MA431 : Mathématiques appliquées à la sécurité

Classement et apprentissage supervisé

D. Barcelo

Grenoble INP ESISAR

2022/2023

Qu'avons-nous vu jusqu'à maintenant ?

- Tests d'hypothèses
- ACP
- Regression linéaire
- Regression logistique

Qu'avons-nous vu jusqu'à maintenant ?

- Tests d'hypothèses
- ACP
- Regression linéaire
- Regression logistique

Quel est l'objectif ?

Qu'avons-nous vu jusqu'à maintenant ?

- Tests d'hypothèses
  Valider/Rejeter des hypothèses
- ACP

- Regression linéaire

- Regression logistique

Quel est l'objectif ?

Qu'avons-nous vu jusqu'à maintenant ?

- Tests d'hypothèses
  Valider/Rejeter des hypothèses
- ACP
  Réduire la taille des données, diminuer le nombre de dimensions, représentation graphique
- Regression linéaire
- Regression logistique

Quel est l'objectif ?

Qu'avons-nous vu jusqu'à maintenant ?

- Tests d'hypothèses
  Valider/Rejeter des hypothèses
- ACP
  Réduire la taille des données, diminuer le nombre de dimensions, représentation graphique
- Regression linéaire
  Relation linéaire entre variables / Prévision d'une variable quantitative
- Regression logistique

Quel est l'objectif ?

Qu'avons-nous vu jusqu'à maintenant ?

- Tests d'hypothèses
  Valider/Rejeter des hypothèses
- ACP
  Réduire la taille des données, diminuer le nombre de dimensions, représentation graphique
- Regression linéaire
  Relation linéaire entre variables / Prévision d'une variable quantitative
- Regression logistique
  Etablir un classement en fonction de variables / Prévision d'une variable binaire

Quel est l'objectif ?

Supervisé ou non supervisé

On veut obtenir une relation de la forme : $Y = f(X)$ avec
$X = (X_1, \ldots, X_p)$ les variables explicatives et Y la variable à expliquer.
L'apprentissage peut être

- **supervisé** :
  Variable Y à étudier/prévoir connue
  besoin d'une base d'apprentissage et d'une base test
- **non supervisé** :
  Variable Y à étudier inconnue

Supervisé ou non supervisé

L'apprentissage peut être

- **supervisé** :
  Y discrète : Classification.
  Y continue : Regression.
- **non supervisé** :
  Y discrète : Clustering.

Classification (fr) ou Classification (eng)

Classification automatique

Le clustering (ou *classification automatique*) permet de regrouper des individus dans des classes (clusters) non définies à priori. Il s'agit d'un apprentissage automatique non supervisé. Les classes sont déterminées au cours de l'algorithme. Elles regroupent des individus ayant des caractéristiques similaires et séparent ceux qui ont des caractéristiques différentes (on maximise l'inertie interclasse et on minimise l'inertie intraclasse).

Classification

La classification (ou *classement*) permet d'affecter des individus à des classes existantes à priori en fonction de ses caractéristiques. Il s'agit d'un apprentissage supervisé. Le résultat du classement est un algorithme permettant d'affecter chaque individu à la meilleure classe.

Qualités attendues d'un modèle

- La précision
- La robustesse
- La rapidité de calcul
- Résultats explicites et concis

Qualités attendues d'un modèle

- La précision
- La robustesse
- La rapidité de calcul
- Résultats explicites et concis

Qualités attendues d'un modèle

- La précision
  Taux d'erreur le plus bas possible. R^2 ou AUC le plus proche de 1
- La robustesse
- La rapidité de calcul
- Résultats explicites et concis

Qualités attendues d'un modèle

- La précision
  Taux d'erreur le plus bas possible. R^2 ou AUC le plus proche de 1
- La robustesse
  peu sensible aux fluctuations d'échantillonnage
- La rapidité de calcul

- Résultats explicites et concis

Qualités attendues d'un modèle

- La précision
  Taux d'erreur le plus bas possible. R^2 ou AUC le plus proche de 1
- La robustesse
  peu sensible aux fluctuations d'échantillonnage
- La rapidité de calcul
  Implémentation logicielle et puissance de calcul disponible
- Résultats explicites et concis

Qualités attendues d'un modèle

- La précision
  Taux d'erreur le plus bas possible. R^2 ou AUC le plus proche de 1
- La robustesse
  peu sensible aux fluctuations d'échantillonnage
- La rapidité de calcul
  Implémentation logicielle et puissance de calcul disponible
- Résultats explicites et concis
  Règles simples, accessibles et compréhensibles

Valisation croisée

On dispose d'un échantillon de taille n pour construire un modèle.
On a donc n réalisations des k variables explicatives et une de la variable à expliquer.

Validation croisée

- Technique pour s'assurer que les résultats trouvés sont généralisables.
- **Double validation croisée** : on sépare les données en deux parties disjointes
  une base d'apprentissage et une base test
- k **validation croisée** : on sépare les données en k parties disjointes
  On détermine k modèles avec un sous-ensemble comme base test et le complémentaire comme base d'apprentissage.
  $(5 \leqslant k \leqslant 15$ avec préférence pour $k = 10)$
- n **validation croisée** : on sépare les données en n parties disjointes
  Base d'apprentissage de taille $n - 1$ et base test de taille 1.

Base d'apprentissage et base de test

Critères à respecter lors de la validation croisée :

- Indépendance des deux bases :
- La variable Y n'est pas sous-représentée ou sur-représentée dans l'une des bases :

Base d'apprentissage et base de test

Critères à respecter lors de la validation croisée :

- Indépendance des deux bases :
  affectation aléatoire dans l'une des deux bases
- La variable Y n'est pas sous-représentée ou sur-représentée dans l'une des bases :

Base d'apprentissage et base de test

Critères à respecter lors de la validation croisée :

- Indépendance des deux bases :
  affectation aléatoire dans l'une des deux bases
- La variable Y n'est pas sous-représentée ou sur-représentée dans l'une des bases :
  Tests de comparaison d'échantillons ou d'homogénéité pour valider la partition

Surapprentissage

Surapprentissage

On parle de surapprentissage ou de surajustement (ou overfitting) lorsque :

- le modèle est trop complexe,
- le modèle correspond trop bien aux données de la base d'apprentissage,
- les prédictions du modèle sont faussées.

Surapprentissage

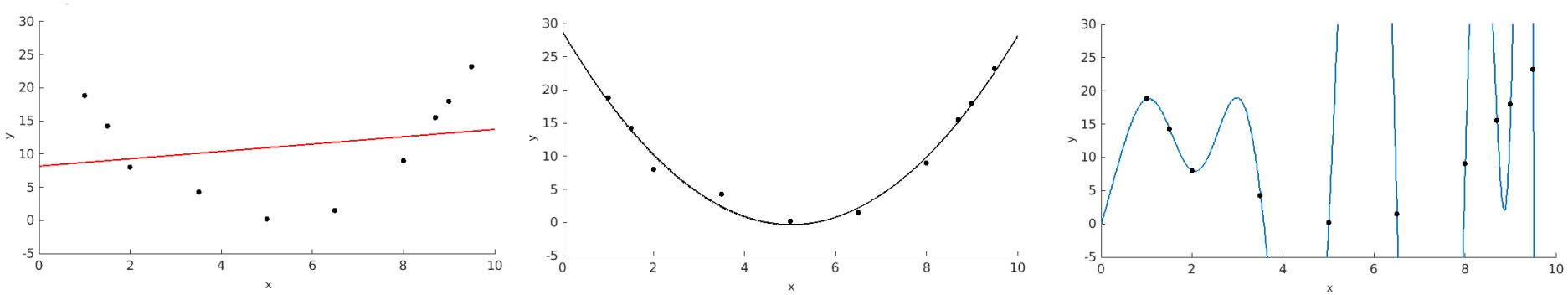


image :*Angélique Perrillat-Mercerot, Paul Dequidt,*
"Des données biologiques aux modèles et inversement", Images des Mathématiques, CNRS, 2019

Surapprentissage

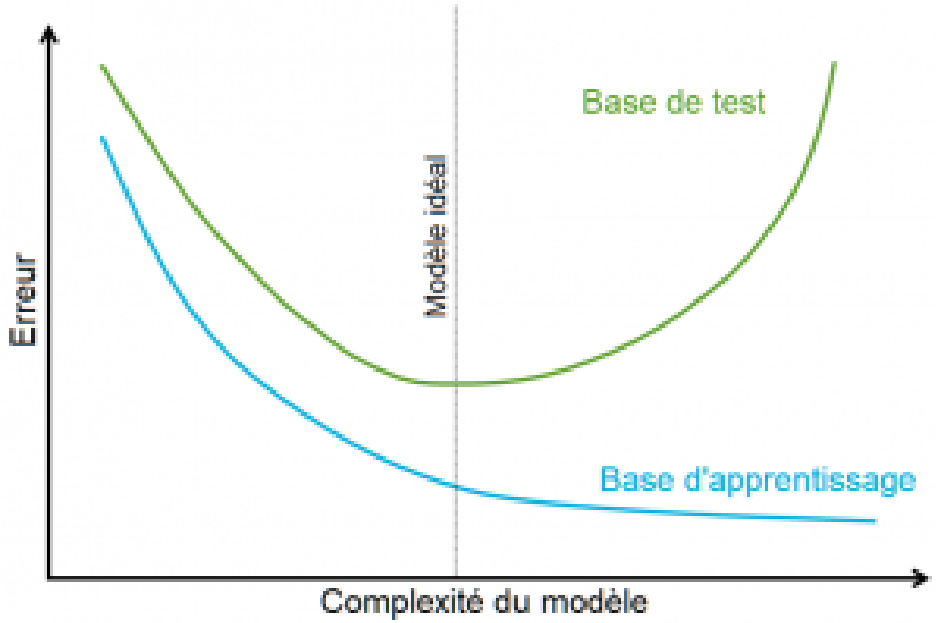


image :*Angélique Perrillat-Mercerot, Paul Dequidt,*
"Des données biologiques aux modèles et inversement", Images des Mathématiques, CNRS, 2019

Biais ou variance ?

Biais ou variance ?

- A une faible complexité du modèle correspond une faible variance du séparateur
- A un faible taux d'erreur sur l'ensemble d'apprentissage correspond un faible biais du modèle.
- Difficile de minimiser le biais et la variance.

Un outil pour quantifier les deux est l'erreur quadratique moyenne (MSE).
On a : MSE= variance + biais^2.

Biais ou variance

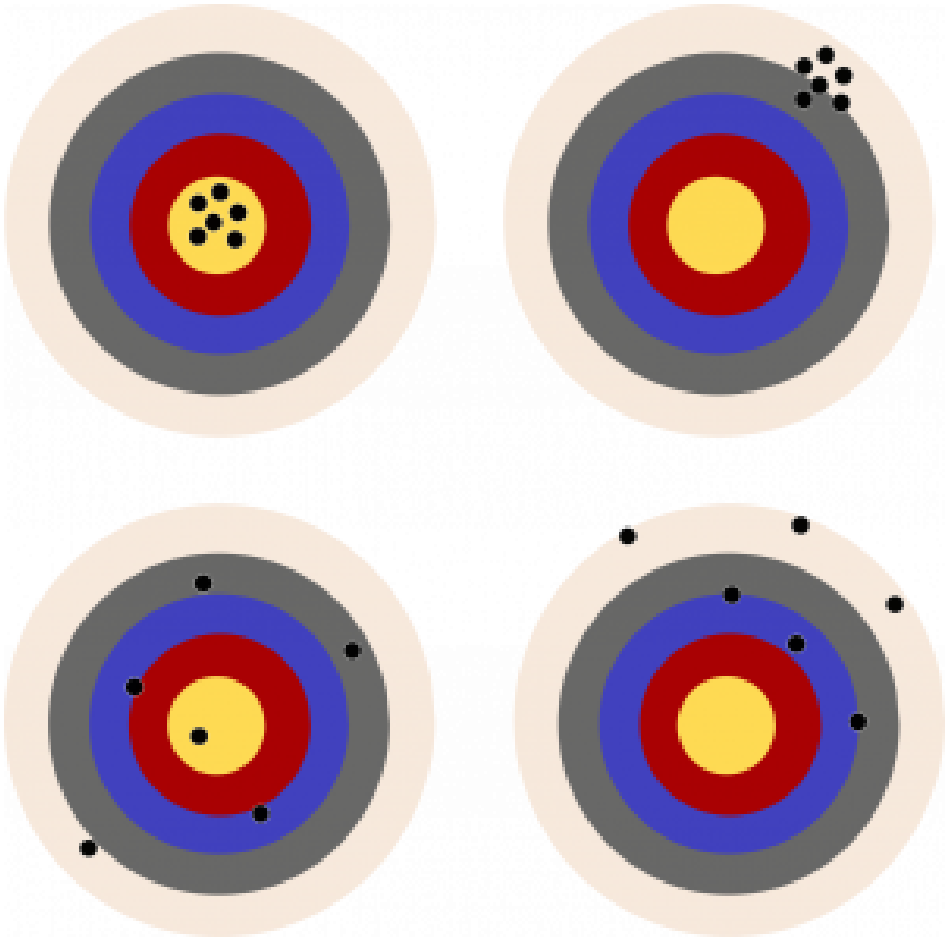


image :*Angélique Perrillat-Mercerot, Paul Dequidt,*
"Des données biologiques aux modèles et inversement", Images des Mathématiques, CNRS, 2019

Exemples de techniques d'apprentissage supervisé

- Les $k-$plus proches voisins ($K-$NN),
- les classifieurs Naïve Bayes,
- la regression linéaire (ce n'est pas du classement, quoique)
- la regression logistique,
- Support Vecteur Machines (SVM),
- etc.

Les $k-$plus proches voisins

Les $k-$plus proches voisins

- Algorithme relativement connu,
- on classe directement chaque individu par rapport aux individus déjà classés, sans construire un modèle,
- principe simple,
- utilisé pour du classement ou de la prédiction.

Les $k-$plus proches voisins

Les $k-$plus proches voisins

Configuration minimale :

- k est le seul paramètre,
- réfléchir au poids des voisins,
- choisir la bonne distance.

Algorithme $K-$NN

- On dispose d'un ensemble d'individus, caractérisés par des variables et affectés à différentes classes,
- on veut déterminer la classe d'un nouvel individu,
- on détermine les $k-$plus proches voisins déjà classés de ce nouvel individu,
- on affecte le nouvel individu à la classe majoritaire dans ses $k-$plus proches voisins.

Les $k-$plus proches voisins, cadre mathématique

Plus mathématiquement :
On observe m variables quantitatives $(X_1, \ldots, X_m)$ et une variable qualitative (Y) sur n individus. La variable qualitative peut prendre q modalités distinctes.

Les $k-$plus proches voisins

On considère $D \subset \mathbb{R}^m \times [\![1; q]\!]$.
On munit $\mathbb{R}^m$ d'une distance d. On définit $V_k(x)$ le voisinage des $k-$plus proches voisins de x au sens de la distance d.
Soit $z \in \mathbb{R}^m$. On définit $\hat{Y}(z)$ la prédiction de modalité de z par $K-$NN.

On pose alors : $\hat{Y}(z) = argmax_{c \in [\![1;q]\!]} \left\{ \sum_{(x,y) \in V_k(z)} \delta_{y,c} \right\}$.

Exemple 1

Exemple 1

Choix de k ?

- k petit : faible biais et forte variance
- k grand : fort biais et faible variance

Choix de k ?

- k : choisi afin d'obtenir le meilleur classement possible.
- k ne peut être déterminé automatiquement.
- Trouver la valeur optimale de k : entrainement sur la base d'apprentissage.
- Test de toutes les valeurs de k possibles entre 2 et n (quoique..).
- Valeur de k choisie : taux d'erreur le plus faible.
- En cas de trop grande dimension : les points sont trop éloignés les uns des autres et on doit quasiment choisir de manière aléatoire les voisins.

Choix de k ?

Inutile de tester un trop grand nombre de voisins (par exemple n).
Pourquoi ?

Choix de k ?

Inutile de tester un trop grand nombre de voisins (par exemple n).
Pourquoi ?
Car si le nombre de voisins est trop important, les distances sont inutiles.
Il suffirait d'affecter la classe majoritaire au nouvel individu.

Distances

L'algorithme repose sur un calcul de distance. Soit $\vec{x} \in \mathbb{R}^n$ et $\vec{y} \in \mathbb{R}^n$.

- Distance euclidienne : $d(\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$.
- Distance Manhattan : $d(\vec{x}, \vec{y}) = \sum_{i=1}^{n} |x_i - y_i|$.
- Distance de Mahalanobis : $d(X; Y) = \sqrt{(X - Y)^t V (X - Y)}$,
  V matrice de variance-covariance des variables observées.
- la mesure cosinus : $d(\vec{x}, \vec{y}) = 1 - \frac{\vec{x}.\vec{y}}{\|\vec{x}\| \, \|\vec{y}\|}$.
- la distance de Hamming : $d(\vec{x}, \vec{y}) = \sum_{i=1}^{n} \mathbb{1}_{\neq}(x_i - y_i)$.

Distances

L'algorithme repose sur un calcul de distance. Soit $\vec{x} \in \mathbb{R}^n$ et $\vec{y} \in \mathbb{R}^n$.

- Distance euclidienne : $d(\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$.

  Simple, classique, rapide à calculer
- Distance Manhattan : $d(\vec{x}, \vec{y}) = \sum_{i=1}^{n} |x_i - y_i|$.
- Distance de Mahalanobis : $d(X; Y) = \sqrt{(X - Y)^t V (X - Y)}$,
  V matrice de variance-covariance des variables observées.
- la mesure cosinus : $d(\vec{x}, \vec{y}) = 1 - \frac{\vec{x}.\vec{y}}{\|\vec{x}\| \, \|\vec{y}\|}$.
- la distance de Hamming : $d(\vec{x}, \vec{y}) = \sum_{i=1}^{n} \mathbb{1}_{\neq}(x_i - y_i)$.

Distances

L'algorithme repose sur un calcul de distance. Soit $\vec{x} \in \mathbb{R}^n$ et $\vec{y} \in \mathbb{R}^n$.

- Distance euclidienne : $d(\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$.

  Simple, classique, rapide à calculer
- Distance Manhattan : $d(\vec{x}, \vec{y}) = \sum_{i=1}^{n} |x_i - y_i|$.
- Distance de Mahalanobis : $d(X; Y) = \sqrt{(X - Y)^t V (X - Y)}$,
  V matrice de variance-covariance des variables observées.
  variances et covariances, poids faible aux composantes dispersées
- la mesure cosinus : $d(\vec{x}, \vec{y}) = 1 - \frac{\vec{x}.\vec{y}}{\|\vec{x}\| \, \|\vec{y}\|}$.
- la distance de Hamming : $d(\vec{x}, \vec{y}) = \sum_{i=1}^{n} \mathbb{1}_{\neq}(x_i - y_i)$.

Distances

L'algorithme repose sur un calcul de distance. Soit $\vec{x} \in \mathbb{R}^n$ et $\vec{y} \in \mathbb{R}^n$.

- Distance euclidienne : $d(\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$.

  Simple, classique, rapide à calculer
- Distance Manhattan : $d(\vec{x}, \vec{y}) = \sum_{i=1}^{n} |x_i - y_i|$.
- Distance de Mahalanobis : $d(X; Y) = \sqrt{(X - Y)^t V (X - Y)}$,
  V matrice de variance-covariance des variables observées.
  variances et covariances, poids faible aux composantes dispersées
- la mesure cosinus : $d(\vec{x}, \vec{y}) = 1 - \frac{\vec{x}.\vec{y}}{\|\vec{x}\| \, \|\vec{y}\|}$.

  mesure la ressemblance entre des documents (textuels, voire images)
- la distance de Hamming : $d(\vec{x}, \vec{y}) = \sum_{i=1}^{n} \mathbb{1}_{\neq}(x_i - y_i)$.

Distances

L'algorithme repose sur un calcul de distance. Soit $\vec{x} \in \mathbb{R}^n$ et $\vec{y} \in \mathbb{R}^n$.

- Distance euclidienne : $d(\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$.

  Simple, classique, rapide à calculer
- Distance Manhattan : $d(\vec{x}, \vec{y}) = \sum_{i=1}^{n}|x_i - y_i|$.
- Distance de Mahalanobis : $d(X; Y) = \sqrt{(X-Y)^t V (X-Y)}$,
  V matrice de variance-covariance des variables observées.
  variances et covariances, poids faible aux composantes dispersées
- la mesure cosinus : $d(\vec{x}, \vec{y}) = 1 - \frac{\vec{x}.\vec{y}}{\|\vec{x}\| \, \|\vec{y}\|}$.

  mesure la ressemblance entre des documents (textuels, voire images)
- la distance de Hamming : $d(\vec{x}, \vec{y}) = \sum_{i=1}^{n} \mathbb{1}(x_i - y_i)$.

  pour des chaines de données

Avantages du $K-$NN

- Il ne fait pas appel à un modèle statistique, juste aux données d'apprentissage. C'est une méthode non paramétrique. On ne cherche pas à déterminer une relation entre $(X_i)_{i\in[\![1;m]\!]}$ et Y.
- Simple à comprendre et à appliquer.
- Il n'y a presque pas de surapprentissage.
- Si on choisit k assez grand, on peut lisser le "bruit" des données d'apprentissage.
- On peut le coupler à une technique de réduction de taille des données.

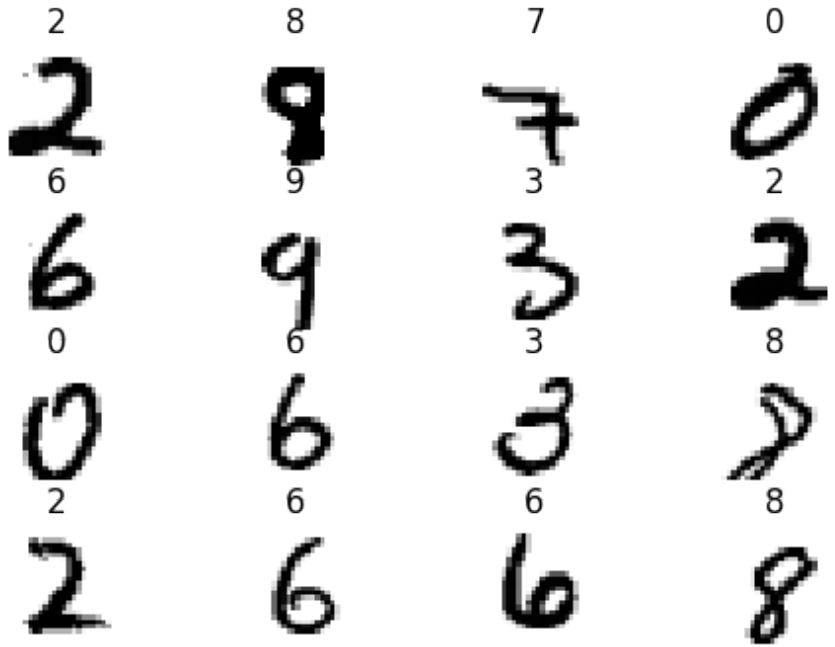
Inconvénients du $K-$NN

- Moins utilisé que les méthodes avec construction de modèle sur la base d'apprentissage.
- Nécessite une grande capacité de stockage.
- Nécessite une grande capacité de calculs.

Applications du $K-$NN

- Compléter les données manquantes dans les bases de données.
- Technique très utilisée en reconnaissance de formes.
- Reconnaissance de Spams.
- L'algorithme peut être utilisé pour détecter et mettre à l'écart les individus "atypiques".

Reconnaissance de forme



Exemple 2

Exemple 2