Généralités sur les méthodes de classification

Toutes les méthodes de prévision, et plus spécifiquement de classification, suivent les mêmes règles, le même processus de construction et de validation. Nous verrons quelques unes des ces règles concernant la préparation des données et la validation du modèle:

- Scaling des variables numériques
- Binarisation (one-hot encoding) des variables catégorielles
- Le calcul de l'erreur
- La validation croisée (base d'apprentissage et base test)
- La détection du sur-apprentissage

Apprentissage supervisé

L'*apprentissage supervisé* permet de modéliser une variable à expliquer (cible) Y en fonction de variables explicatives $X_1,\dots,X_p$

$$Y=f(X_1,\dots,X_p)+\varepsilon$$ où ε est une erreur.

Quand le modèle *f* est construit on peut s'en servir pour ***prédire*** de nouvelles valeurs/classes de Y connaissant les valeurs des variables explicatives.

Y quantitative

Y qualitative

Méthodes de régression

Cible	Explicatives
Prix d'un appartement	Superficie Standing Isolation
Concentration d'ozone	Température vitesse du vent Trafic routier Jour
Espérance de vie	Taux d'illettrisme PIB …

Méthodes de classification (supervisée)

Cible	Explicatives
Genre	Poids Taille Poids
Présence d'une maladie	Age Tabac
Fiabilité d'un client	Revenu Situation maritale Secteur d'activité …

Remarque: Lorsqu'aucune variable cible n'est identifiée dans une étude, on parle d'apprentissage non supervisé.
Les taches sont alors diverses : description des données, clustering, règles d'association (cf. ING1)

Data pre-processing

Scaling des variables numériques

	Pop. (*1000)	Nat. Rate (/1000)	Life exp.	Nb. children
Argentina	41050	16.87	75.87	2.19
Armenia	3099	15.47	74.44	1.77
Australia	21731	12.56	81.99	1.85
Austria	8407	9.01	80.55	1.40
…	…	…	…	…

La distance entre les pays Argentine et Armenie est quasiment la même si on implique toutes les variables ou uniquement la variable Population:

Pop. Nat. rate Life exp. Nb. Children

$(41050-3099)^2+(16.87-15.47)^2+(75.87-74.44)^2+(2.19-1.77)^2=1440278405$

Pop.

$(41050-3099)^2=1440278401$

Toute l'information est contenue dans les variables Natality rate, Life expectancy et Number of children est perdue car elles ont un ordre de grandeur trop petit comparé à la variable Population

⇒ Scaling des variables

	Pop. (*1000)	Nat. Rate (/1000)	Life exp.	Nb. children
Argentina	41050	16.87	75.87	2.19
Armenia	3099	15.47	74.44	1.77
Australia	21731	12.56	81.99	1.85
Austria	8407	9.01	80.55	1.40
…	…	…	…	…
$\bar{x}$	18571.75	13.48	78.21	1.80
s	16911.34	3.48	3.63	0.32

Centrer et réduire

$$\frac{x_i - \bar{x}}{s}$$

	Pop. (*1000)	Nat. Rate (/1000)	Life exp.	Nb. children
Argentina	1.33	0.98	-0.65	1.19
Armenia	-0.91	0.57	-1.04	-0.10
Australia	0.19	-0.26	1.04	0.15
Austria	-0.60	-1.28	0.64	-1.24
…	…	…	…	…
$\bar{x}$	0	0	0	0
s	1	1	1	1

Data pre-processing

Encodage des variables catégorielles

Les variables catégorielles ne sont pas numériques, Il est donc impossible de calculer des distances, des moyennes des écart-types,…, nécessaires à la plupart des algorithme d'apprentissage. Pour palier ce problème, il faut coder les variables catégorielles en les binarisant à l'aide de fonctions indicatrices.

	Continent
Argentina	AMERICA
Armenia	ASIA
Australia	OCEANIA
Austria	EUROPE
…	…

Binarisation →

	AMERICA	ASIA	EUROPE	OCEANIA
Argentina	1	0	0	0
Armenia	0	1	0	0
Australia	0	0	0	1
Austria	0	0	1	0
…	…	…	…	…

- La binarisation augmente la dimension du jeu de données. Ici la variable continent (une dimension) engendre 4 fonctions indicatrices (4 dimensions).

  N.B. Certaines librairies (Python ou R) ne tiendront pas compte de la dernière indicatrice car elle peut se retrouver à partir des autres (redondance d'information)

- Il n'est pas nécessaire de centrer et réduire les variables binaires.

- Attention à bien identifier et binariser les variables catégorielles dont les modalités sont codées avec des nombres. Sinon, l'algorithme les traitera comme des variables quantitatives, ce qui n'a aucun sens.

	Continent
Argentina	1
Armenia	2
Australia	4
…	…
Mean	2.5
Variance	1.25

Non sens

Base d'apprentissage

Pour construire le modèle *f*, il est nécessaire d'avoir une base d'apprentissage, c-à-d plusieurs observations pour lesquelles on connait les valeurs des variables explicatives mais surtout la valeur de la variable cible.

Sepal length	Sepal width	Petal length	Petal width	Species
6.3	3.3	6	2.5	virginica
5.8	2.7	5.1	1.9	virginica
7	3.2	4.7	1.4	versicolor
5.1	3.5	1.4	0.2	setosa
4.9	3	1.4	0.2	setosa
6.9	3.1	4.9	1.5	versicolor
5.5	2.3	4	1.3	versicolor
6.3	2.9	5.6	1.8	virginica
5.7	**2.9**	**4.8**	**0.9**	**???**

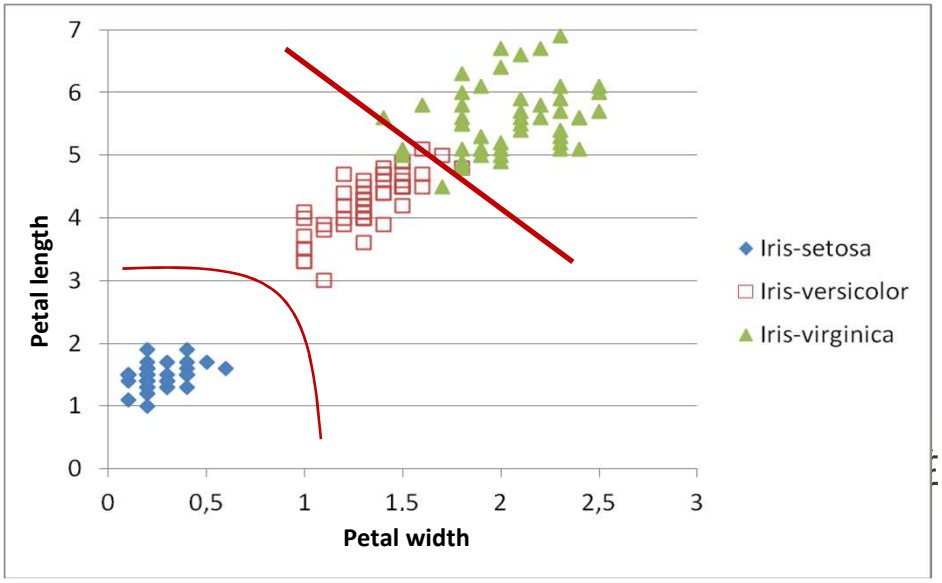
Training dataset

Nouvel iris

http://archive.ics.uci.edu/ml/datasets/Iris

⇒ Ce qui pose le problème de l'étiquetage des données qui doit être fait en amont.

En classification, le modèle correspond aux frontiers qui séparent les classes.



Performance du modèle

La matrice de confusion

Une fois le modèle construit, nous devons vérifier qu'il donne bien les résultats attendus. Pour chaque observations, on compare la valeur de la variable cible observée sur l'échantillon et celle prédite par le modèle.

Sepal length	Sepal width	Petal length	Petal width	Species		Predicted value
6.3	3.3	6	2.5	virginica		virginica
5.8	2.7	5.1	1.9	virginica	f	versicolor
7	3.2	4.7	1.4	versicolor		versicolor
5.1	3.5	1.4	0.2	setosa		setosa
4.9	3	1.4	0.2	setosa		setosa
6.9	3.1	4.9	1.5	versicolor		versicolor
5.5	2.3	4	1.3	versicolor		virginica
6.3	2.9	5.6	1.8	virginica		virginica

Erreurs du modèle

On construit ensuite le tableau de contingence entre les vraies valeurs et les valeurs prédites de chacune des classes de la variable cible. Ce tableau s'appelle la *matrice de confusion*,

		Predicted value		
		setosa	versicolor	virginica
True value	setosa	2	0	0
	versicolor	0	2	1
	virginica	0	1	2

Sur la diagonale, on trouve le nombre d'observations bien classées par le modèle. Le taux de bien classés est donc égal à la trace de la matrice divisée par le nombre total d'observations. Le taux de mal classés est égal à 1 moins le taux de bien classés.

Performance du modèle

Métriques de performance

		Classes prédites			
		C1	C2	…	Ck
Vraies classes	C1				
	C2				
	…				
	Ck				

Matrice de confusion

Performance globale

$$\frac{Nombre\ d'observations\ bien\ classées}{Nombre\ total\ d'observations}$$

Performance par classe

$$\frac{Nombre\ d'observations\ de\ la\ classe\ C_k\ bien\ classées}{Nombre\ total\ d'observations\ de C_k}$$

Avec l'exemple precedent, on a:

$$Performance\ globale = \frac{6}{8} = 75\%$$

et

		Valeur prédite			
		setosa	versicolor	virginica	Accuracy
Vraie valeur	setosa	2	0	0	100%
	versicolor	0	2	1	66.6%
	virginica	0	1	2	66.6%

75% des onservations sont bien classées.
100% des iris setosa sont bien classés
66.6% des iris versicolor et virginica sont bien classés

Performance du modèle

Cas de la classification binaire

Dans le cas spécial d'une classification binaire (quand la variable cible a 2 modalités: positive = 1, négative = 0), on calcule différents indicateurs sur la matrice de confusion suivante.

Valeur prédite

Vraie valeur	positive	negative
positive	True Positive (TP)	False Negative (FN)
negative	False Positive (FP)	True Negative (TN)

True Positive Rate : $TPR = \frac{TP}{(TP+FN)}$

True Negative Rate : $TNR = \frac{TN}{(FP+TN)}$

Positive Predicted value : $PPV = \frac{TP}{(TP+FP)}$

On peut calculer des ratios par colonne:

- Le true positive rate (TPR) est aussi appelé *sensitivity* ou *recall*. C'est la proportion des données de la classe positive pour lesquelles la prédiction est correcte.
- Le true negative rate (TNR) est appelé *specificity*. C'est la proportion des données de la classe négative pour lesquelles la prédiction est correcte.

Ou bien par ligne:

- Le positive predictive value (PPV) est appelé *precision*. C'est la proportion des données prédites positives pour lesquelles la classe est correcte.

Performance du modèle

Cas des classes déséquilibrées

Quand la distribution des classes de la variable cible est déséquilibrée, la mesure de performance globale n'est pas une bonne métrique pour évaluer le modèle.

Cas d'un diagnostic médical sur une base constituée de 5 personnes malades et 995 saines. Si le classifieur considère les 1000 personnes comme étant saines alors le taux d'erreur global est 0,5%. Autrement dit, une très bonne performance alors qu'il n'a détecté aucune des personnes malades.

Pour palier ce problème, on calcule l'erreur pondérée,

$$\frac{1}{K}\sum_{k=1}^{K}\frac{Nombre\ d'observations\ de\ C_k\ bien\ classées}{Nombre\ total\ d'observations\ de C_k}$$

où K est le nombre de classes de la variables cible.

Dans l'exemple precedent, l'erreur pondérée est 50% : $\frac{1}{2}\left(\frac{995}{995}+\frac{0}{5}\right)=\frac{1}{2}$.

Dans le cas de la classification binaire, on en déduit le F1_score

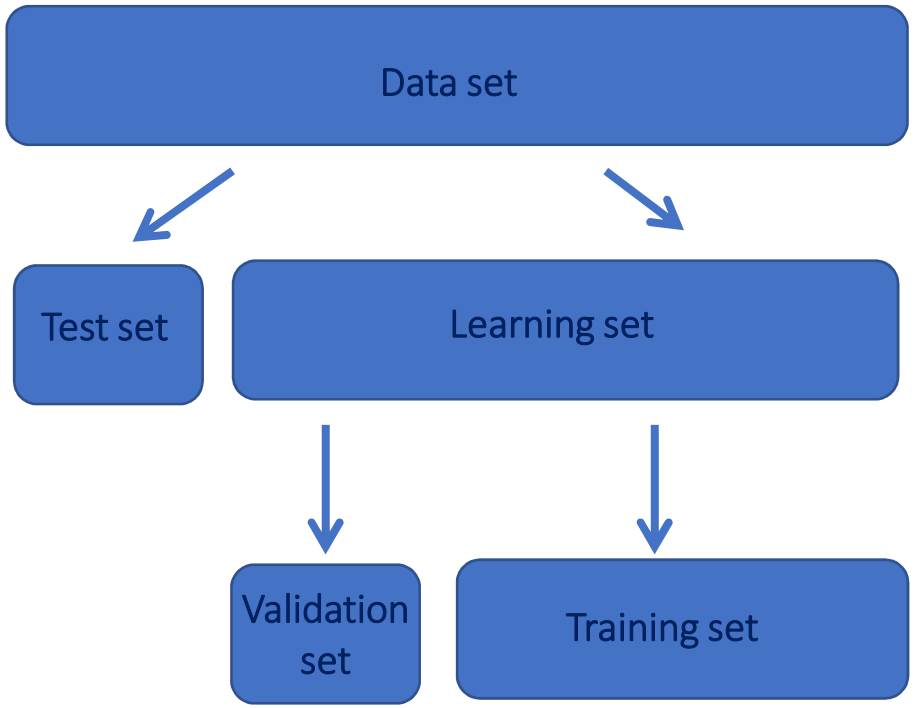
$$F1_{score}=\frac{2}{\frac{1}{Precision}+\frac{1}{Recall}}=\frac{TP}{TP+\frac{1}{2}(FN+FP)}$$

Avec l'exemple precedent, F1_score=0.

Training, validation et test

Le jeu de données se divise en trois:

- La ***base d'apprentissage*** est utilisée pour ajuster les paramètres du modèle (les poids d'un réseau de neurones, les coefficients d'une régression linéaire, les tests de partage dans un arbre de décision, …).
- La ***base de validation*** est utilisée pour définir les hyperparamètres d'un modèle (le nombre de neurones, le degré d'un polynôme, la profondeur d'un arbre,…).
- La ***base de test*** est utilisée pour mesurer la performance d'un modèle sur des données qui n'ont encore jamais été utilisées dans les deux étapes précédentes.

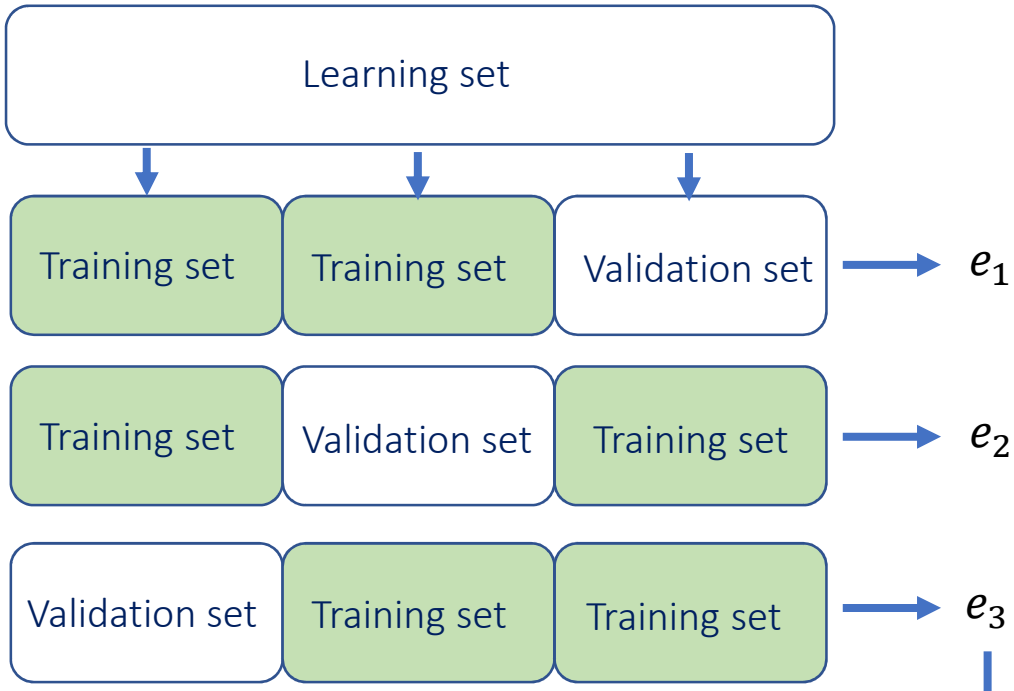


Remarque: Parfois le jeu de données et uniquement divisé en deux et les base de validation et de test sont confondues.

Erreur d'ajustement *vs* erreur de prédiction

On distingue deux types d'erreur:

- *l'erreur d'ajustement* : calculée sur la base d'apprentissage. Une petite erreur d'ajustement signifie que le modèle reproduit bien les données connues.
- *l'erreur de prévision* : calculée sur de nouvelles données. Une petit erreur de prévision signifie que le modèle est capable de prédire de nouvelles valeurs



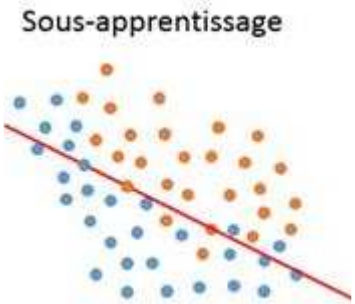
$$e = \frac{e_1 + e_2 + e_3}{3}$$

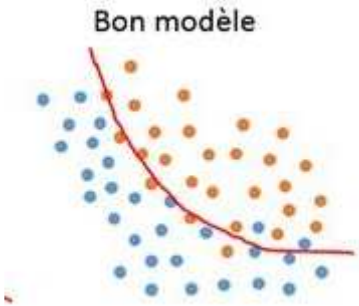
La *validation croisée* consiste à :

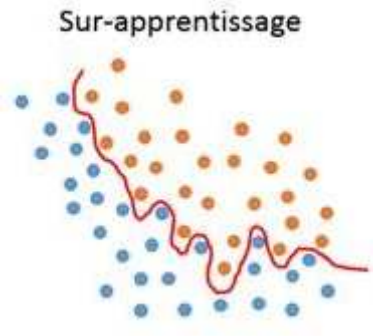
- Diviser la base d'apprentissage en p sous-ensembles
- Pour chaque sous-ensemble i,
  - Apprendre un modèle sur les (p-1) sous-ensembles restant
  - À l'aide du modèle, prédire les valeurs du sous ensemble i
  - Comparer les valeurs prédites ou vraies valeurs
- Calculer un score de prévision

Le sur-apprentissage (overfitting)

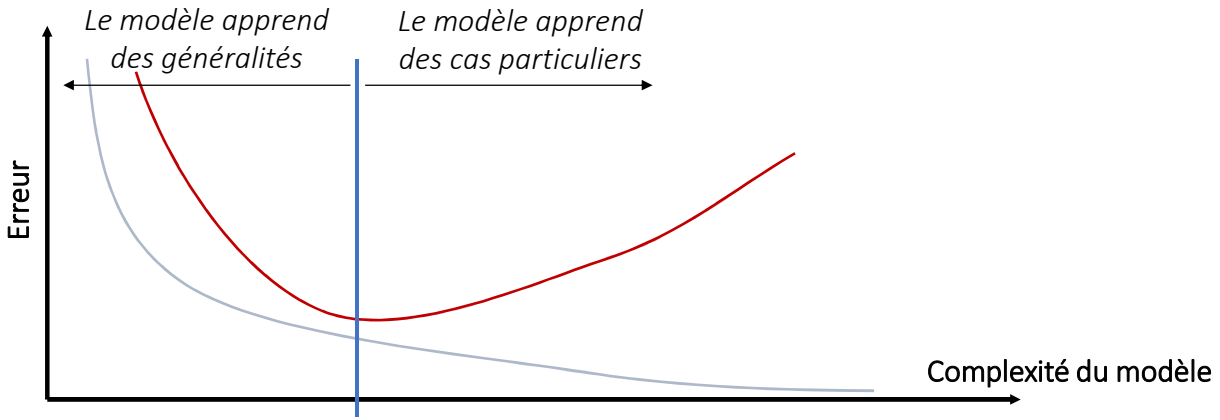
Dans la phase d'apprentissage certains modèles ont tendance à se complexifier pour ajustement au plus près les données d'apprentissage. On parle alors de *sur-ajustement* ou *sur-apprentissage*. Le modèle aura alors une très faible capacité de généralisation. Cela signifie que le modèle va reproduire parfaitement les données qui ont servies à le construire mais sera incapable de prédire de nouveaux exemples.







La détection du sur-apprentissage se fait en comparant l'évolution des erreurs d'apprentissage et de prévision en fonction de la complexité du modèle (profondeur d'un arbre, nombre de neurones…).



Méthodes de classification

Objectif : définir une méthode d'affectation d'une $(n+1)^{ème}$ observation dans une des classes C_1, …, C_k connaissant les valeurs de X_1, …, X_p pour cette observation.

Trois approches possibles :

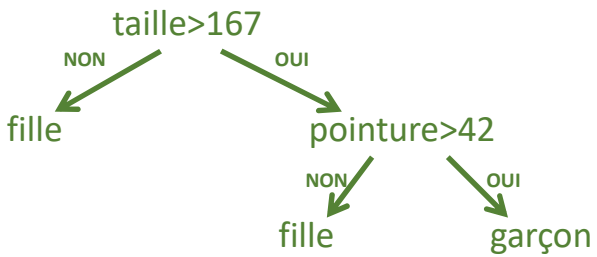
➢ On estime les probabilités conditionnelles : $P(Y=y_i|X=x)$, i=1,…k, et on choisit la classe la plus probable (*Naive Bayes*)

$P(Y=fille|taille=165,poids=55,pointure=37)=0,8$
$P(Y=garçon|taille=165,poids=55,pointure=37)=0,2$
$\Rightarrow$ **Y=fille**

➢ On applique un fonction de seuillage au modèle $f(X_1,…,X_p)$. Par exemple Y=0 si $f(X_1,…,X_p)>0$ et Y=1 sinon (*réseaux de neurones*)

$f(taille,poids,pointure)=-taille+poids+2\times pointure$
$f(165,55,37)=-36<0 \Rightarrow$ Y=fille

➢ On procède de façon itérative en séparant l'ensemble d'apprentissage variable par variable (*arbre de décision, forêt aléatoire*)



Remarque: La liste des méthodes mentionnées (Naive Bayes, réseaux de neurones, …), n'est pas exhaustive. Il s'agit de la liste des méthodes que nous détaillerons dans les séances à venir.

Questions?

N.B.

- Faire attention aux classes déséquilibrées
- Faire attention à la prise en compte des variables catégorielles codées
- Centrer et réduire les variables quantitatives suivant la méthode
- Faire attention aux outliers suivant la méthode
- Réduire la dimension si besoin (ACP)