Apprentissage automatique supervisé

Brice Chardin

2022–2023

INTRODUCTION

APPRENTISSAGE AUTOMATIQUE

Apprentissage : qualité d'une entité à se modifier de manière à espérer que ses performances futures s'améliorent

APPRENTISSAGE AUTOMATIQUE

Apprentissage : qualité d'une entité à se modifier de manière à espérer que ses performances futures s'améliorent

Apprentissage nécessaire si :

- spécification initiale difficile, définition informelle basée sur des exemples (*exemple : reconnaître la lettre 'a'*)
- spécifications évoluant en fonction du temps ou de l'environnement (*exemple : identifier des messages indésirables*)

APPRENTISSAGE AUTOMATIQUE SUPERVISÉ

Objectif : extraire de données expérimentales une approximation satisfaisante d'une fonction (un modèle)

Apprentissage $\rightarrow$ l'approximation s'améliore avec l'intégration de données supplémentaires

APPRENTISSAGE AUTOMATIQUE SUPERVISÉ

Objectif : extraire de données expérimentales une approximation satisfaisante d'une fonction (un modèle)

Apprentissage $\rightarrow$ l'approximation s'améliore avec l'intégration de données supplémentaires

Possibilité de consulter le modèle pour d'autres entrées (inférence)

GÉNÉRALISATION EN DEHORS DU DOMAINE D'APPRENTISSAGE

Exemple d'une fonction booléenne à deux variables $y = f(x_1, x_2)$

- $f(\text{Faux}, \text{Faux}) = \text{Faux}$
- $f(\text{Vrai}, \text{Faux}) = \text{Vrai}$
- $f(\text{Vrai}, \text{Vrai}) = \text{Vrai}$

GÉNÉRALISATION EN DEHORS DU DOMAINE D'APPRENTISSAGE

Exemple d'une fonction booléenne à deux variables $y = f(x_1, x_2)$

- $f(\text{Faux}, \text{Faux}) = \text{Faux}$
- $f(\text{Vrai}, \text{Faux}) = \text{Vrai}$
- $f(\text{Vrai}, \text{Vrai}) = \text{Vrai}$

Si $f(\text{Faux}, \text{Vrai}) = \text{Faux}$, alors $f = x_1$
Si $f(\text{Faux}, \text{Vrai}) = \text{Vrai}$, alors $f = x_1 \vee x_2$

GÉNÉRALISATION EN DEHORS DU DOMAINE D'APPRENTISSAGE

Exemple d'une fonction booléenne à deux variables $y = f(x_1, x_2)$

- $f(\text{Faux}, \text{Faux}) = \text{Faux}$
- $f(\text{Vrai}, \text{Faux}) = \text{Vrai}$
- $f(\text{Vrai}, \text{Vrai}) = \text{Vrai}$

Si $f(\text{Faux}, \text{Vrai}) = \text{Faux}$, alors $f = x_1$
Si $f(\text{Faux}, \text{Vrai}) = \text{Vrai}$, alors $f = x_1 \vee x_2$

Fonctions à n paramètres booléens : 2^n entrées possibles (ensemble E)

$\rightarrow |\mathcal{P}(E)| = 2^{2^n}$ fonctions booléennes distinctes

$\rightarrow$ Seul moyen de les distinguer : tester les 2^n entrées possibles

GÉNÉRALISATION

Aucune généralisation (extension de f en dehors du domaine d'entraînement) n'est a priori objectivement meilleure qu'une autre [1]

$\rightarrow$ Un problème d'apprentissage automatique supervisé est, systématiquement, mathématiquement mal posé

1. Yann Le Cun. Modèles connexionnistes de l'apprentissage. Thèse de Doctorat, Université Paris 6 (1987).

GÉNÉRALISATION

Aucune généralisation (extension de f en dehors du domaine d'entraînement) n'est a priori objectivement meilleure qu'une autre [1]

$\rightarrow$ Un problème d'apprentissage automatique supervisé est, systématiquement, mathématiquement mal posé

$\rightarrow$ Nécessité d'introduire un *biais* a priori

1. Yann Le Cun. Modèles connexionnistes de l'apprentissage. Thèse de Doctorat, Université Paris 6 (1987).

CLASSE DE FONCTION (OU DE MODÈLE)

Biais : hypothèses sur la classe de fonction de f

- biais absolus $\rightarrow$ restriction de classe (exemple : f est 2-CNF [2])
- biais de préférence $\rightarrow$ classement des solutions, explicite ou implicite (exemple : nombre moyen d'éléments à modifier pour inverser la sortie)

2. Par exemple : $f = (x_0 \vee x_2) \wedge (x_0 \vee \neg x_3) \wedge (\neg x_1 \vee x_3)$

CLASSE DE FONCTION (OU DE MODÈLE)

Biais : hypothèses sur la classe de fonction de f

- biais absolus $\rightarrow$ restriction de classe (exemple : f est 2-CNF [2])
- biais de préférence $\rightarrow$ classement des solutions, explicite ou implicite
  (exemple : nombre moyen d'éléments à modifier pour inverser la sortie)

Choix de la classe de modèle difficile

Quantification possible de la pertinence du choix a posteriori

Techniques pour aider à choisir (par exemple : optimisation des hyperparamètres)

2. Par exemple : $f = (x_0 \vee x_2) \wedge (x_0 \vee \neg x_3) \wedge (\neg x_1 \vee x_3)$

REPRÉSENTATION INTERNE DU MODÈLE

Problème d'apprentissage supervisé

$\rightarrow$ Problème de généralisation

$\rightarrow$ Capacité du système à construire une représentation interne adaptée

Mémorisation intégrale ou modèle 1-NN (inférence au plus proche voisin)

Pour une distance d et un ensemble d'entraînement E

$$\hat{y} = f(\mathbf{x}) = f\left(\underset{\mathbf{x'} \in E}{\operatorname{argmin}}\, d(\mathbf{x'}, \mathbf{x})\right)$$

REPRÉSENTATION INTERNE DU MODÈLE

Problème d'apprentissage supervisé

→ Problème de généralisation

→ Capacité du système à construire une représentation interne adaptée

Mémorisation intégrale ou modèle 1-NN (inférence au plus proche voisin)

Pour une distance d et un ensemble d'entraînement E

$$\hat{y} = f(\mathbf{x}) = f\left(\underset{\mathbf{x'} \in E}{\operatorname{argmin}}\, d(\mathbf{x'}, \mathbf{x})\right)$$

→ Capacité de généralisation limitée

→ Peu efficace pour un jeu d'entraînement important

REPRÉSENTATION INTERNE DU MODÈLE

Régression linéaire

$$\hat{y} = f(\mathbf{x}) = a_0 + \sum_{i=1}^{n} a_i x_i$$

Représentation interne : paramètres $\{a_i\}_{0 \leq i \leq n}$

REPRÉSENTATION INTERNE DU MODÈLE

Régression linéaire

$$\hat{y} = f(\mathbf{x}) = a_0 + \sum_{i=1}^{n} a_i x_i$$

Représentation interne : paramètres $\{a_i\}_{0 \leq i \leq n}$

Problème d'apprentissage supervisé :

- choix d'un espace de représentation
- stratégie de recherche d'une solution dans cet espace (sur la base de critères subjectifs censés favoriser la généralisation)

Difficulté du problème d'apprentissage lié à l'espace des représentations possibles, et non à la complexité intrinsèque du problème

CATÉGORIES D'APPRENTISSAGE

CATÉGORIES D'APPRENTISSAGE

Si la sortie est une variable catégorielle (qualitative) $\rightarrow$ problème de **classification**

- classification binaire si seulement deux catégories
- classification multi-classes sinon

CATÉGORIES D'APPRENTISSAGE

Si la sortie est une variable catégorielle (qualitative) $\rightarrow$ problème de **classification**

- classification binaire si seulement deux catégories
- classification multi-classes sinon

Catégorisation de types d'iris (multi-classes)

$y \in \{\text{setosa, versicolor, virginica}\}$

CATÉGORIES D'APPRENTISSAGE

Si la sortie est une variable catégorielle (qualitative) $\rightarrow$ problème de **classification**

- classification binaire si seulement deux catégories
- classification multi-classes sinon

Catégorisation de types d'iris (multi-classes)

$y \in \{\text{setosa, versicolor, virginica}\}$

Si la sortie est une variable numérique (quantitative) $\rightarrow$ problème de **régression**

CATÉGORIES D'APPRENTISSAGE : SUPERVISION

Apprentissage supervisé : apprendre à partir d'exemples d'entrées et de sorties

Reconnaissance de chiffres manuscrits

Données de références annotées

0 3 7 2 0

CATÉGORIES D'APPRENTISSAGE : SUPERVISION

Apprentissage supervisé : apprendre à partir d'exemples d'entrées et de sorties

Reconnaissance de chiffres manuscrits

Données de références annotées

0 3 7 2 0

Apprentissage non supervisé : apprendre uniquement à partir d'exemples d'entrées

Reconnaissance de chiffres manuscrits

Identifier dix groupes en espérant que chacun corresponde à un chiffre

Peu applicable aux problèmes de régression $\rightarrow$ synonyme de partitionnement ou *clustering*

CATÉGORIES D'APPRENTISSAGE : SUPERVISION

Apprentissage semi-supervisé : apprendre d'exemples partiellement annotés

Données annotées $\rightarrow$ identifier les classes

Données non annotées $\rightarrow$ affiner la séparation

$\rightarrow$ identifier des points aberrants, isolés

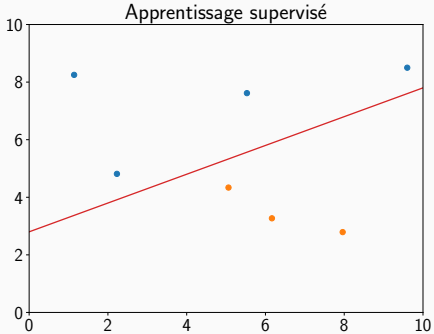
CATÉGORIES D'APPRENTISSAGE : SUPERVISION

Apprentissage semi-supervisé : apprendre d'exemples partiellement annotés

Données annotées → identifier les classes

Données non annotées → affiner la séparation

→ identifier des points aberrants, isolés



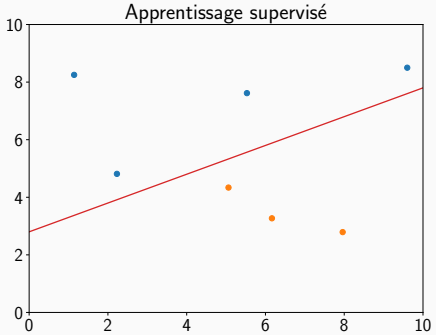
CATÉGORIES D'APPRENTISSAGE : SUPERVISION

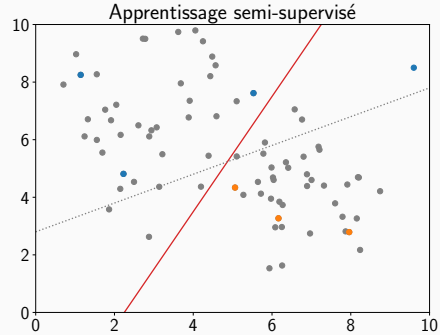
Apprentissage semi-supervisé : apprendre d'exemples partiellement annotés

Données annotées → identifier les classes

Données non annotées → affiner la séparation

→ identifier des points aberrants, isolés





CATÉGORIES D'APPRENTISSAGE : SUPERVISION

Apprentissage actif : identifier des exemples à annoter

Sélectionner les échantillons à annoter (par un humain)

- optimiser l'apport de la nouvelle annotation
- minimiser le nombre de demandes

CATÉGORIES D'APPRENTISSAGE : SUPERVISION

Apprentissage par renforcement : apprendre une stratégie optimale

- automate fini avec un ensemble d'états possibles $E = \{\mathbf{e}_i\}$ et d'actions $A = \{\mathbf{a}_i\}$
- table de transitions calculable $(\mathbf{e}_i \xrightarrow{\mathbf{a}} \mathbf{e}_j)$
- fonction de récompense $r(\mathbf{e})$

Objectif : trouver une fonction $f : E \to A, f(\mathbf{e}_i) = \mathbf{a}$ identifiant l'action $\mathbf{a}$ maximisant la récompense $r(\mathbf{e}_j)$ associée à l'état obtenu après transition

$\to$ Éviter une recherche exhaustive

$\to$ Mécanismes possibles pour compléter r si valeur connue pour un nombre limité d'états

EXEMPLES DE CATÉGORIES D'APPRENTISSAGE

EXEMPLES DE TYPES D'APPRENTISSAGE

Identifier l'espèce d'une plante

- Entrée : 
- Sortie : *Ruscus aculeatus*

EXEMPLES DE TYPES D'APPRENTISSAGE

Identifier l'espèce d'une plante

- Entrée : 
- Sortie : *Ruscus aculeatus*

→ problème de **classification**

EXEMPLES DE TYPES D'APPRENTISSAGE

Déterminer la résistance à la traction d'un film plastique

- Entrée :
  - Composition (HDPE : 81%, $CaCO_3$: 16%, Copolymère : 3%)
  - Procédé de fabrication
    - Taille moyenne des particules de $CaCO_3$: 7 µm
    - T1 : 173 °C, T2 : 166 °C, T3 : 192 °C, T4 : 155 °C
    - Vitesse de mélange : 35 rpm
- Sortie : 9,6 MPa

EXEMPLES DE TYPES D'APPRENTISSAGE

Déterminer la résistance à la traction d'un film plastique

- Entrée :
  - Composition (HDPE : 81%, $CaCO_3$: 16%, Copolymère : 3%)
  - Procédé de fabrication
    - Taille moyenne des particules de $CaCO_3$: 7 µm
    - T1 : 173 °C, T2 : 166 °C, T3 : 192 °C, T4 : 155 °C
    - Vitesse de mélange : 35 rpm
- Sortie : 9,6 MPa

→ problème de **régression**

EXEMPLES DE TYPES D'APPRENTISSAGE

Prédire le nombre d'étudiants en cours

- Entrée : DOI, B. Chardin, mercredi, 8h, séance no. 8
- Sortie : 9

EXEMPLES DE TYPES D'APPRENTISSAGE

Prédire le nombre d'étudiants en cours

- Entrée : DOI, B. Chardin, mercredi, 8h, séance no. 8
- Sortie : 9

Variable de sortie discrète mais non catégorielle

$\rightarrow$ problème de **régression**

EXEMPLES DE TYPES D'APPRENTISSAGE

Recommander une classe pour un trajet en avion

- Entrée :
  - Profil de l'acheteur (salaire est., catégorie socioprofessionnelle, etc.)
  - Caractéristiques du trajet (distance, nb. voyageurs, etc.)
- Sorties possibles : économique, économique premium, affaires ou première

EXEMPLES DE TYPES D'APPRENTISSAGE

Déterminer la classe d'un groupe de passagers du Titanic

- Entrée : nationalité, civilité, nb. de passagers, âge moyen, etc.
- Sorties possibles : 1re, 2e ou 3e classe

EXEMPLES DE TYPES D'APPRENTISSAGE

Déterminer la classe d'un groupe de passagers du Titanic

- Entrée : nationalité, civilité, nb. de passagers, âge moyen, etc.
- Sorties possibles : 1re, 2e ou 3e classe

Variable de sortie à la fois catégorielle et ordinale

$\rightarrow$ problème de **régression** ou de **classification**

Exemple de transformation

économique $\rightarrow$ 0

économique premium $\rightarrow$ 1

affaires $\rightarrow$ 2

première $\rightarrow$ 3

EXEMPLES DE TYPES D'APPRENTISSAGE

Déterminer un niveau de satisfaction

- Sorties possibles : très satisfait, satisfait, insatisfait, très insatisfait

EXEMPLES DE TYPES D'APPRENTISSAGE

Déterminer un niveau de satisfaction

- Sorties possibles : très satisfait, satisfait, insatisfait, très insatisfait

$\rightarrow$ problème de **régression** ou de **classification**

Exemple de transformation (intervalles non uniformes)

très satisfait $\rightarrow 9$

satisfait $\rightarrow 6$

insatisfait $\rightarrow 4$

très insatisfait $\rightarrow 0$

EXEMPLES DE TYPES D'APPRENTISSAGE

Déterminer un niveau de satisfaction

- Sorties possibles : très satisfait, satisfait, insatisfait, très insatisfait ou *non applicable*

EXEMPLES DE TYPES D'APPRENTISSAGE

Déterminer un niveau de satisfaction

- Sorties possibles : très satisfait, satisfait, insatisfait, très insatisfait ou *non applicable*

Variable de sortie non ordinale $\rightarrow$ problème de **classification**

Décomposition possible en deux problèmes

- un problème de **classification** (applicabilité)
- un problème de **régression** (niveau de satisfaction)

EXEMPLES DE TYPES D'APPRENTISSAGE

Déterminer la catégorie d'un film plastique

- Entrée : composition, procédé de fabrication
- Sorties possibles : Cat. A ($R_m \geq 11$ MPa), Cat. B ($11\text{MPa} > R_m \geq 9$ MPa), Cat. C ($9\text{MPa} > R_m \geq 7$ MPa) ou NC ($R_m < 7$ MPa)

Cas no. 1 : la valeur de résistance à la traction est connue pour l'apprentissage

Cas no. 2 : seule la catégorie est connue pour l'apprentissage

EXEMPLES DE TYPES D'APPRENTISSAGE

Déterminer la catégorie d'un film plastique

- Entrée : composition, procédé de fabrication
- Sorties possibles : Cat. A ($R_m \geq 11$ MPa), Cat. B ($11\text{MPa} > R_m \geq 9$ MPa), Cat. C ($9\text{MPa} > R_m \geq 7$ MPa) ou NC ($R_m < 7$ MPa)

Cas no. 1 : la valeur de résistance à la traction est connue pour l'apprentissage

$\rightarrow$ problème de **régression** ou de **classification**

Cas no. 2 : seule la catégorie est connue pour l'apprentissage

$\rightarrow$ problème de **classification**

$\rightarrow$ problème de **régression** en associant une valeur de résistance typique à chaque catégorie (par ex. resp. 12, 10, 8 et 6 MPa)

VALIDATION DE MODÈLES

VALIDATION

Comment qualifier un modèle de *valide* ?

VALIDATION

Comment qualifier un modèle de *valide* ?

- parce qu'il donne de bons résultats $\rightarrow$ mesures de qualité
- parce qu'il fait sens, parce qu'il est logique $\rightarrow$ modèle explicable

VALIDATION DES MODÈLES DE RÉGRESSION

Mean absolute error (MAE) $= \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$

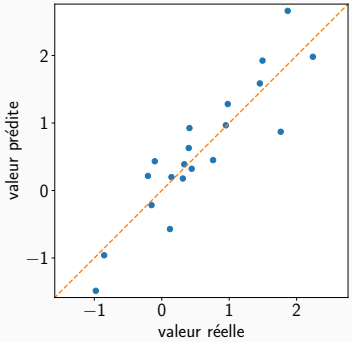
Root mean square error (RMSE) $= \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$

Coefficient de détermination $R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}$

Et autres mesures qui peuvent dépendre des besoins applicatifs :

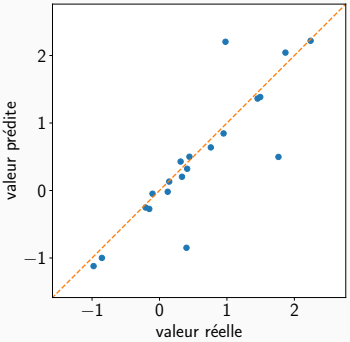
- *mean absolute percentage error* (MAPE)
- plus grande sous-estimation
- plus grande sur-estimation
- biais (tendance à sur- ou sous-estimer)
- etc.

VALIDATION DES MODÈLES DE RÉGRESSION



MAE : 0.33

RMSE : **0.42**



MAE : **0.27**

RMSE : 0.49

VALIDATION DES MODÈLES DE CLASSIFICATION BINAIRE

Exemple d'application en maintenance prédictive (100 observations)

Prédicteur idéal

	Défaillance prédite	
Défaillance réelle	Vrai	Faux
Vrai	3	0
Faux	0	97

Prédicteur aléatoire

	Défaillance prédite	
Défaillance réelle	Vrai	Faux
Vrai	2	1
Faux	46	51

Prédicteur de classe majoritaire

	Défaillance prédite	
Défaillance réelle	Vrai	Faux
Vrai	0	3
Faux	0	97

Autre prédicteur

	Défaillance prédite	
Défaillance réelle	Vrai	Faux
Vrai	2	1
Faux	8	89

VALIDATION DES MODÈLES DE CLASSIFICATION BINAIRE

Matrice de confusion

		Classe prédite
Classe réelle	**Vrai**	**Faux**
Vrai	TP	FN
Faux	FP	TN

- TP : True Positive (vrai positif) ← bien prédit
- TN : True Negative (vrai négatif) ← bien prédit
- FP : False Positive (faux positif) ← mal prédit
- FN : False Negative (faux négatif) ← mal prédit

VALIDATION DES MODÈLES DE CLASSIFICATION BINAIRE

Matrice de confusion

	Classe prédite	
Classe réelle	**Vrai**	**Faux**
Vrai	TP	FN
Faux	FP	TN

- TP : True Positive (vrai positif) ← bien prédit
- TN : True Negative (vrai négatif) ← bien prédit
- FP : False Positive (faux positif) ← mal prédit
- FN : False Negative (faux négatif) ← mal prédit

$$\textbf{précision} = \frac{TP}{TP + FP}$$

$$\textbf{rappel} = \frac{TP}{TP + FN}$$

$$\textbf{spécificité} = \frac{TN}{TN + FP}$$

$$\textbf{exactitude (accuracy)} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\textbf{balanced accuracy (BA)} = \frac{\text{rappel} + \text{spécificité}}{2}$$

$$\textbf{F1-score} = \frac{2 \times \text{précision} \times \text{rappel}}{\text{précision} + \text{rappel}}$$

VALIDATION DES MODÈLES DE CLASSIFICATION BINAIRE

	Classe prédite	
Classe réelle	**Vrai**	**Faux**
Vrai	2	1
Faux	8	89

$$\text{précision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{2}{2+8} = 20\% \rightarrow \text{défaillances prédites souvent fausses}$$

VALIDATION DES MODÈLES DE CLASSIFICATION BINAIRE

	Classe prédite	
Classe réelle	**Vrai**	**Faux**
Vrai	2	1
Faux	8	89

$$\text{précision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{2}{2+8} = 20\% \rightarrow \text{défaillances prédites souvent fausses}$$

$$\text{rappel} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{2}{2+1} = 66.7\% \rightarrow \text{pièces défaillantes parfois classées } \textit{non défaillantes}$$

VALIDATION DES MODÈLES DE CLASSIFICATION BINAIRE

	Classe prédite	
Classe réelle	**Vrai**	**Faux**
Vrai	2	1
Faux	8	89

$$\text{précision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{2}{2+8} = 20\% \rightarrow \text{défaillances prédites souvent fausses}$$

$$\text{rappel} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{2}{2+1} = 66.7\% \rightarrow \text{pièces défaillantes parfois classées } \textit{non défaillantes}$$

$$\text{spécificité} = \frac{\text{TN}}{\text{TN} + \text{FP}} = \frac{89}{89+8} = 91.2\% \rightarrow \text{pièces non défaillantes rarement classées défaillantes}$$

VALIDATION DES MODÈLES DE CLASSIFICATION BINAIRE

Mesures de qualité globales

Prédicteur idéal

Défaillance réelle	Défaillance prédite : Vrai	Défaillance prédite : Faux
Vrai	3	0
Faux	0	97

Exactitude : 100%
BA : 100%
F1 : 100%

Prédicteur aléatoire

Défaillance réelle	Défaillance prédite : Vrai	Défaillance prédite : Faux
Vrai	2	1
Faux	46	51

Exactitude : 53%
BA : 59.6%
F1 : 7.8%

Prédicteur de classe majoritaire

Défaillance réelle	Défaillance prédite : Vrai	Défaillance prédite : Faux
Vrai	0	3
Faux	0	97

Exactitude : 97%
BA : 50%
F1 : 0% (convention)

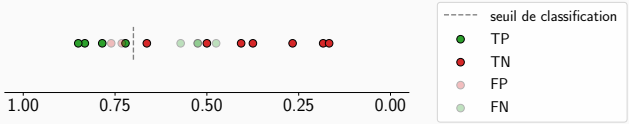
Autre prédicteur

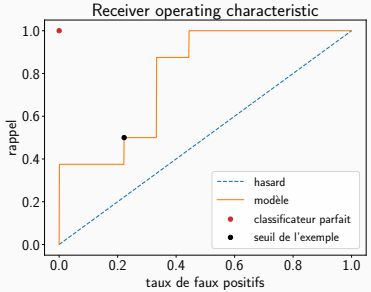
Défaillance réelle	Défaillance prédite : Vrai	Défaillance prédite : Faux
Vrai	2	1
Faux	8	89

Exactitude : 91%
BA : 79.2%
F1 : 30.8%

COURBE ROC

Ajustement du seuil de classification





Seuil à gauche de la courbe : classification à 100% dans la classe rouge (négatif)

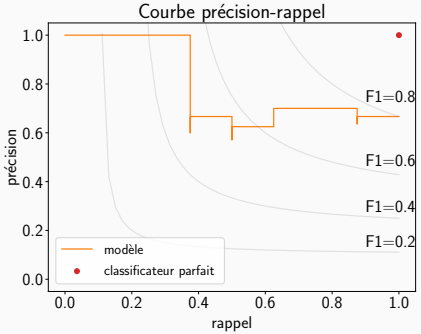
Seuil à droite de la courbe : classification à 100% dans la classe verte (positif)

Mesure de qualité globale : aire sous la courbe (AUC, *Area Under the Curve*)

Taux de faux positifs : $\frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}}$

COURBE PRÉCISION-RAPPEL

Ajustement du seuil de classification



À gauche de la courbe : classification à 100% dans la classe rouge (négatif)

À droite de la courbe : meilleure classification pour laquelle le rappel vaut 1

VALIDATION DES MODÈLES AVEC CLASSES MULTIPLES

Matrice de confusion

	Classe prédite		
Classe réelle	**A**	**B**	**C**
A	15	7	10
B	12	20	5
C	3	2	3

exactitude :

$$\frac{\text{nb éléments bien classés}}{\text{nb éléments total}} = \frac{15 + 20 + 3}{77} = 49.4\%$$

VALIDATION DES MODÈLES AVEC CLASSES MULTIPLES

Matrice de confusion

	Classe prédite		
Classe réelle	**A**	**B**	**C**
A	15	7	10
B	12	20	5
C	3	2	3

Correspondance avec une classification binaire :

$\mathrm{TP}(\mathrm{A}) = 15$

$\mathrm{FP}(\mathrm{A}) = 12 + 3 = 15$

$\mathrm{FN}(\mathrm{A}) = 7 + 10 = 17$

$\mathrm{TN}(\mathrm{A}) = 20 + 2 + 5 + 3 = 30$

$$\mathrm{rappel}(\mathrm{A}) = \frac{\mathrm{TP}(\mathrm{A})}{\mathrm{TP}(\mathrm{A}) + \mathrm{FN}(\mathrm{A})} = \frac{15}{15 + 17} = 46.9\%$$

Matrice de confusion

	Classe prédite		
Classe réelle	**A**	**B**	**C**
A	15	7	10
B	12	20	5
C	3	2	3

Correspondance avec une classification binaire :

$\mathrm{TP}(\mathrm{A}) = 15$

$\mathrm{FP}(\mathrm{A}) = 12 + 3 = 15$

$\mathrm{FN}(\mathrm{A}) = 7 + 10 = 17$

$\mathrm{TN}(\mathrm{A}) = 20 + 2 + 5 + 3 = 30$

$$\mathrm{rappel}(\mathrm{A}) = \frac{\mathrm{TP}(\mathrm{A})}{\mathrm{TP}(\mathrm{A}) + \mathrm{FN}(\mathrm{A})} = \frac{15}{15 + 17} = 46.9\%$$

Mesures globales : moyenne des scores de chaque classe

- mesures non pondérés (poids égaux)
- mesures pondérés par le nombre d'instances réelles

VALIDATION DES MODÈLES AVEC CLASSES MULTIPLES

$$\textbf{rappel} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \qquad \textbf{spécificité} = \frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}}$$

En classification binaire, rappel(P) = spécificité(N) (et inversement)

→ balanced accuracy : moyenne des scores de rappel pour chaque classe

Matrice de confusion

	Classe prédite		
Classe réelle	**A**	**B**	**C**
A	15	7	10
B	12	20	5
C	3	2	3

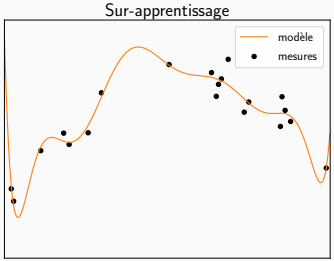
balanced-accuracy (moyenne des rappels) :

$$\frac{1}{3}\left(\frac{15}{15+7+10}+\frac{20}{12+20+5}+\frac{3}{3+2+3}\right)$$

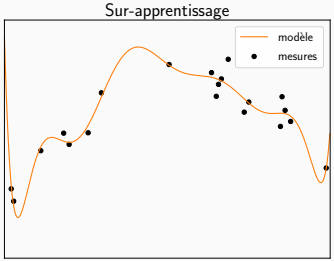
$$\frac{1}{3}(0.469+0.541+0.375) = 46.1\%$$

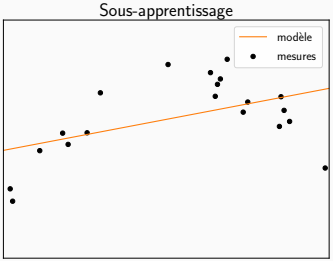
MÉTHODOLOGIE DE VALIDATION

SUR- ET SOUS-APPRENTISSAGE



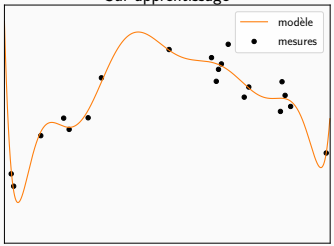
SUR- ET SOUS-APPRENTISSAGE



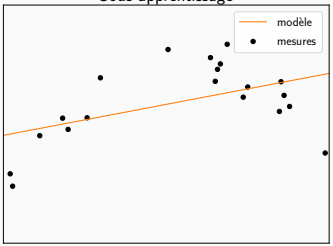


SUR- ET SOUS-APPRENTISSAGE

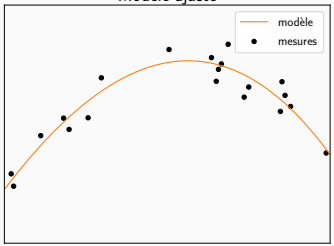
Sur-apprentissage



Sous-apprentissage



Modèle ajusté



MÉTHODOLOGIE DE VALIDATION

Mélange et séparation des données en deux ensembles :

- entraînement ($\sim 80\%$)
- test ($\sim 20\%$)

Évaluation du modèle sur les données de test n'ayant *jamais* été vues par celui-ci

$\rightarrow$ Estimation des performances réelles

VALIDATION CROISÉE

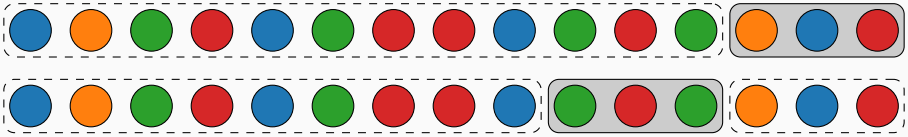
Résultats de validation pouvant varier significativement selon le découpage

Principe : réaliser plusieurs découpages sur lesquels réaliser l'entraînement puis la validation, pour calculer une précision moyenne globale

VALIDATION CROISÉE

Résultats de validation pouvant varier significativement selon le découpage

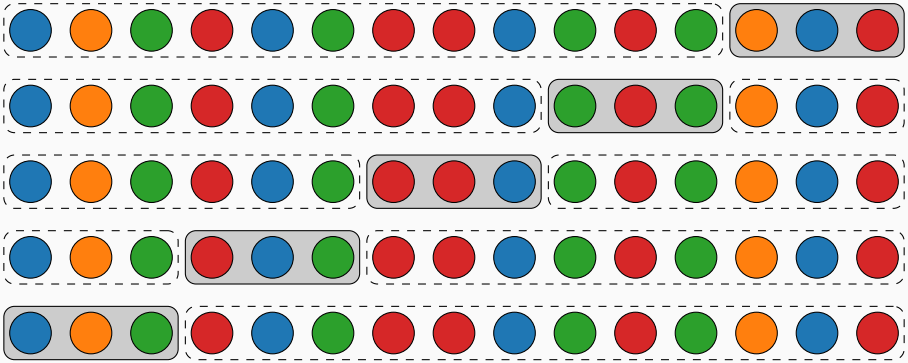
Principe : réaliser plusieurs découpages sur lesquels réaliser l'entraînement puis la validation, pour calculer une précision moyenne globale



VALIDATION CROISÉE

Résultats de validation pouvant varier significativement selon le découpage

Principe : réaliser plusieurs découpages sur lesquels réaliser l'entraînement puis la validation, pour calculer une précision moyenne globale



VALIDATION CROISÉE

Leave one out : $n - 1$ données d'entraînement, 1 donnée de test

Leave p out : $n - p$ données d'entraînement, p données de test

$\rightarrow$ explosion combinatoire : $\binom{n}{p}$ découpages

k-plis (*k-fold*)

- mélange des données
- définition de k partitions
- utilisation de $k - 1$ partitions pour l'entraînement et une partition pour le test

VALIDATION CROISÉE

Avantage : liste de scores et non score unique $\rightarrow$ étude de la variance du modèle

Méthodologie communément recommandée : k-plis (temps de calcul, pertinence)

Variantes de k-plis existantes pour :

- équilibrer les classes dans les plis générés
- équilibrer des groupes de données arbitraires
- traiter des séries temporelles

OPTIMISATION DES HYPER-PARAMÈTRES

Pour les modèles paramétrés (la plupart)

Méthodologie :

- définir une fonction de score pour comparer objectivement les résultats
- fixer l'espace de recherche des paramètres
- définir trois ensembles : entraînement, validation et test

→ Score obtenu sur les données de validation détermine les paramètres optimaux

OPTIMISATION DES HYPER-PARAMÈTRES

Pour les modèles paramétrés (la plupart)

Méthodologie :

- définir une fonction de score pour comparer objectivement les résultats
- fixer l'espace de recherche des paramètres
- définir trois ensembles : entraînement, validation et test

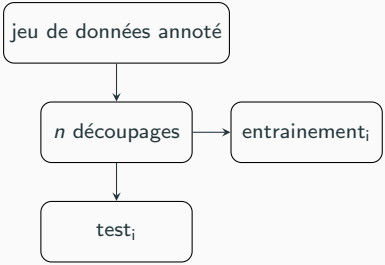
→ Score obtenu sur les données de validation détermine les paramètres optimaux

Nested cross-validation : k-plis imbriqués pour entraînement, validation et test

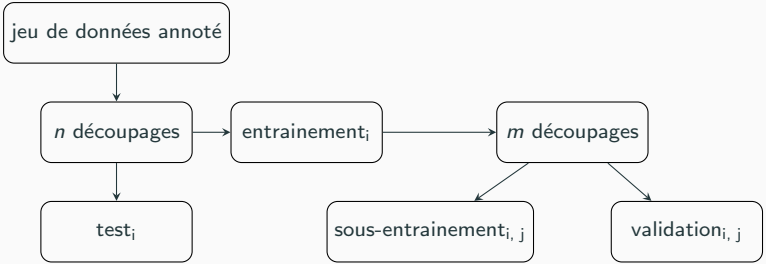
MÉTHODOLOGIE GLOBALE

jeu de données annoté

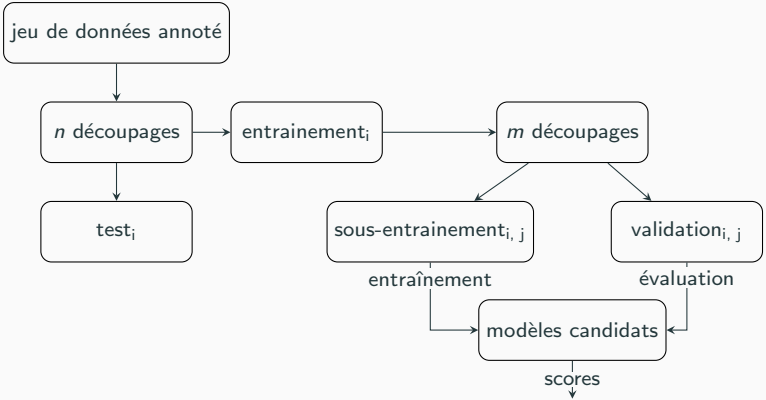
MÉTHODOLOGIE GLOBALE



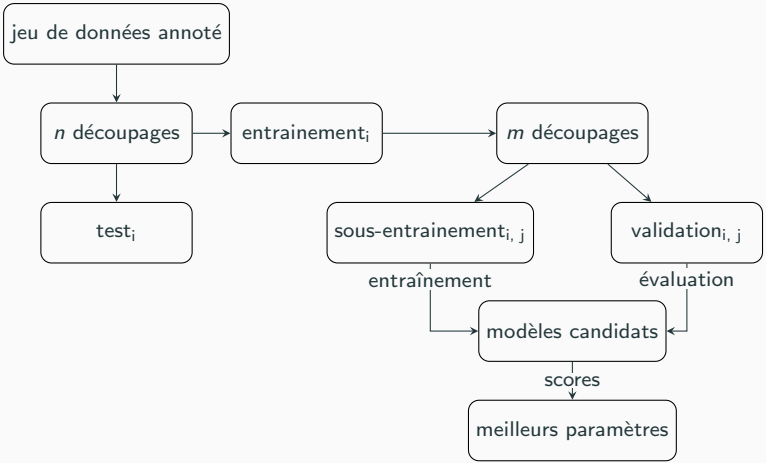
MÉTHODOLOGIE GLOBALE



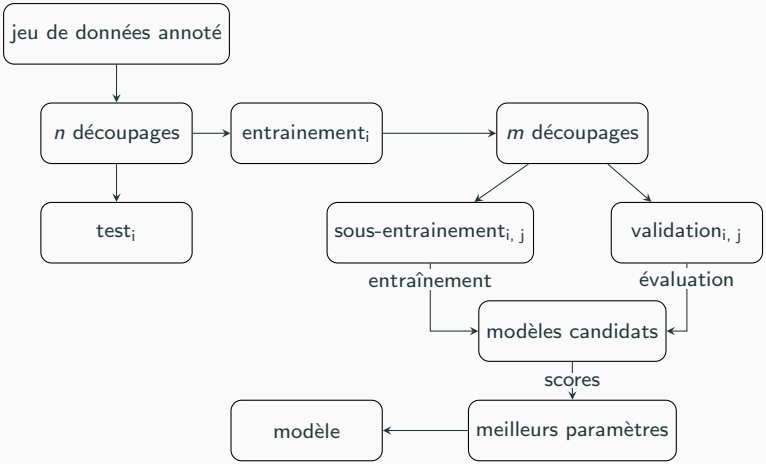
MÉTHODOLOGIE GLOBALE



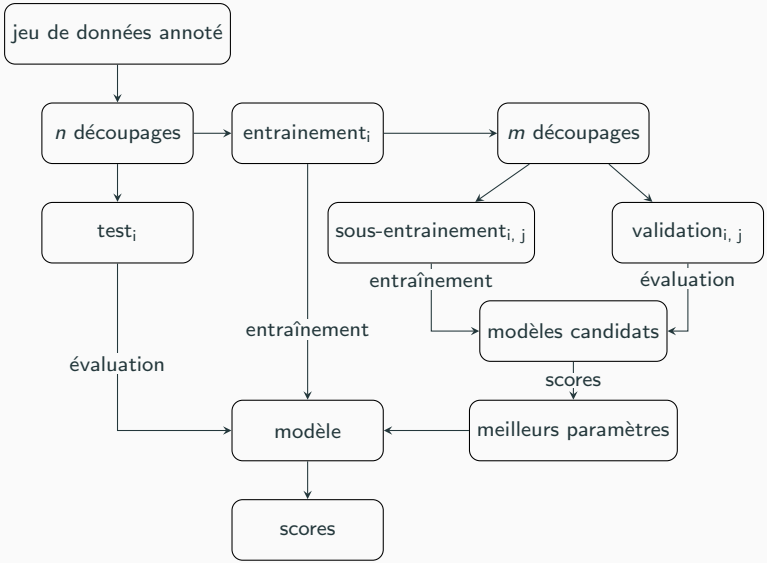
MÉTHODOLOGIE GLOBALE



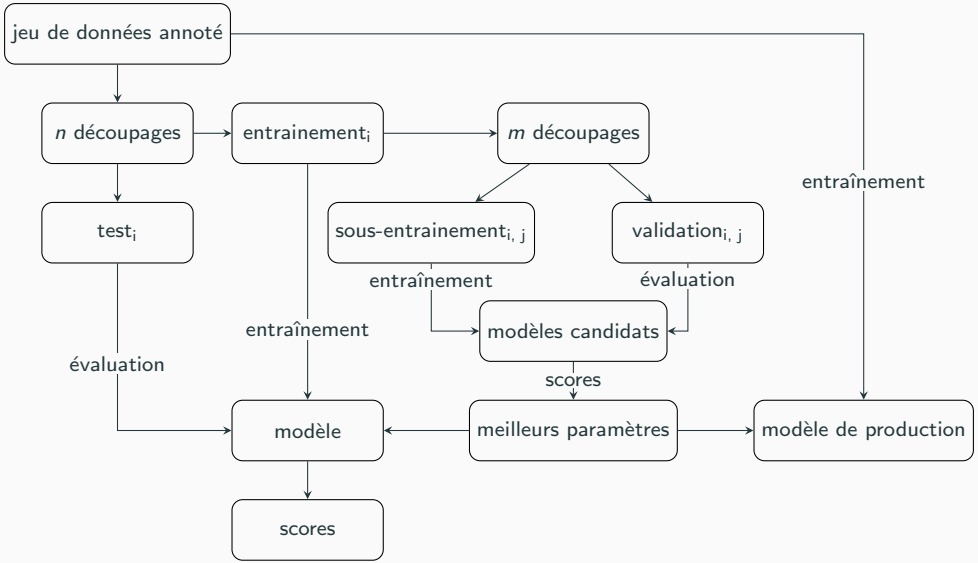
MÉTHODOLOGIE GLOBALE



MÉTHODOLOGIE GLOBALE



MÉTHODOLOGIE GLOBALE



ÉQUIVALENCES DE PROBLÈMES

CONVERSION RÉGRESSION VERS CLASSIFICATION

Remplacer la classe (binaire ou ordinale) par un score

Conversion des données d'entraînement et des prédictions

patient	diagnostic
p_1	positif
p_2	positif
p_3	négatif
prediction(p_4) $\rightarrow$	négatif

patient	diagnostic
p_1	1
p_2	1
p_3	0
prediction(p_4) $\rightarrow$	0.12

CONVERSION CLASSIFICATION BINAIRE VERS MULTI-CLASSE

Entrainement de plusieurs classificateurs binaires pour N classes

Résultat d'une prédiction : degré d'appartenance à une classe (e.g. taux de succès)

One versus One : $\frac{N(N-1)}{2}$ classificateurs

$\rightarrow$ un modèle associé à chaque paire de classes

One versus Rest : N classificateurs

$\rightarrow$ un modèle par classe

CONVERSION CLASSIFICATION BINAIRE VERS MULTI-CLASSE

Quatre classes : A, B, C et D

One versus One

- $\text{prediction}_{A,B}(x) = B$
- $\text{prediction}_{A,C}(x) = A$
- $\text{prediction}_{A,D}(x) = D$
- $\text{prediction}_{B,C}(x) = C$
- $\text{prediction}_{B,D}(x) = B$
- $\text{prediction}_{C,D}(x) = D$

- $\text{Score}(A) = 1$
- $\text{Score}(B) = 2$
- $\text{Score}(C) = 1$
- $\text{Score}(D) = 2$

One versus Rest

- $\text{prediction}_{A}(x) = 0$
- $\text{prediction}_{B}(x) = 1$
- $\text{prediction}_{C}(x) = 0$
- $\text{prediction}_{D}(x) = 0$

CONVERSION CLASSIFICATION BINAIRE VERS MULTI-CLASSE

Quatre classes : A, B, C et D

One versus One

- $\text{prediction}_{A,B}(x) = B$
- $\text{prediction}_{A,C}(x) = A$
- $\text{prediction}_{A,D}(x) = D$
- $\text{prediction}_{B,C}(x) = C$
- $\text{prediction}_{B,D}(x) = B$
- $\text{prediction}_{C,D}(x) = D$

- $\text{Score}(A) = 1$
- $\text{Score}(B) = 2 \leftarrow$ vainqueur du duel entre B et D
- $\text{Score}(C) = 1$
- $\text{Score}(D) = 2$

One versus Rest

- $\text{prediction}_{A}(x) = 0$
- $\text{prediction}_{B}(x) = 1$
- $\text{prediction}_{C}(x) = 0$
- $\text{prediction}_{D}(x) = 0$

CONVERSION CLASSIFICATION BINAIRE VERS MULTI-CLASSE

Quatre classes : A, B, C et D

One versus One

- $\text{prediction}_{A,B}(x) = B$
- $\text{prediction}_{A,C}(x) = A$
- $\text{prediction}_{A,D}(x) = D$
- $\text{prediction}_{B,C}(x) = C$
- $\text{prediction}_{B,D}(x) = B$
- $\text{prediction}_{C,D}(x) = D$

- $\text{Score}(A) = 1$
- $\text{Score}(B) = 2 \leftarrow$ vainqueur du duel entre B et D
- $\text{Score}(C) = 1$
- $\text{Score}(D) = 2$

One versus Rest

- $\text{prediction}_{A}(x) = 0$
- $\text{prediction}_{B}(x) = 1$
- $\text{prediction}_{C}(x) = 0$
- $\text{prediction}_{D}(x) = 0$

Fonctionne mieux si la prédiction donne un score (réel entre 0 et 1) et non un booléen

QUELQUES GRANDES CATÉGORIES DE MODÈLES

INTERPOLATION AU PLUS PROCHE VOISIN

Jeu d'entraînement : $f(x_i) = y_i$

Déterminer $f(x)$ peut être vu comme un problème d'interpolation

$$f(x) = f(\underset{x_i}{\operatorname{argmin}}\, d(x, x_i)) = y_i$$

Méthode non généralisatrice : se contente de mémoriser les données d'apprentissage

INTERPOLATION AUX k PLUS PROCHES VOISINS

Prédiction : combinaison des labels des k plus proche voisin

Points x_i triés par distance croissante par rapport à x : $\|x - x_i\| \leq \|x - x_{i+1}\|$

- vote majoritaire parmi les k voisins $\{x_i\}_{1 \leq i \leq k}$
- pondération par des poids $\{w_i\}_{1 \leq i \leq k}$: $f(x) = \dfrac{\sum_{i=1}^{k} w_i y_i}{\sum_{i=1}^{k} w_i}$

E.g. w_i est l'inverse de la distance :

$$w_i = \begin{cases} \dfrac{1}{\|x - x_i\|} & \text{si } x \neq x_i \\ +\infty & \text{sinon} \end{cases}$$

Méthode non généralisatrice

MODÈLES LINÉAIRES

Déterminer W (weight) et B (bias) : $Y \sim WX + B$

Variantes en fonction des conditions d'optimalité et des méthodes de calcul

MODÈLES POLYNOMIAUX

Génération de combinaisons polynomiales des composantes de x

Pour un ordre trois

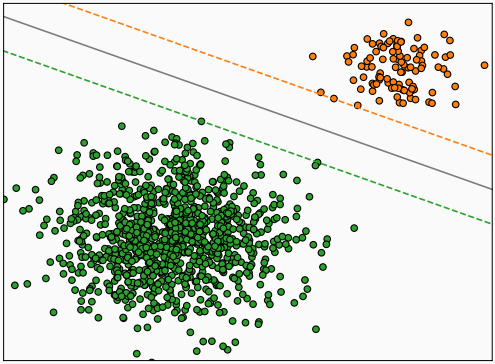
$x = [x_1, x_2, x_3]$

$$x' = [1, x_1, x_2, x_3, x_1x_2, x_1x_3, x_2x_3, x_1^2, x_2^2, x_3^2, x_1x_2x_3$$
$$x_1^2x_2, x_1^2x_3, x_1x_2^2, x_2^2x_3, x_1x_3^2, x_2x_3^2, x_1^3, x_2^3, x_3^3]$$

Utiliser un modèle linéaire sur x' revient à utiliser un modèle polynomial d'ordre 3 sur x

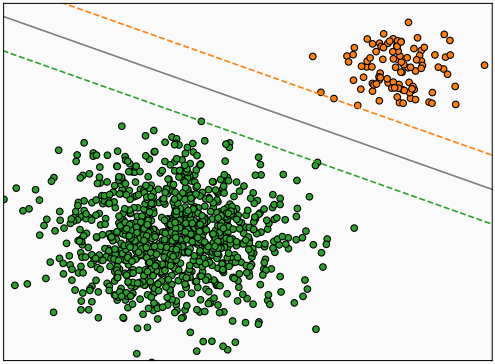
MACHINES À VECTEURS DE SUPPORT (SVM)

Calculer un hyperplan séparateur entre deux classes (maximisant la marge)



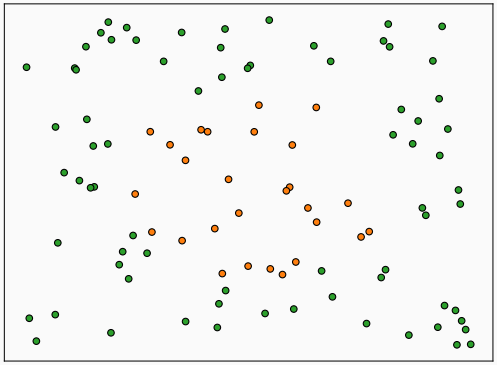
MACHINES À VECTEURS DE SUPPORT (SVM)

Calculer un hyperplan séparateur entre deux classes (maximisant la marge)

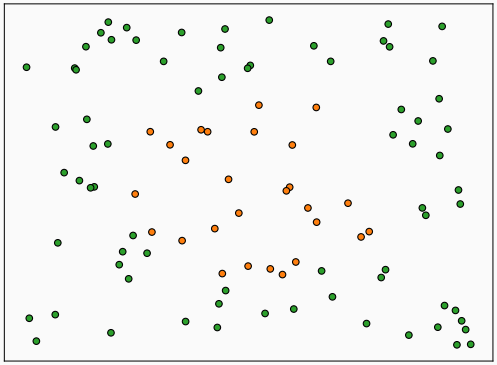


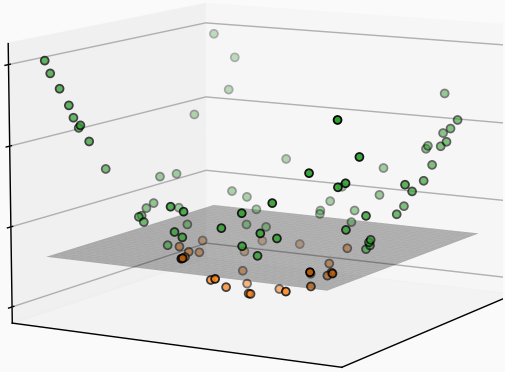
Notion de tolérance sur des points mal classés (avec une pénalité)

Données non linéairement séparables



Données non linéairement séparables

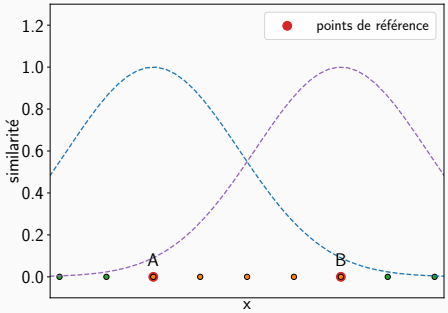




Projection dans un espace de plus grande dimension (e.g. combinaisons polynomiales)

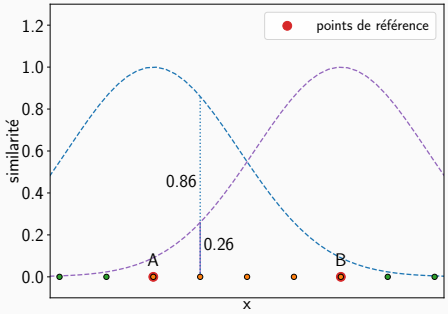
FONCTION DE BASE RADIALE GAUSSIENNE

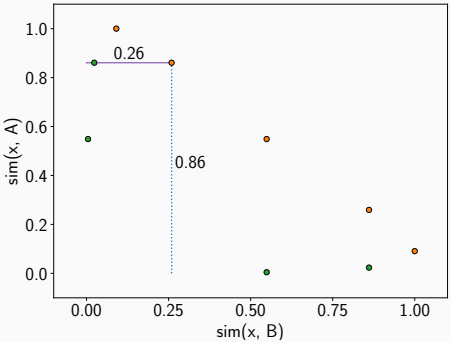
Gaussian RBF : autre projection, dans l'espace des points



FONCTION DE BASE RADIALE GAUSSIENNE

Gaussian RBF : autre projection, dans l'espace des points

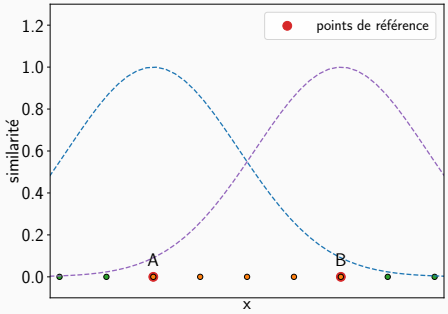


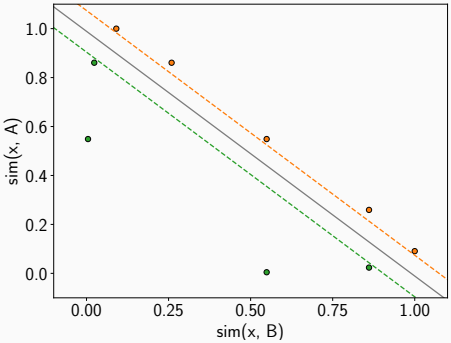


$\text{sim}(x, A) = 0.86$, $\text{sim}(x, B) = 0.26$

FONCTION DE BASE RADIALE GAUSSIENNE

Gaussian RBF : autre projection, dans l'espace des points





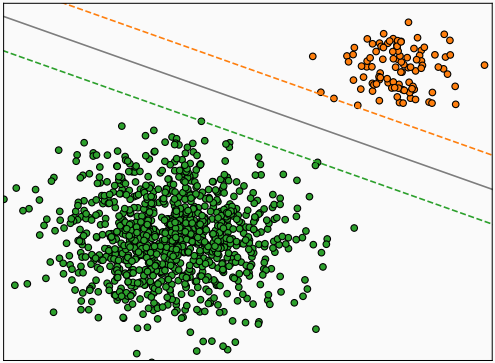
ASTUCE DU NOYAU

Problème dual d'optimisation : calcul d'un produit scalaire dans l'espace de projection (codomaine de Φ)

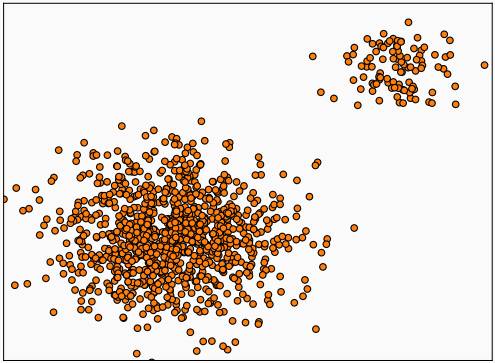
Il existe une fonction noyau k : $\langle \Phi(x), \Phi(x') \rangle = k(x, x')$

$\rightarrow$ permet de ne pas expliciter (ni calculer) la projection

SVM POUR LA REGRESSION

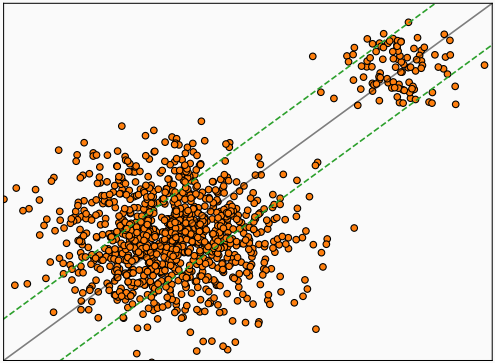


SVM POUR LA REGRESSION

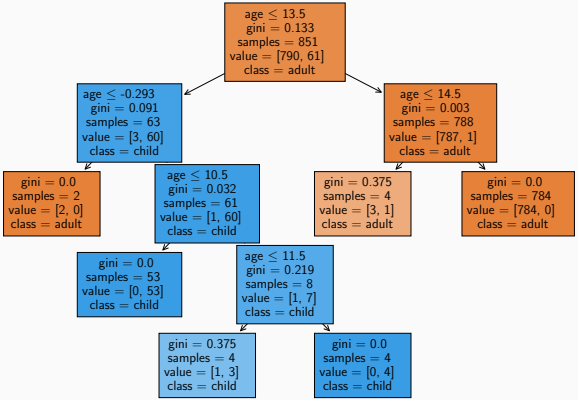


SVM POUR LA REGRESSION

Maximiser le nombre d'éléments dans la marge



Construction d'un arbre de classification ou de régression



Indice de Gini (notion d'inégalité) :

$$\text{Gini}(P) = 1 - \sum_k p_k^2$$

- P : population
- p_k : proportion de la classe k dans P

Exemples pour trois classes

- $\text{Gini}([0, 50, 0]) = 1 - \left(\left(\frac{0}{50}\right)^2 + \left(\frac{50}{50}\right)^2 + \left(\frac{0}{50}\right)^2\right) = 0$ (score parfait, nœud pur)
- $\text{Gini}([21, 12, 17]) = 1 - \left(\left(\frac{21}{50}\right)^2 + \left(\frac{12}{50}\right)^2 + \left(\frac{17}{50}\right)^2\right) = 0.65$

ALGORITHME DE CONSTRUCTION

Algorithme CART (*Classification and Regression Tree*)

Principe : scinder récursivement les nœuds de l'arbre

Objectif : maximiser la pureté des sous-ensembles gauche et droite en équilibrant leurs tailles

Identifier l'attribut k et le seuil s_k divisant la population P en P_l et P_r, qui minimise une fonction de coût, par exemple :

$$|P_l| \times \text{Gini}(P_l) + |P_r| \times \text{Gini}(P_r)$$

ARBRES DE DÉCISION

Modèles *boîte blanche* et efficaces

Pas ou peu de préparation de données nécessaire

Modèles non paramétriques : nombre de paramètres (degré de liberté) non fixé à l'avance $\rightarrow$ modèles arbitrairement complexes sensibles au sur-apprentissage et aux classes déséquilibrées

Processus de régularisation pour ajouter des contraintes sur le modèle (par exemple, taille minimale des feuilles pouvant être scindées, profondeur maximale, etc.)

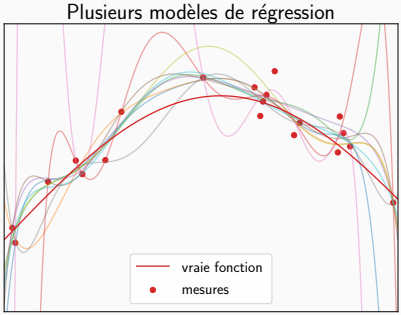
FORÊTS DE DÉCISION ET MÉTHODES ENSEMBLISTES

Entraînement indépendant de plusieurs arbres (estimateurs) et agrégation de leurs prédictions :

- jeux de données d'entraînement différents : sous ensemble d'échantillons, de caractéristiques ou des deux
- introduction d'aléa dans la construction du modèle

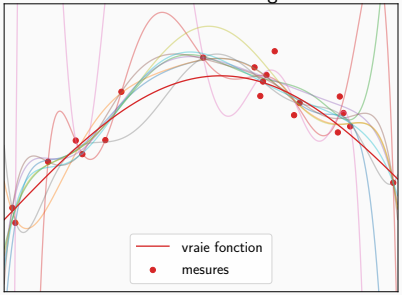
Agrégation des prédictions :

- moyenne
- médiane
- vote majoritaire

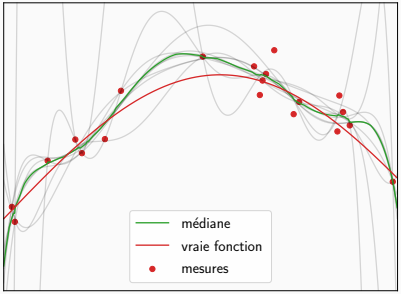

Plusieurs modèles de régression
vraie fonction
mesures

MÉTHODES ENSEMBLISTES

Plusieurs modèles de régression



Valeur médiane des modèles



Lissent les effets du sur apprentissage

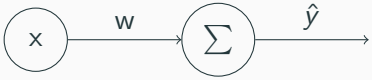
Proposent statistiquement une plus grande précision que le meilleur modèle de l'ensemble

PERCEPTRON

Modèle linéaire entraîné itérativement

Exemple pour une entrée unique x

$\hat{y} = wx + b$ avec w le poids et b le bias



DESCENTE DU GRADIENT

$$\text{err}(b, w) = \hat{y} - y = wx + b - y$$

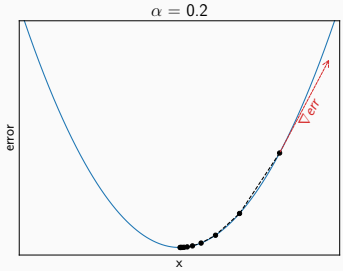
$$\frac{\partial \,\text{err}}{\partial b} = 1 \text{ et } \frac{\partial \,\text{err}}{\partial w} = x$$

Descente du gradient itérative (coefficient d'apprentissage α)

$$b_{i+1} = b_i - \alpha \,\text{err}(b_i, w_i)\frac{\partial \,\text{err}}{\partial b} = b_i - \alpha \,\text{err}(b_i, w_i)$$

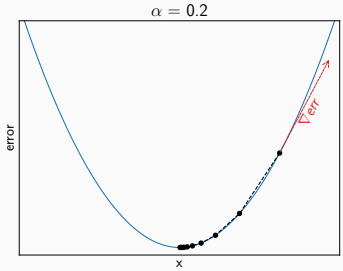
$$w_{i+1} = w_i - \alpha \,\text{err}(b_i, w_i)\frac{\partial \,\text{err}}{\partial w} = w_i - \alpha \,\text{err}(b_i, w_i)x$$

EFFET DU COEFFICIENT D'APPRENTISSAGE

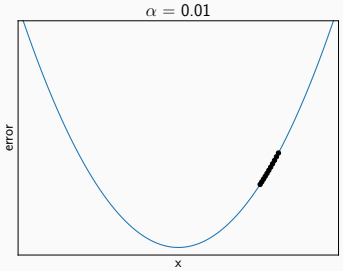


Coefficient α ajusté

EFFET DU COEFFICIENT D'APPRENTISSAGE

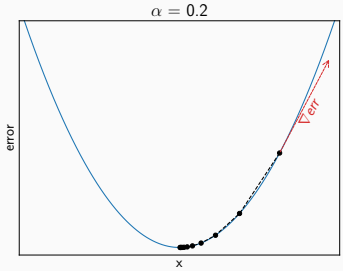


Coefficient α ajusté

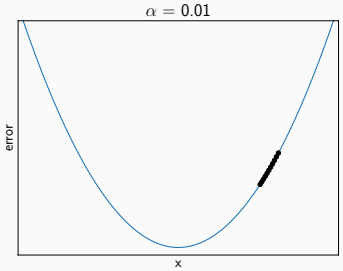


Apprentissage lent

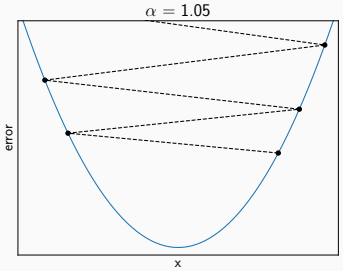
EFFET DU COEFFICIENT D'APPRENTISSAGE



Coefficient α ajusté

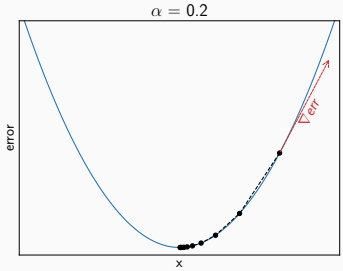


Apprentissage lent

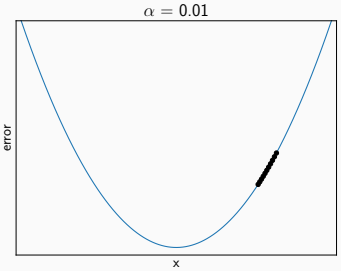


Divergence

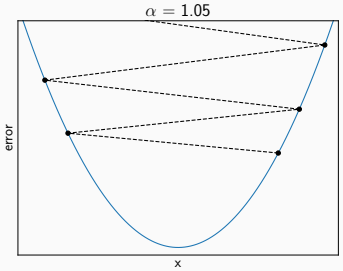
EFFET DU COEFFICIENT D'APPRENTISSAGE



Coefficient α ajusté



Apprentissage lent



Divergence

α déterminé empiriquement (généralement variable)

DESCENTE DU GRADIENT PAR LOT (BATCH)

Gradient variable en fonction de l'entrée considérée $\rightarrow$ traitement par lots de n valeurs

$$\text{err}(b, w) = \text{MSE}(b, w) = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2$$

$$\frac{\partial \text{err}}{\partial b} = \frac{2}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i) \qquad \frac{\partial \text{err}}{\partial w} = \frac{2}{n}\sum_{i=1}^{n} w(\hat{y}_i - y_i)$$

DESCENTE DU GRADIENT PAR LOT (BATCH)

Gradient variable en fonction de l'entrée considérée $\rightarrow$ traitement par lots de n valeurs

$$\text{err}(b, w) = \text{MSE}(b, w) = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2$$

$$\frac{\partial \text{err}}{\partial b} = \frac{2}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i) \qquad \frac{\partial \text{err}}{\partial w} = \frac{2}{n}\sum_{i=1}^{n} w(\hat{y}_i - y_i)$$

Contenu du lot :

- un échantillon $\rightarrow$ descente du gradient stochastique (lent, non parallélisable)
- l'intégralité des données $\rightarrow$ descente du gradient full-batch
  (lent si jeu d'entraînement volumineux)
- un sous ensemble des données $\rightarrow$ descente du gradient mini-batch

NOTION D'ÉPOQUE

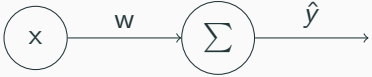
Itération : traitement d'un lot $\rightarrow$ mise à jour des poids suite au calcul d'un gradient

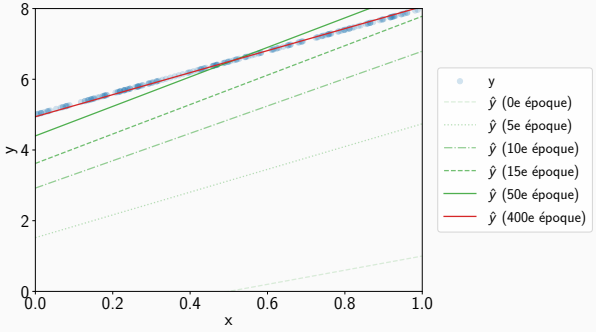
Époque (*epoch*) : ensemble d'itérations couvrant l'intégralité du jeu d'entraînement

Nombre d'époques $\rightarrow$ nombre de passes réalisées sur le jeu d'entraînement

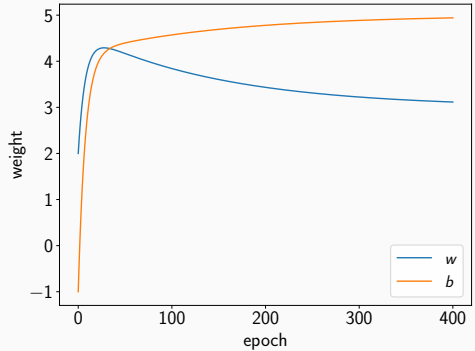
EXEMPLE DE CONVERGENCE

$f(x) = 3x + 5$



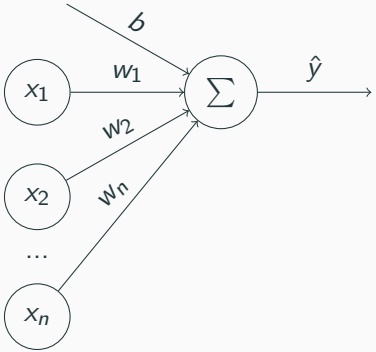


EXEMPLE DE CONVERGENCE



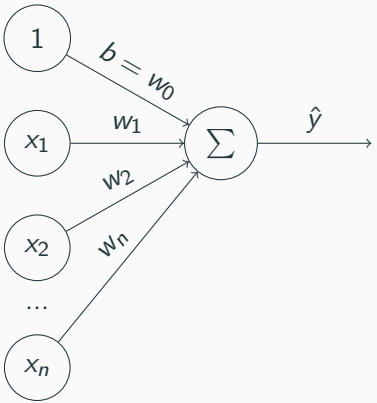
PERCEPTRON AVEC PLUSIEURS ENTRÉES

Pour m entrées $\mathbf{x}$ de dimension n



PERCEPTRON AVEC PLUSIEURS ENTRÉES

Pour m entrées $\mathbf{x}$ de dimension n



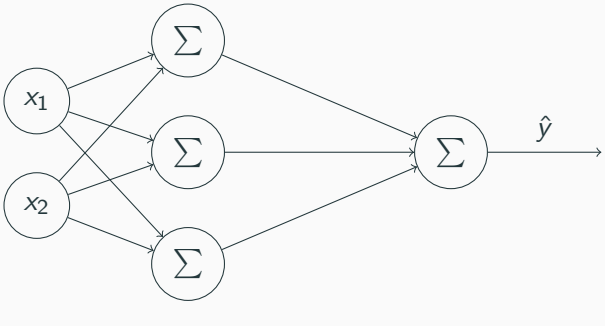
$$\hat{\mathbf{y}} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_{1,1} & x_{1,2} & \cdots & x_{1,m} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,m} \\ \cdots & \cdots & \cdots & \cdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,m} \end{pmatrix} \begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ \cdots \\ w_n \end{pmatrix} = \mathbf{X}\mathbf{w}$$

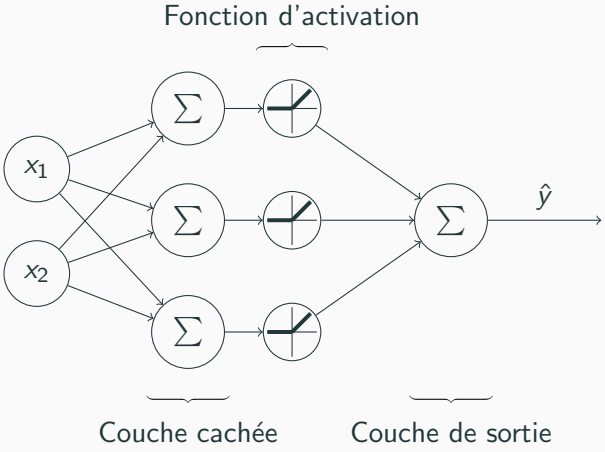
$$\mathbf{w}_{i+1} = \mathbf{w}_i - \alpha \nabla_{\mathbf{w}} \operatorname{err}(\mathbf{w})$$

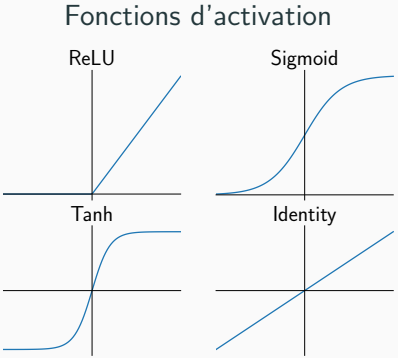
Pour $\operatorname{err}(\mathbf{w}) = \operatorname{MSE}(\mathbf{w})$:

$$\nabla_{\mathbf{w}} \operatorname{err}(\mathbf{w}) = \frac{2}{m} \mathbf{X}^\intercal (\mathbf{X}\mathbf{w} - \mathbf{y})$$

MULTILAYER PERCEPTRON




Fonction d'activation
x_1
x_2
$\hat{y}$
Couche cachée
Couche de sortie


Fonctions d'activation
ReLU
Sigmoid
Tanh
Identity

EXEMPLE DE PERCEPTRON MULTICOUCHE

$f(a, b) = \max(a, b)$

EXEMPLE DE PERCEPTRON MULTICOUCHE

$$f(a, b) = \max(a, b)$$

$$f(a, b) = \mathrm{ReLU}(a - b) + b$$

EXEMPLE DE PERCEPTRON MULTICOUCHE

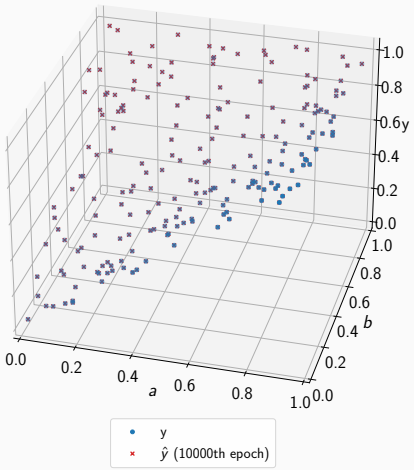
$f(a, b) = \max(a, b)$

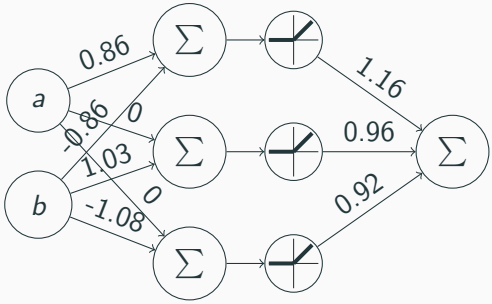
$f(a, b) = \text{ReLU}(a - b) + b$

$f(a, b) = \text{ReLU}(a - b) + \text{ReLU}(b) - \text{ReLU}(-b)$

$\rightarrow$ fonction modélisable avec une couche cachée de trois neurones

EXEMPLE DE PERCEPTRON MULTICOUCHE





$$\hat{y} \simeq 1.16\,\mathrm{ReLU}(0.86a - 0.86b) + 0.96\,\mathrm{ReLU}(1.03b) - 0.92\,\mathrm{ReLU}(-1.08b)$$

THÉORÈME UNIVERSEL D'APPROXIMATION

Avec un fonction d'activation continue non polynomiale, un réseau de neurones possédant une seule couche cachée peut approximer n'importe quelle fonction continue avec une précision arbitraire.

RÉSEAUX DE NEURONES PROFONDS

Tout réseau de neurones avec au moins une couche cachée

Grande diversité des couches et des architectures :

- couches denses : multilayer perceptron
- couches de convolution
- couches de *pooling*
- couches récurrentes
- etc.