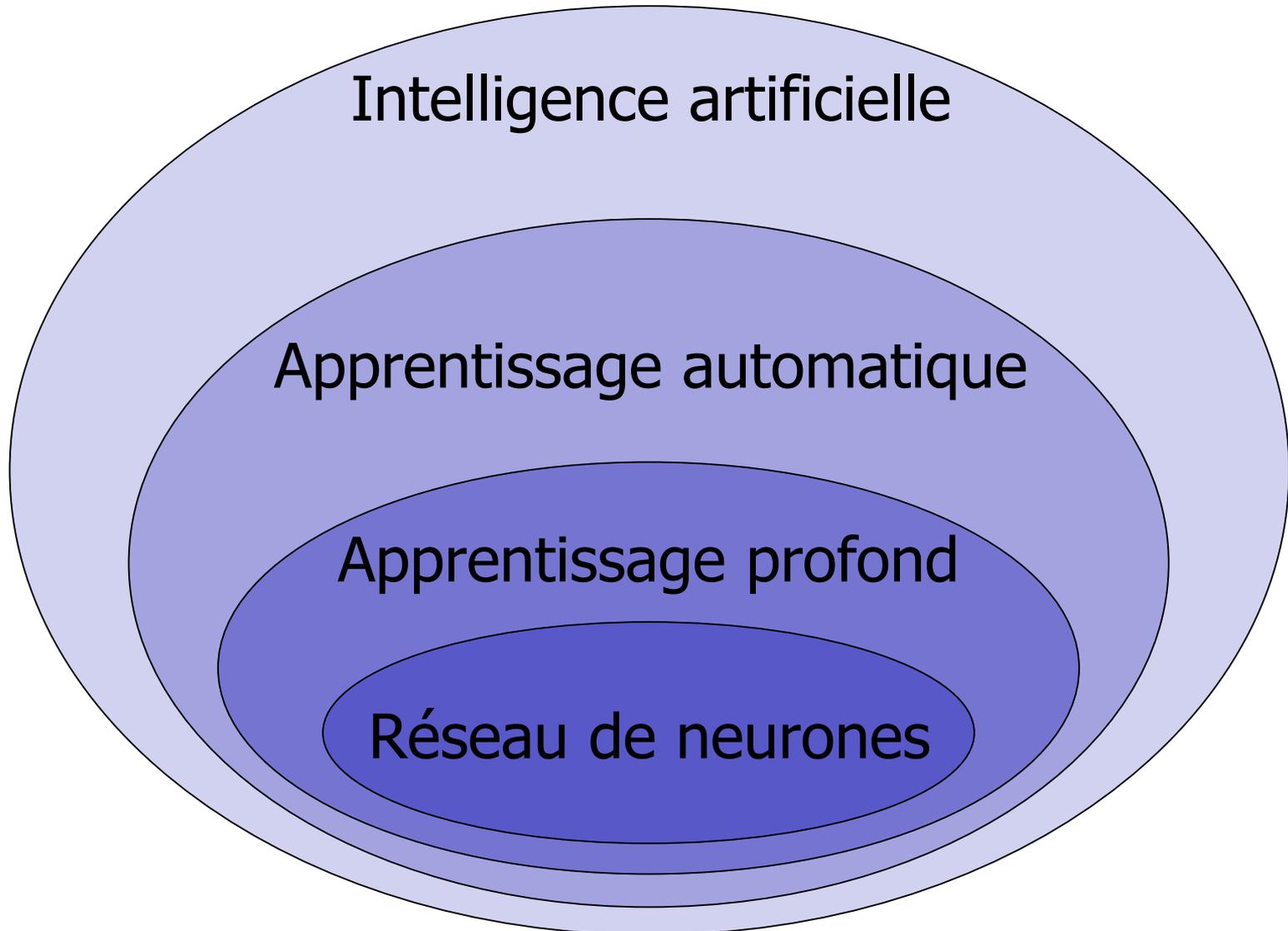
Intelligence Artificielle

- Naissance dans les années 1950
 - ✓ Réalisation de tâches humaines par des machines mimant l'activité du cerveau
- Développement dans les années 1980
 - ✓ Développement des techniques informatiques, augmentation de la puissance de calcul
- Nouvel essor depuis 2010
 - ✓ Massification des données
 - ✓ Nouvelle puissance de calcul

Approches de l'IA

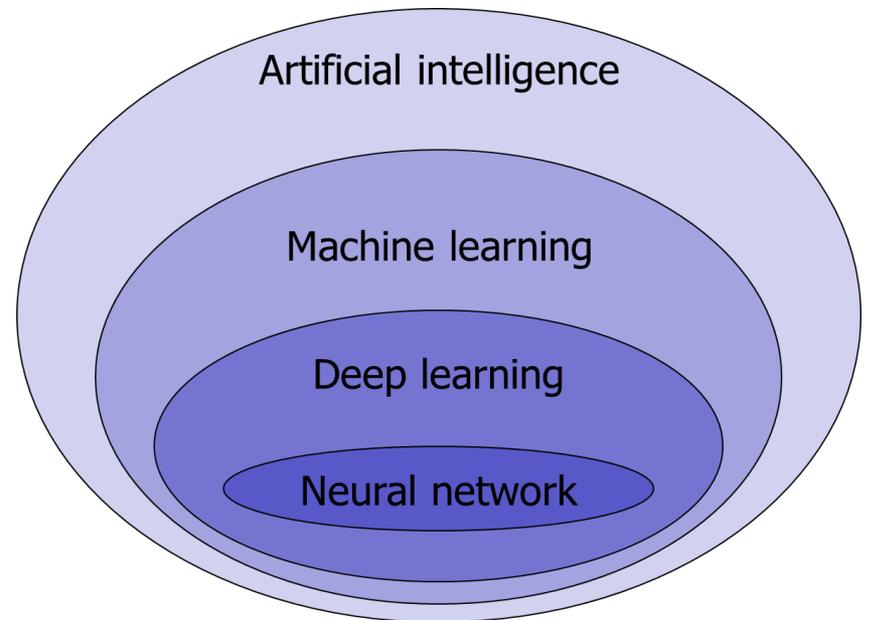
- Approche symbolique
 - ✓ Paradigme dominant jusqu'au début des années 1990
 - ✓ Recherche à expliquer les connaissances
 - ✓ Raisonnement en appliquant des règles logiques
- Approche numérique
 - ✓ Développée grâce à l'augmentation de la puissance de calcul, la massification des données
 - ✓ Recherche à extraire les connaissances implicitement, sans modèle préétabli
 - ✓ Raisonne sur les données, expérience accumulée

Positionnement des Notions d'IA



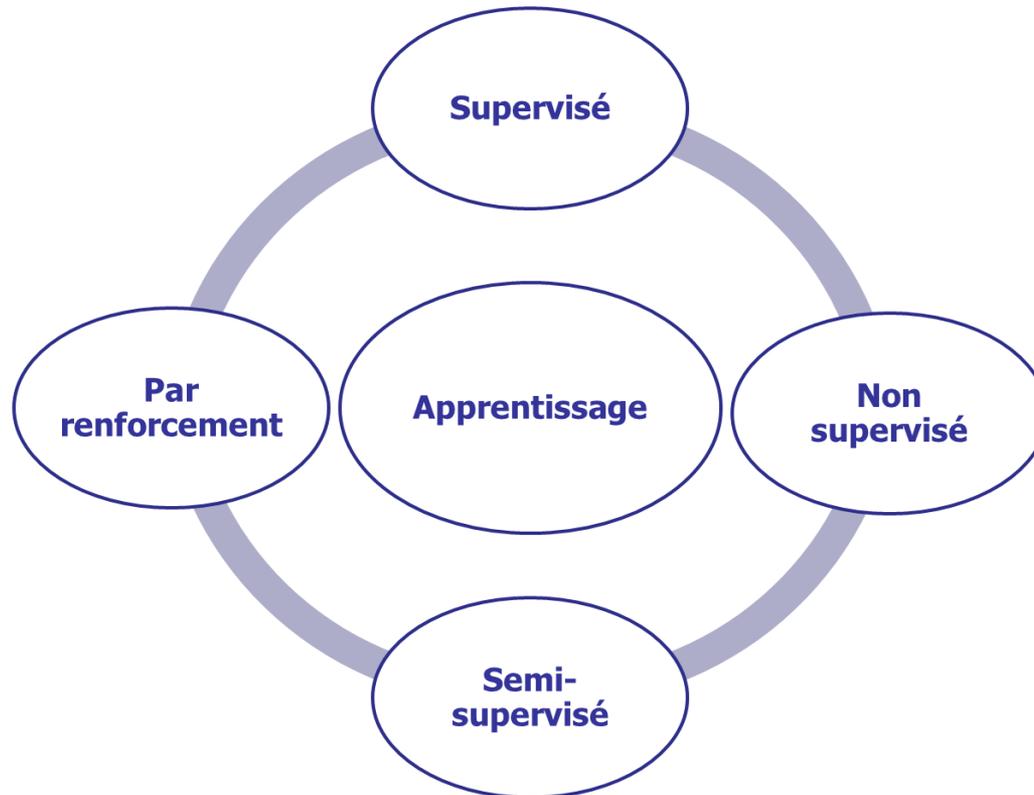
Artificial Intelligence (AI) Vocabulary

- Intelligence artificielle
 - ✓ Artificial intelligence
- Apprentissage automatique
 - ✓ Machine learning
- Apprentissage profond
 - ✓ Deep learning
- Réseau de neurones
 - ✓ Neural network



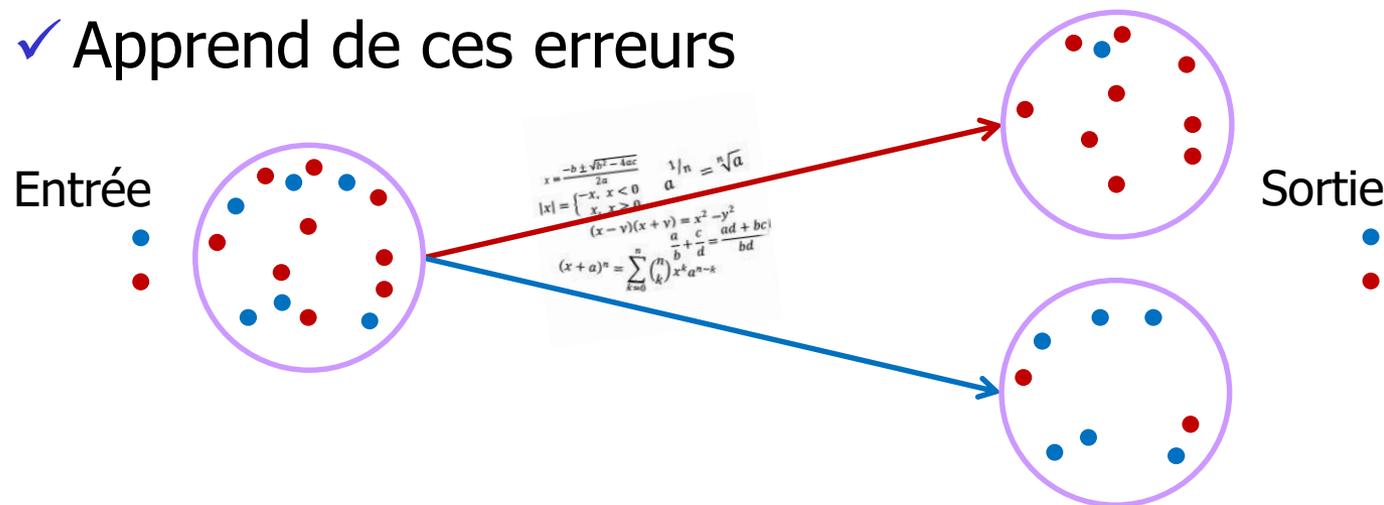
Machine Learning

- Améliorer les performances à résoudre des tâches sans être explicitement programmé pour chacune, par essais et erreurs, avant de se débrouiller tout seul



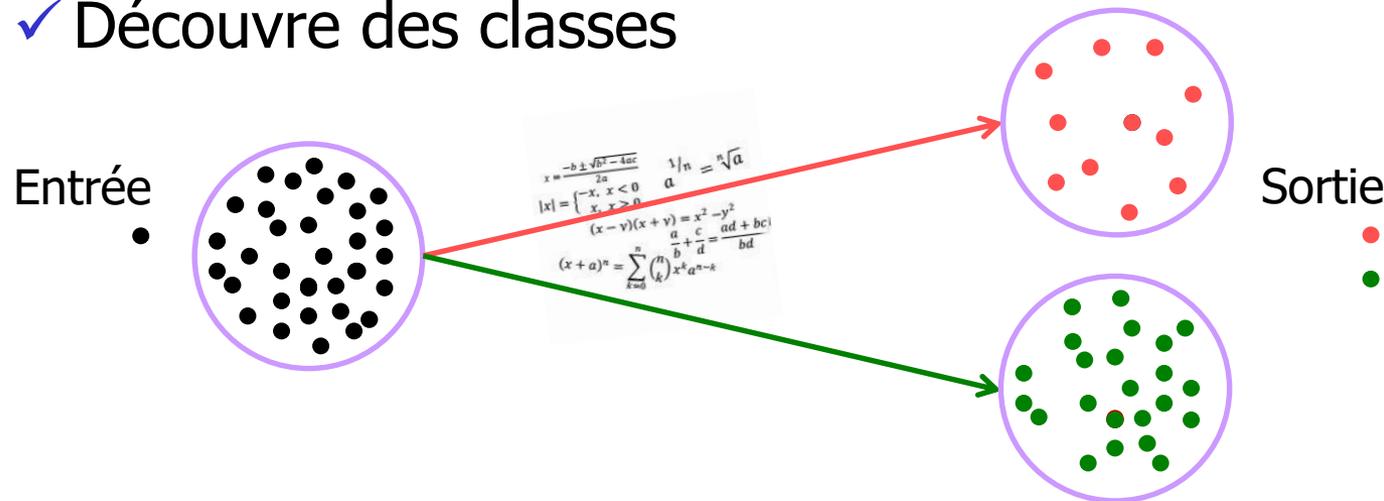
Apprentissage Supervisé

- Connaissance des classes, de leurs étiquettes
 - ✓ Vivant / Décédé
 - ✓ Maladie absente / Maladie présente
 - ✓ Grade I, II ou III du cancer du sein...
- Modèle entraîné sur ces données
 - ✓ Apprend à classer ces données
 - ✓ Apprend de ces erreurs



Apprentissage Non Supervisé

- Pas de connaissance *a priori* des classes, de leurs étiquettes
 - ✓ Données sociodémographiques, relatives à l'état de santé, à l'activité, environnement,...
- Modèle entraîné sur ces données
 - ✓ Découvre la structure des données
 - ✓ Découvre des classes



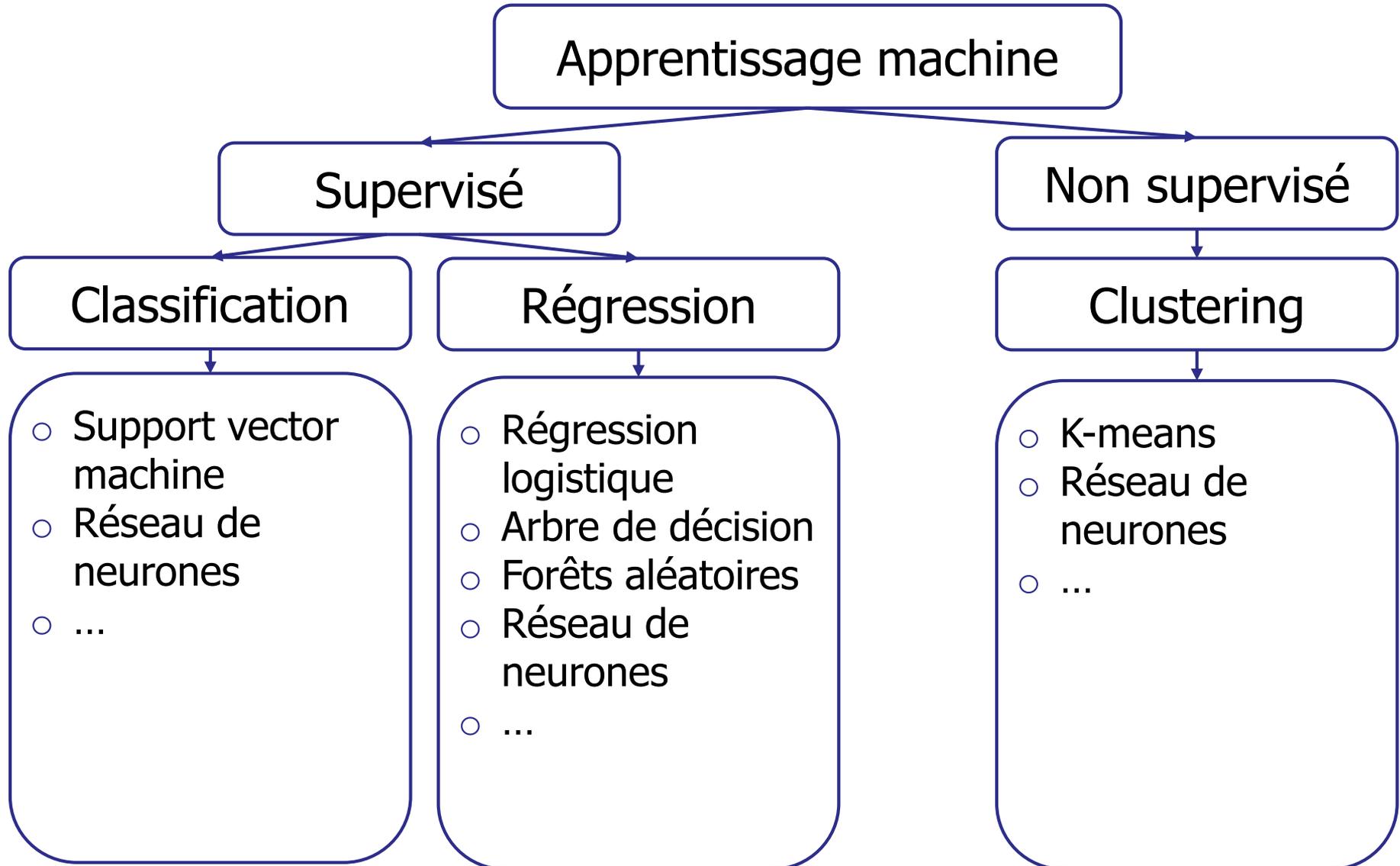
Apprentissage Semi-Supervisé

- Mêmes applications que l'apprentissage supervisé
- Modèle entraîné sur des données étiquetées et sur des données non étiquetées

Apprentissage par Renforcement

- L'algorithme (agent) découvre par tâtonnement (interactions avec l'environnement) les actions pour trouver la solution optimale
 - ✓ Erreur → pénalisation
 - ✓ Bonne décision → récompense
 - Exemple : création de prothèses de jambe de nouvelle génération capables de reconnaître la façon de marcher des utilisateurs et de s'adapter pour rendre le mouvement plus facile et plus efficace (*Stanford Neuromuscular Biomechanics Laboratory*)

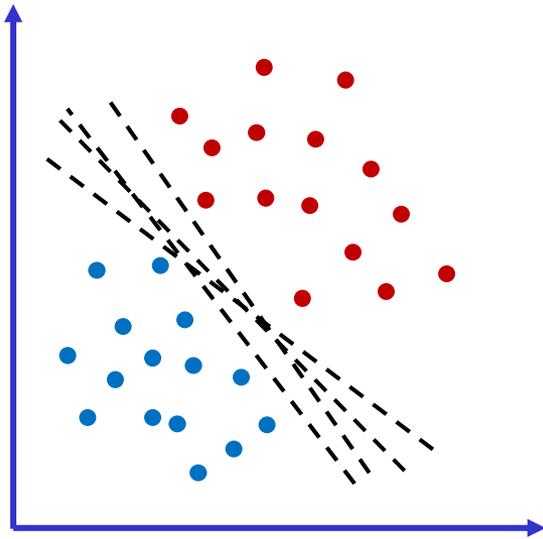
Des Méthodes



- SVM (Support Vector Machine)
- Classifieurs permettant de
 - ✓ Traiter des problèmes de discrimination non linéaire
 - ✓ Se placer dans un cadre d'optimisation (*sens math.*)

Ex : Rythme Sinusal (RS) / Fibrillation Atriale (FA)

- ✓ Intervalle RR sépare 2 contractions ventriculaires
- ✓ Valeurs différentes selon RS / FA (idem pour certaines variables dérivées de la variable RR)
- Trouver comment séparer au mieux les patients en **RS** et en **FA**, en minimisant les erreurs de classification

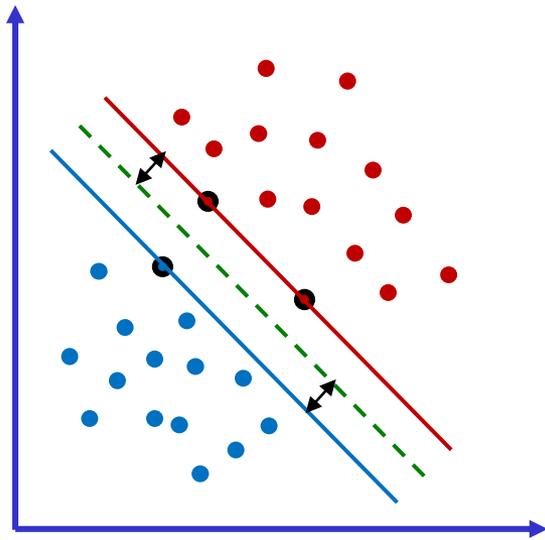


--- Hyperplans séparateurs

- ✓ Pour un ensemble de points linéairement séparables, il existe une infinité d'hyperplans séparateurs

Machines à Vecteurs de Support

(3)

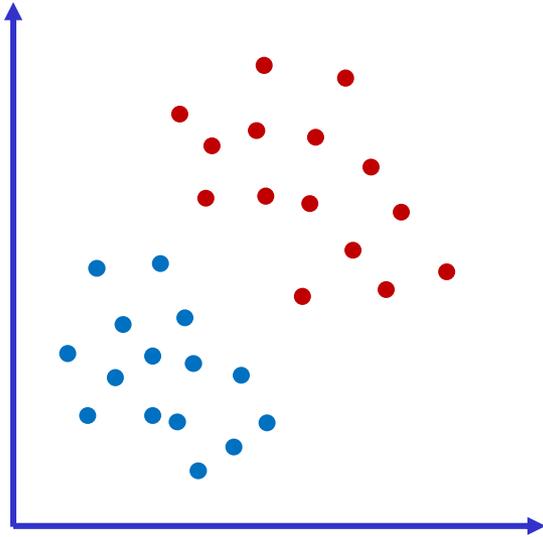


- Hyperplan optimal
- Plans supports
- Vecteurs supports

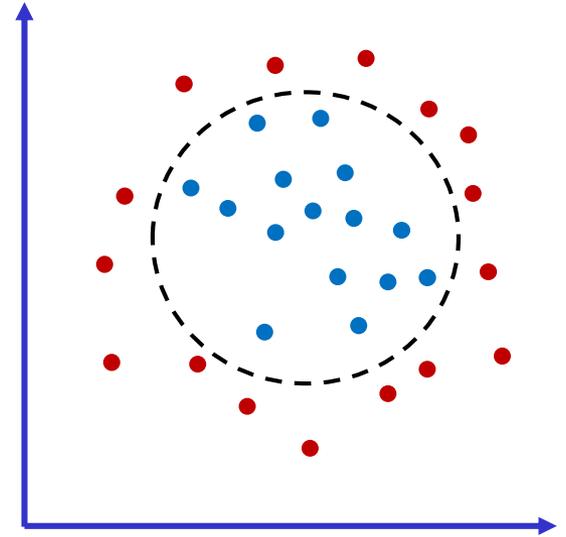
- ✓ Il existe un unique hyperplan optimal, qui maximise la marge entre les échantillons et l'hyperplan séparateur

Machines à Vecteurs de Support

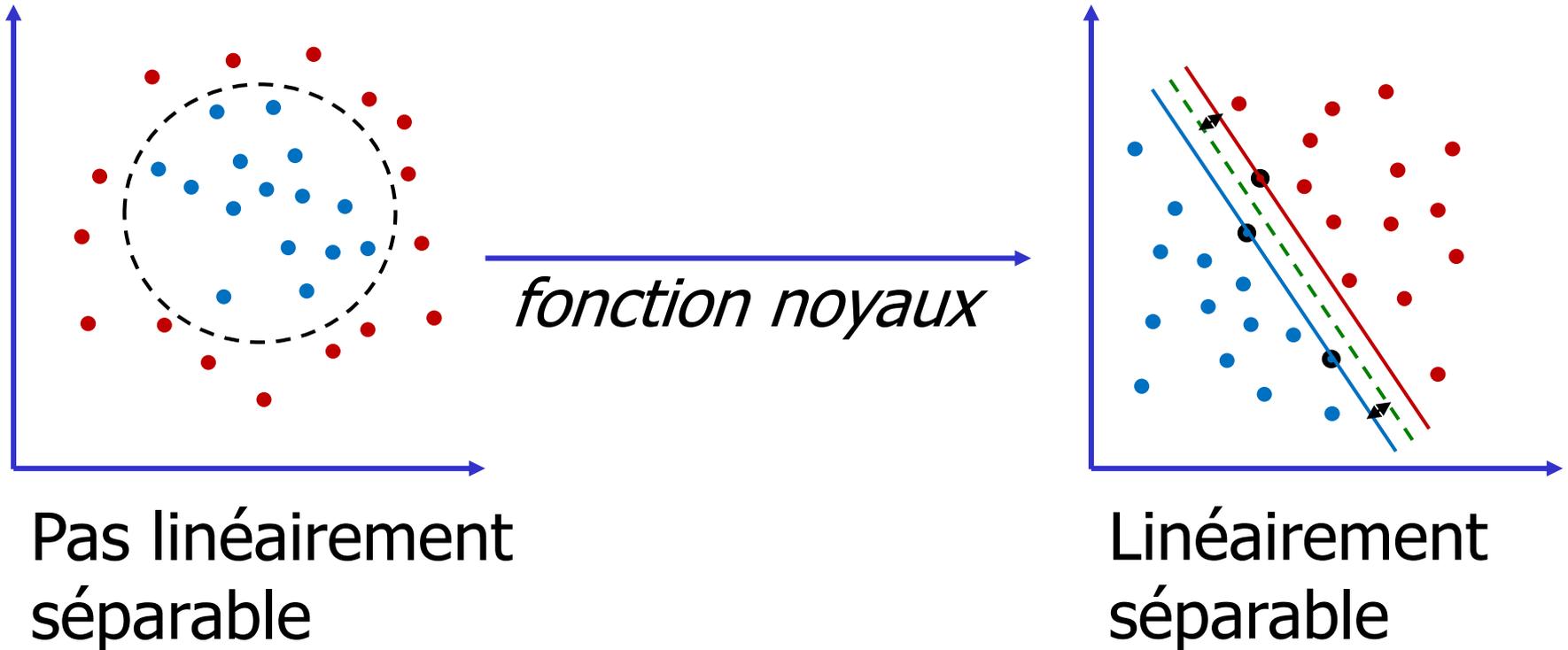
(4)



Linéairement
séparable



Pas linéairement
séparable

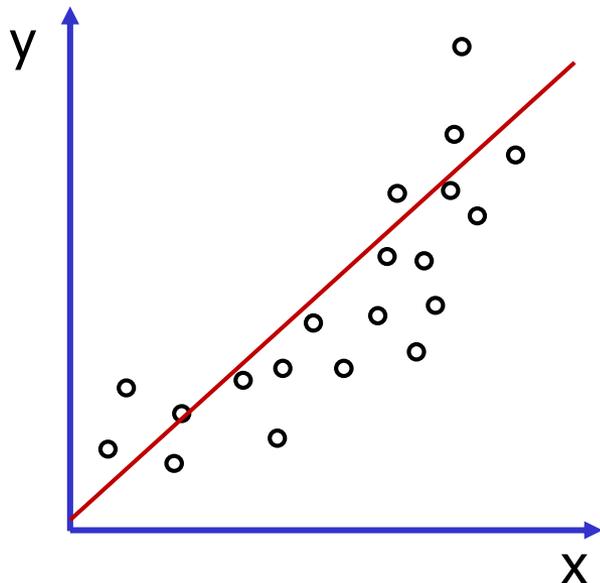


- ✓ Il est possible de résoudre un problème non linéairement séparable en linéairement séparable en effectuant une transformation dans un nouvel espace de redescription

Modèle de Régression

- Régression linéaire simple

- ✓ Etudie la relation f entre une variable Y (continue) et une variable X (continue, pour l'exemple)



- ✓ Hypothèse que f est de la forme $f(X) = a + \beta X$
- ✓ A partir des observations de X et de Y , il faut estimer les coefficients a et β tels que $Y = \hat{a} + \hat{\beta}X$
- ✓ Ce qui permettra de prédire des valeurs de Y à partir de valeurs de X : $\hat{Y} = \hat{a} + \hat{\beta}X$

- Modèle de régression binomial avec comme variable de sortie une variable Y ayant 2 classes
- La probabilité d'appartenance à la classe est fonction de variables prédictives X_p (continues, catégorielles)
- Fonction logit (softmax)
- Un algorithme va trouver les meilleurs coefficients β_p associés aux variables X_p pour minimiser les erreurs entre la prédiction faite à partir de ces variables et la vraie valeur prise Y

Régression Logistique

(2)

$Y \{0, 1\}, X_P$: observés

$$\hookrightarrow Y \sim \hat{\beta}_1 X_1 + \hat{\beta}_2 X_2 + \hat{\beta}_3 X_3 + \dots$$

$\hat{\beta}_P$: estimés au vue de ces données

$\hat{\beta}_P$: estimés ; Y, X_P : observés sur d'autres données

$$\hookrightarrow Y \sim \hat{\beta}_1 X_1 + \hat{\beta}_2 X_2 + \hat{\beta}_3 X_3 + \dots$$

$\hat{Y} \{0, 1\}$: prédits pour ces autres données

		\hat{Y}	
		0	1
Y	0	VN	FP
	1	FN	VP

Matrice de confusion

Erreurs de classement : FN, FP

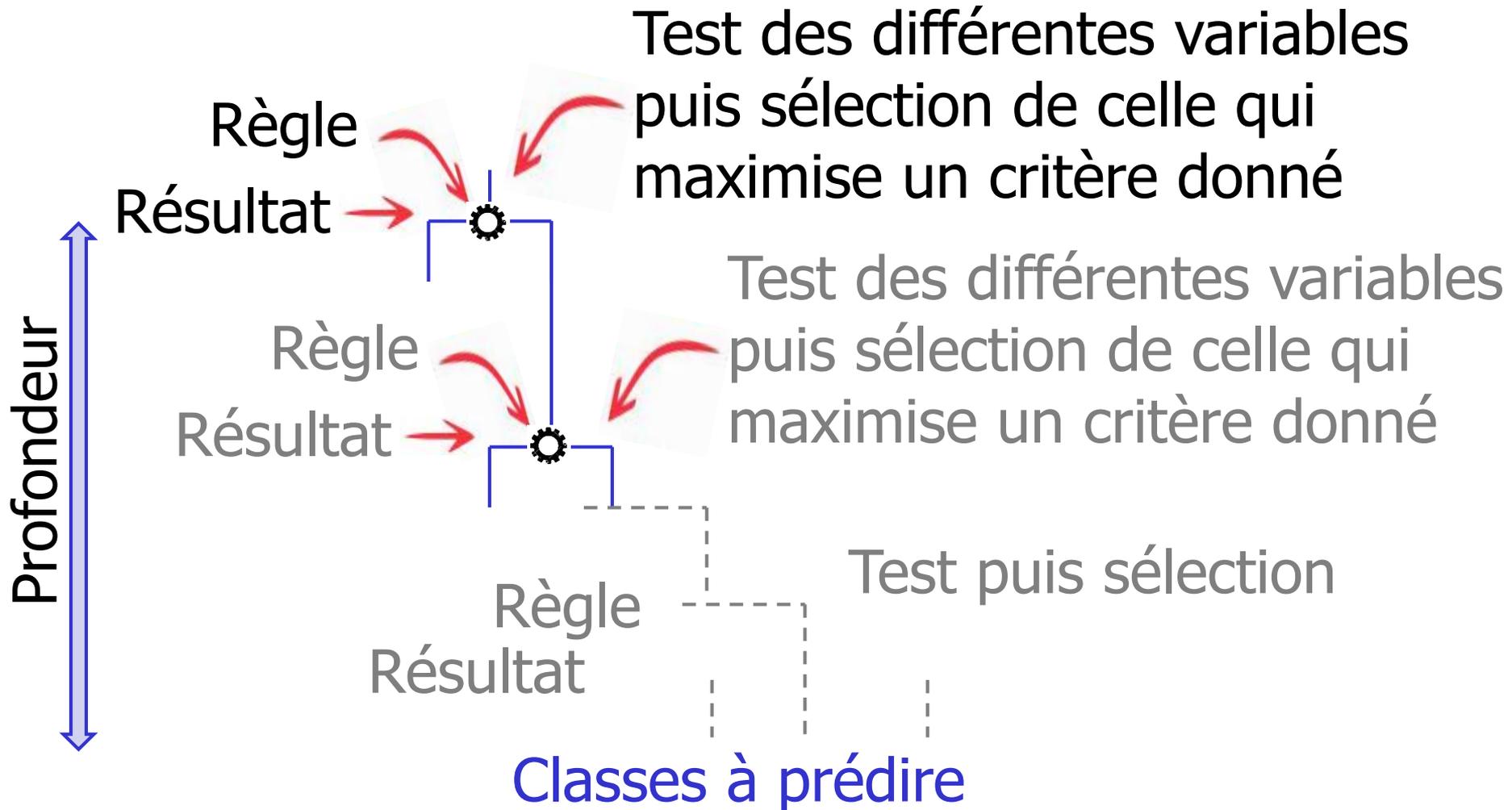
Vrais Négatifs, Faux Positifs, Faux Négatifs, Vrais Positifs

- Utilisation d'un arbre de décision comme modèle prédictif
- Classification d'observations futures connaissant des observations déjà classées
- Constitution d'un arbre
 - ✓ Nœud
 - Représente une règle attachée à la valeur d'une variable
 - Règle qui divise au mieux l'ensemble des observations
 - ✓ Profondeur
 - Nombre maximum de nœuds avant d'atteindre une feuille
 - ✓ Feuille
 - Classe de la variable de sortie

Arbre de Décision

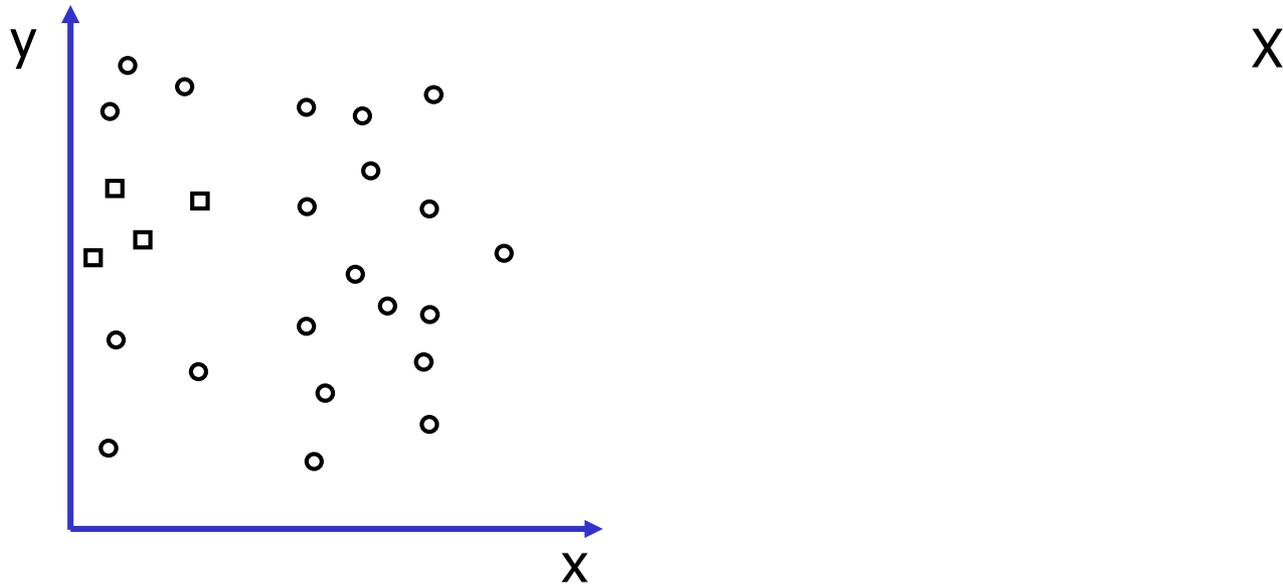
(2)

Ensemble des variables d'apprentissage



- Critères de segmentation
 - ✓ Caractérisent l'homogénéité (gain en homogénéité) des sous-classes obtenues par la segmentation
- Pour des variables catégorielles et/ou continues
- Principe de parcimonie
 - ✓ Arbre qui soit le plus petit possible avec la performance optimale
 - ✓ Objectif de stabilité dans ses prévisions futures
- Pré- et post-élagage
 - ✓ Arrêt de la segmentation selon critères d'arrêt
 - ✓ Réduction du nombre de feuilles

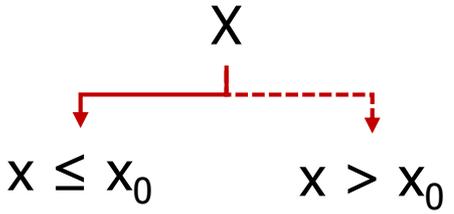
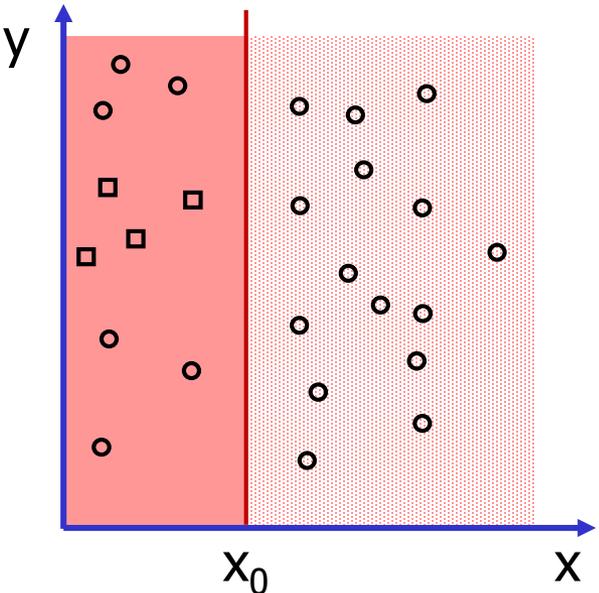
Exemple fictif



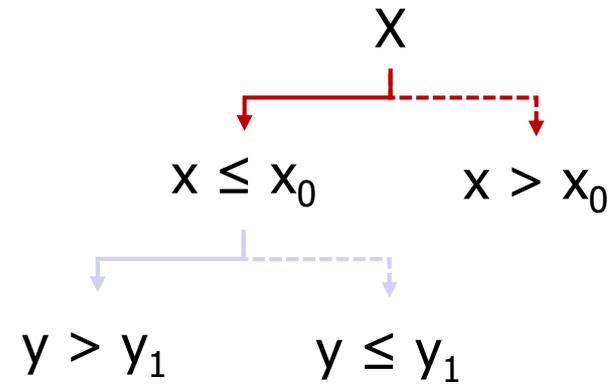
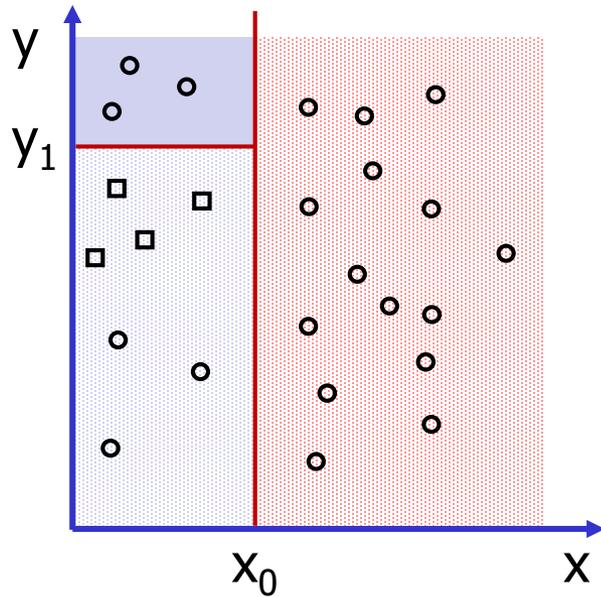
Arbre de Décision

(5)

Exemple fictif



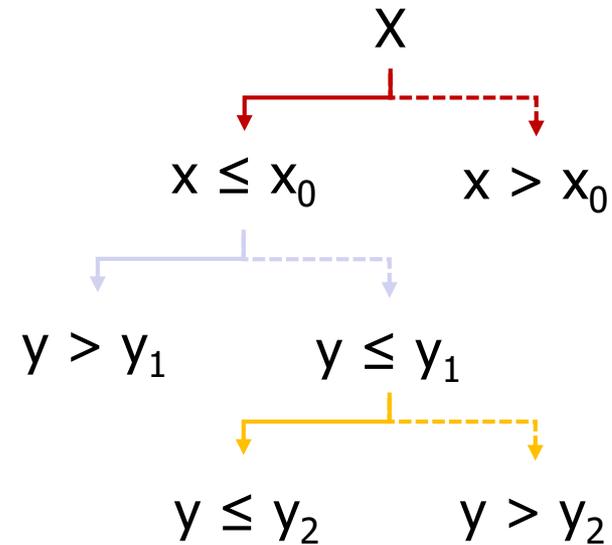
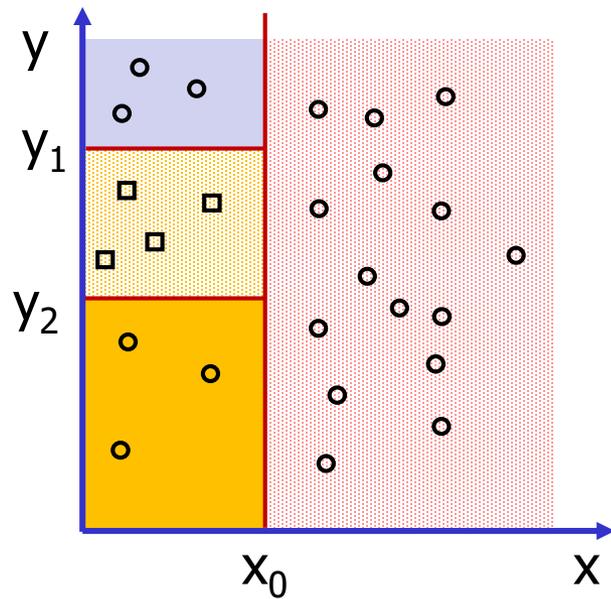
Exemple fictif



Arbre de Décision

(7)

Exemple fictif

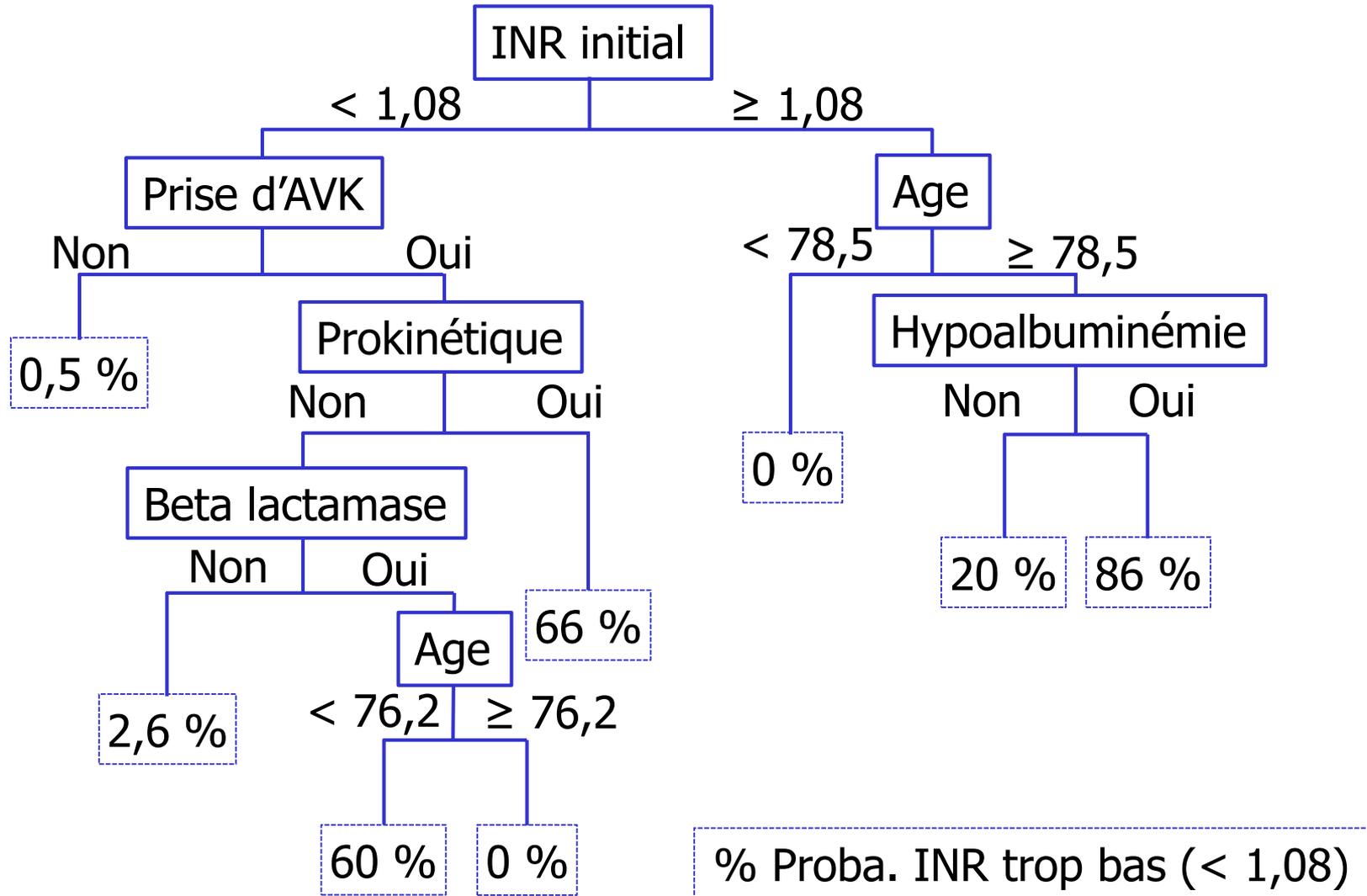


Exemple : data-mining-based detection of adverse drug events (Chazard E et al. *Stud Health Technol Inform* 2009;150:552-6.)

- Objectif d'améliorer la production de connaissances épidémiologiques sur les accidents liés aux médicaments
- Données de bases hospitalières
- Variable à prédire (classes d'appartenance)
 - ✓ INR trop bas ($< 1,08$ oui / non)
- Parmi les variables prédictives candidates
 - ✓ INR initial (cont.), age (cont.), prise d'AVK (oui / non), prise prokinétique (oui / non), prise beta lactamase (oui / non), présence hypoalbuminémie (oui / non)

Arbre de Décision

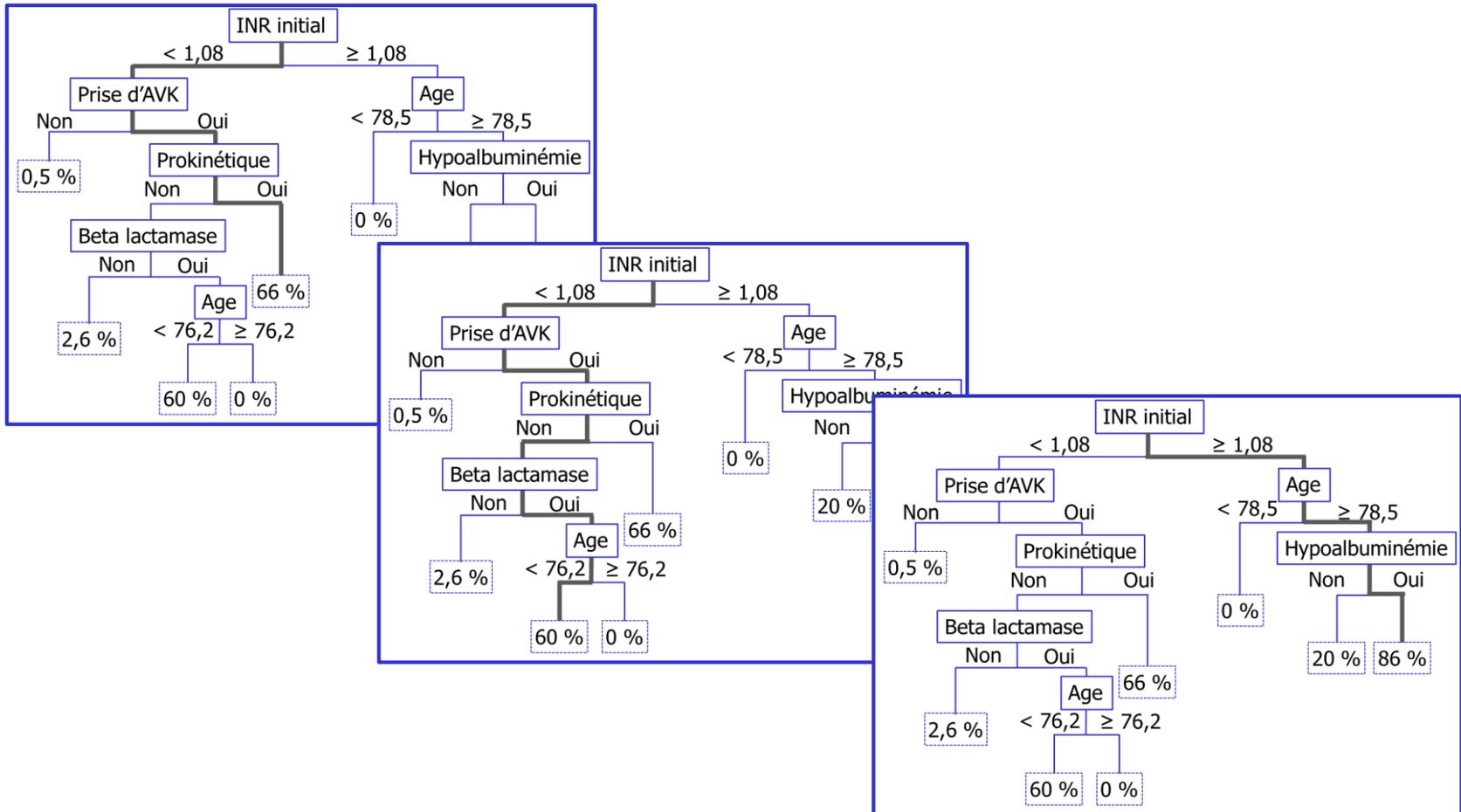
(9)



Arbre de Décision

(10)

Règles conduisant à une augmentation de la probabilité d'avoir un INR trop bas

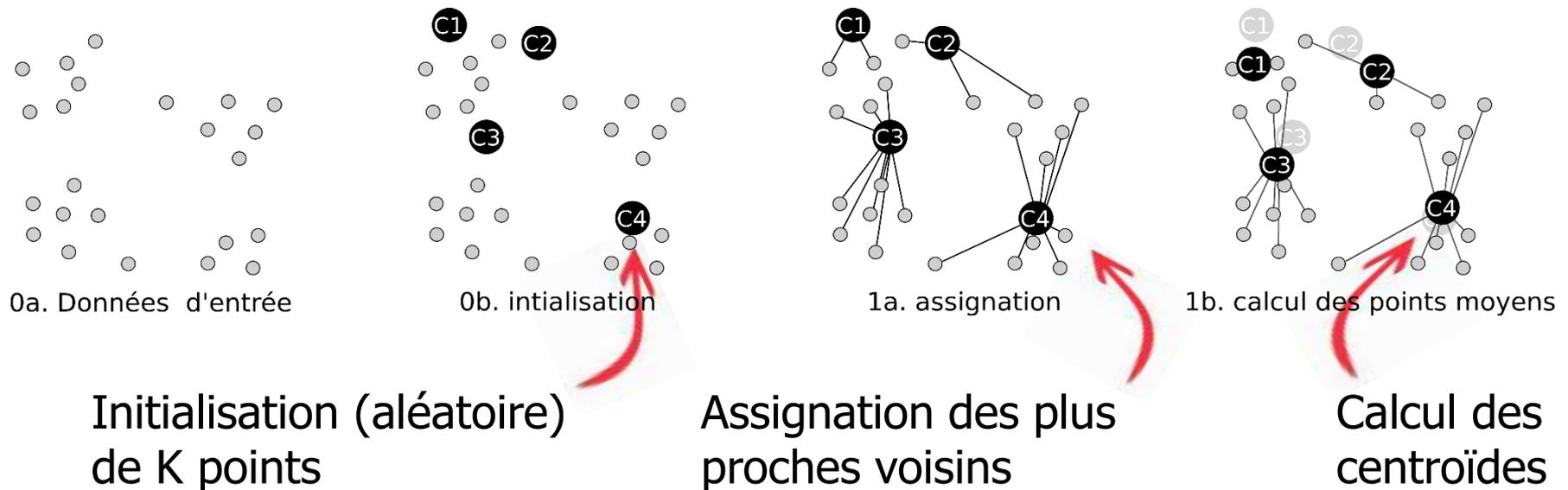


- D'un arbre de décision vers une forêt d'arbres
 - ✓ Arbre de décision et risque de surapprentissage (cf. plus loin)
- Etapes de la construction de la forêt
 1. Tirage au sort aléatoire, avec remise, de N observations du jeu de données initial
 2. Tirage au sort aléatoire de m variables parmi les p disponibles
 3. Sur cet échantillon, on entraîne un arbre de décision
 4. Répétition des étapes 1. à 3. B fois de manière à obtenir B arbres

- Chaque arbre propose une classification (prédiction) différente
- Classification (prédiction) finale est un simple vote majoritaire (ensemble learning)
 - ✓ Classe la plus représentée (moyenne des prédictions)
- Inconvénient
 - ✓ Perte de l'aspect visuel des arbres de décision uniques

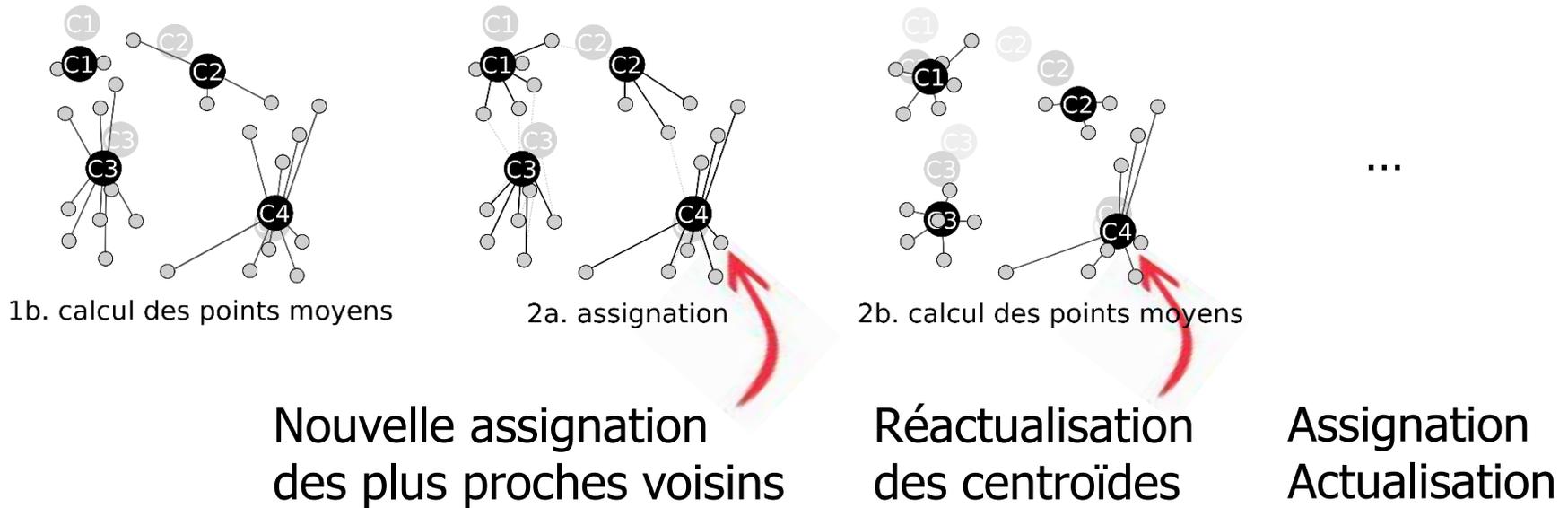
- K plus proches voisins
- Algorithme de clustering non supervisé
 - ✓ Identification de clusters de patients atteints de diabète de type 2 dans la population générale (Carrillo-Larco M et al. *BMJ Open Diabetes Res Care* 2021;9(1):e001889)
 - ✓ Identification de clusters spatiaux de microcéphalie liée au virus Zika (Souza WW et al. *BMC Public Health* 2018;18(1):130)
- Observations divisées en k clusters (groupes)
 - ✓ Groupes d'observations qui se ressemblent, sont similaires en fonction de leurs caractéristiques
 - ✓ K choisi selon critères mathématiques
 - ✓ De manière à minimiser certaine fonction

Principe intuitif de l'algorithme



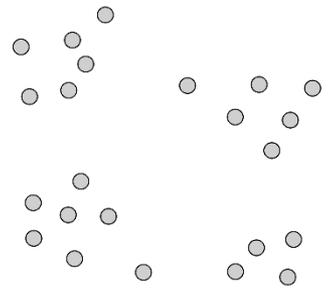
Crédit : Mquantin, CC BY-SA 4.0, via Wikimedia Commons

Principe intuitif de l'algorithme



Crédit : Mquantin, CC BY-SA 4.0, via Wikimedia Commons

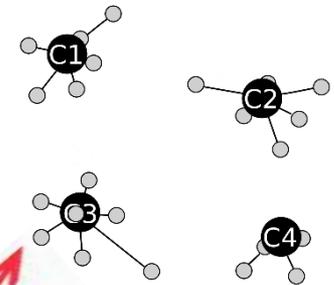
Principe intuitif de l'algorithme



0a. Données d'entrée

Assignation
Actualisation

Jusqu'à



4a. assignation
clusters stables (fin)

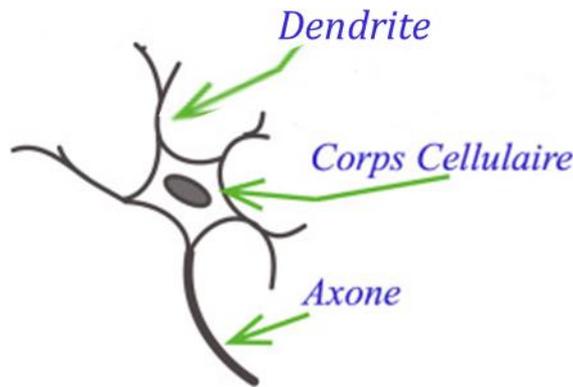
Stabilisation de
l'algorithme

Crédit : Mquantin, CC BY-SA 4.0, via Wikimedia Commons

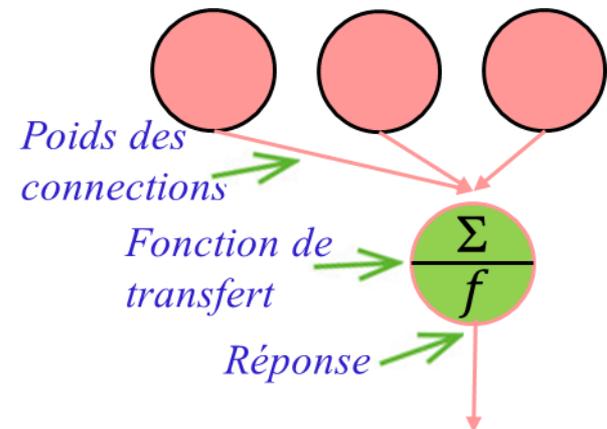
Réseau de Neurones

(1)

- Domaine du deep-learning
- Inspiré du fonctionnement d'un neurone biologique
- Pour concevoir un neurone formel



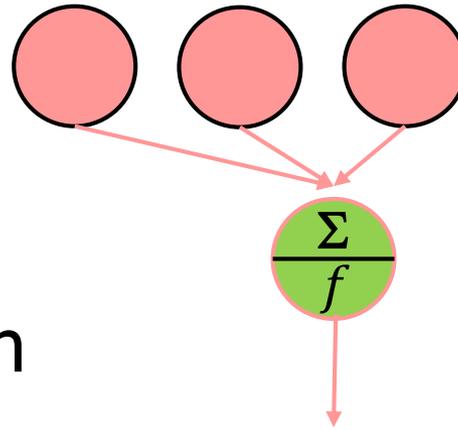
Entrée
Traitement
Sortie



Le Perceptron

(1)

Données d'entrée



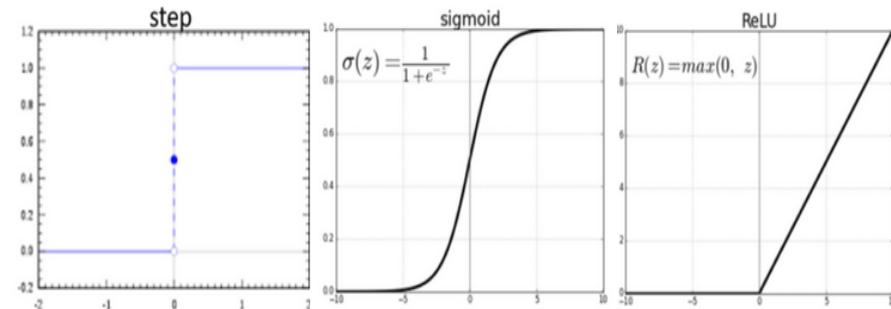
Poids attribués aux données d'entrée

Fonction d'activation du neurone

Propagation du neurone

Somme pondérée des poids avec les données d'entrée

Règle d'activation du neurone



- Le perceptron est un réseau simple
 - ✓ Un seul neurone (une seule couche)
- Apprentissage par propagation
 - ✓ Vers l'avant avec les poids fixés initialement
 - ✓ Vers l'arrière avec calcul des poids de manière à minimiser les erreurs

Réseau de Neurones

(2)

- Concevoir un neurone formel et organiser un réseau de neurones

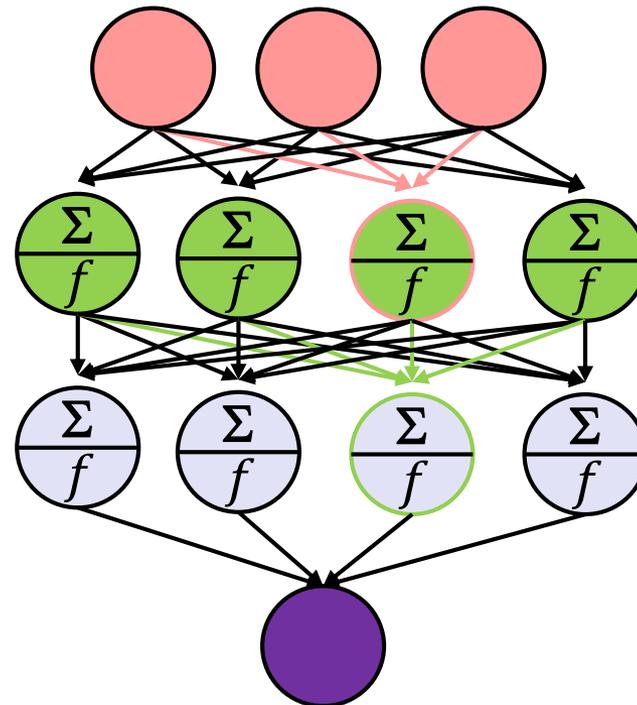
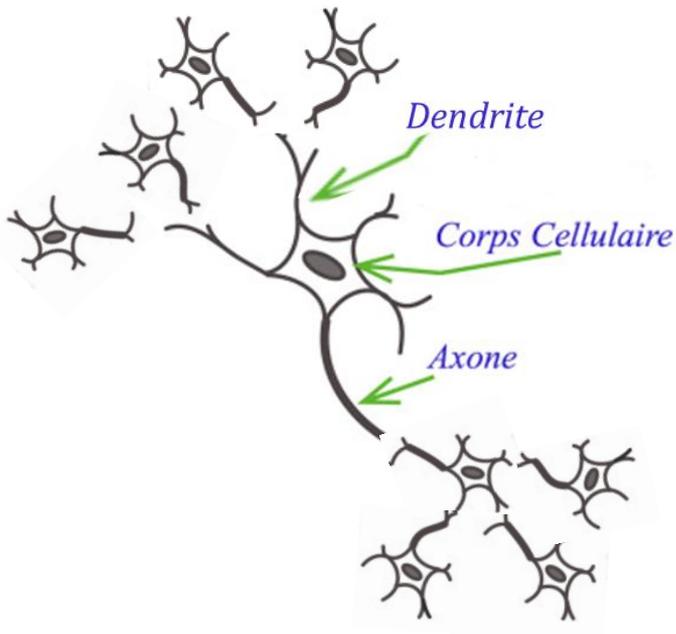
Couche

Entrée

Cachée 1

Cachée 2

Sortie



- Ajout de couches et de neurones par couche
 - ✓ Couches « cachées » faisant le lien entre les entrées et les sorties
- Profondeur du réseau
 - ✓ Ajout des couches pour permettre au modèle de découvrir et d'apprendre des relations de plus en plus complexes
- Apprentissage par propagation
 - ✓ Vers l'avant
 - ✓ Vers l'arrière

Pertinence et Qualité des Algorithmes (1)

- Il faut des données, beaucoup, pour apprendre
 - ✓ Au bout de combien de dictées n'avez-vous plus fait de fautes d'orthographe ?
- Diversité et qualité de l'étiquetage (annotation) des attributs des données
 - ✓ Travail en amont conséquent
- Qualité des données
 - ✓ Données manquantes, valeurs aberrantes, erreurs potentielles

Pertinence et Qualité des Algorithmes (2)

- Un modèle doit apprendre à classer
 - ✓ Classer les observations dans le bon groupe
 - ✓ Ne pas classer d'observations dans le mauvais groupe
 - ✓ Effectué sur un échantillon d'apprentissage
- Il ne doit pas surapprendre (overfitting, biais d'optimisme)
 - ✓ Grande capacité à « capturer » des informations
 - ✓ Perte dans la généralisation des caractéristiques des données, la prédiction sur de nouveaux échantillons

Pertinence et Qualité des Algorithmes (3)

- Comment éviter le surapprentissage ?
 - ✓ Séparer les données en un échantillon d'apprentissage et un échantillon de validation
 - Validation interne, externe, croisée
 - ✓ Pénalisation des valeurs extrêmes
 - Régulation Ridge, Lasso
 - ✓ Ajouter des données d'entraînements
 - ✓ Retirer les variables qui sont peu, pas, informatives, corrélées entre elles
 - ✓ Pratiquer l'early stopping
 - Arrêter l'entraînement avant qu'il ne commence à overfit

