Classification supervisée

Méthode des k plus proches voisins

François LE GAC
TPE Réalisé dans le cadre du M1 ISIFAR
2018 - 2019

université PARIS DIDEROT PARIS 7

Introduction et cadre théorique

La classification supervisée consiste à prédire à partir d'observations une quantité discrète, souvent binaire, telle que « blanc ou noir », « 0 ou 1 », « malade ou sain », « vrai ou faux ».

Plus formellement, il s'agit d'attribuer à un vecteur x $\in R^d$ une classe ou étiquette y. Le but de la classification supervisée est de construire une fonction

g : $R^d \rightarrow$ {1, . . . , M}, qui représente la prédiction de y sachant x.

Cette fonction est appelée **classificateur.**

Soit (X, Y) ∈ R^d × {1, . . . , M} un couple de variables aléatoires. Le classificateur commet une erreur si g(X) ≠ Y et sa probabilité d'erreur L est donnée par

$$L(g) = P(g(X) \neq Y)$$

Le meilleur classificateur possible est défini par

$$g* = \underset{R^d \times \{1,\ldots,M\}}{argmin} P(g(X) \neq Y)$$

Ainsi, g* , appelé le **classificateur de Bayes**, dépend de la distribution de (X, Y). On note L*= L(g*) l'erreur associée.

Considérons le cas particulier de la classification binaire. Le classificateur de Bayes est donné par

$$g*(x) = \begin{cases} 1 & si \qquad \eta(x) > 1/2 \\ 0 & \text{sinon.} \end{cases}$$

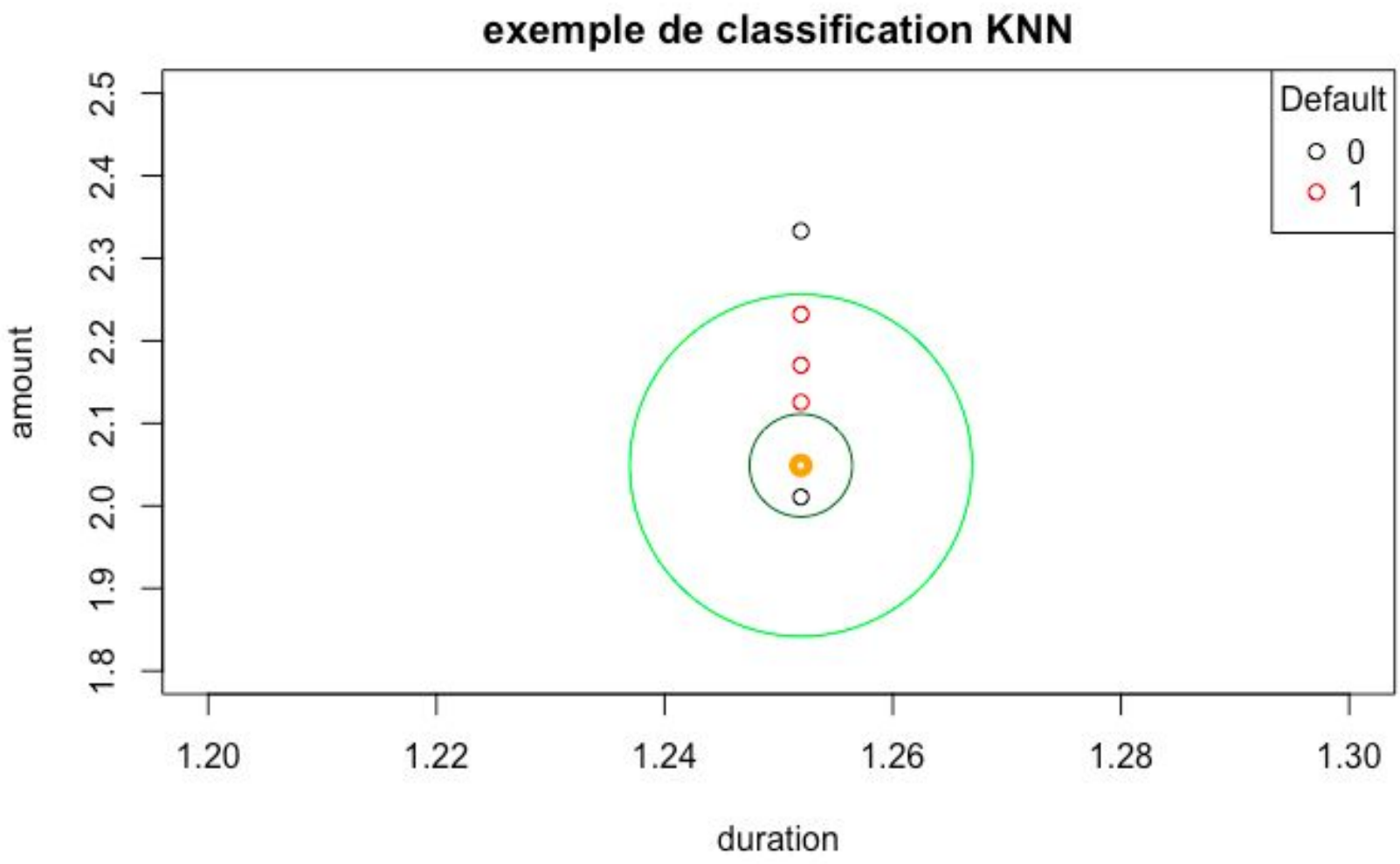
où $\eta(x) = P(Y = 1|X = x) = \mathbb{E}[Y|X = x]$

En pratique, la distribution de (X, Y) est inconnue, de sorte que g⋆ ne peut être calculé. L'objectif sera de construire un classificateur gn basé sur un échantillon (Xi , Yi), i = 1, . . . , n, qui s'en approche le plus possible.

Le classificateur g_n peut être basé sur la **règle des k plus proches voisins**. Cette règle consiste à affecter à une nouvelle observation x l'étiquette majoritaire parmi ses « voisins ». On considère les k voisins les plus proches de x, par exemple au sens de la distance euclidienne.

En définissant les poids w_{ni} par w_{ni} = 1/k si Xi fait partie des k plus proches voisins de x et w_{ni} = 0 sinon, le classificateur des plus proches voisins s'écrit

$$g_n(x) = \begin{cases} 1 & si \quad \sum_{i=1}^{n} w_{ni} 1_{\{Y_i=1\}} \geq \sum_{i=1}^{n} w_{ni} 1_{\{Y_i=0\}} \\ 0 & \text{sinon.} \end{cases}$$


exemple de classification KNN
Default
0
1
amount
2.5
2.4
2.3
2.2
2.1
2.0
1.9
1.8
1.20
1.22
1.24
1.26
1.28
1.30
duration

La règle des plus proches voisins bénéficie de garanties théoriques.

En particulier, Stone (1977) a démontré que si k $\rightarrow +\infty$ et k/n $\rightarrow$ 0, alors,

quelle que soit la loi de (X, Y), $\mathbb{E}Ln \rightarrow L*$

Cette propriété s'appelle la **consistance universelle**

Application de la méthode

Contexte :

$$\begin{matrix} & X & & & Y \\ \begin{bmatrix} x_{1,1} & \dots & x_{1,m} \\ \dots & x_{i,j} & \dots \\ x_{n,1} & \dots & x_{n,m} \end{bmatrix} & & & & \begin{bmatrix} y_1 \\ \dots \\ y_n \end{bmatrix} \end{matrix}$$

où Y est **discrète** et **prédéterminée**

Exemple :

X Y

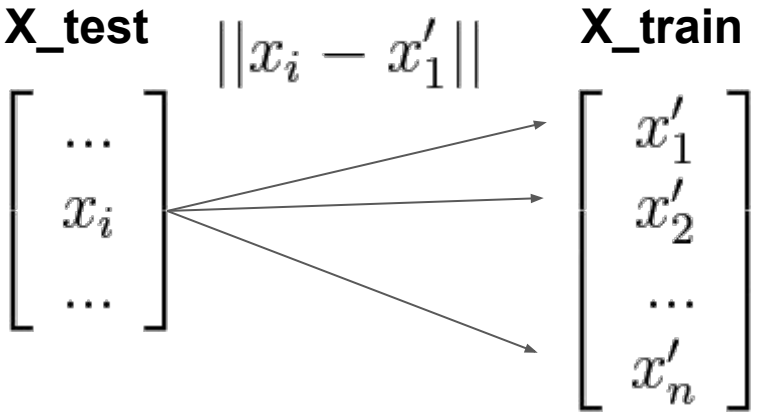
n = 1000

##	duration	amount	installment	age	history	purpose	Default
## 1	6	1169	4	67	terrible	goods/repair	0
## 2	48	5951	2	22	poor	goods/repair	1
## 3	12	2096	2	49	terrible	edu	0
## 4	42	7882	2	45	poor	goods/repair	0
## 5	24	4870	3	53	poor	newcar	1
## 6	36	9055	2	35	poor	edu	0

Y = “Default” appartient à {0,1}

X_train (700 individus)
X_test (300 individus)

Coder l'algorithme



	Distance	Classe
d(xi, x1')	0.1	0
...	...	...
d(xi, xn')	10	1

On réarrange et on sélectionne les K plus proches voisins. → Vote

Choix des distances

Données quantitatives :

>>> Distance Euclidienne : $||X - Y|| = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$

Données qualitatives:

>>> Distance de Hamming : $d(X, Y) = \#\{i : x_i \neq y_i\}$

exemple : Soit x et y deux vecteurs avec 4 classes possibles : { 0 , 1 , 2 , 3 }

x1 = (1 , 0 , 1 , 3) et y1 = (1 , 0 , 1 , 2) >>> D1 = 1

x2 = (0 , 2 , 1 , 3) et y2 = (1 , 2 , 3 , 0) >>> D2 = 3

Complexité algorithmique

1ère méthode :

$$||x_i - x'_1|| = \sqrt{||x_i||^2 + ||x'_i||^2 - 2\langle x_i, x'_i \rangle}$$

Complexité	Précision	Temps
O(n^2)	69%	12s

2ème méthode :

n

n

$$\begin{bmatrix} ||x_1 - x'_1|| & \dots & ||x_1 - x'_n|| \\ \dots & ||x_i - x'_i|| & \dots \\ \dots & \dots & ||x_n - x'_n|| \end{bmatrix}$$

Complexité	Précision	Temps
O(n^2)	68%	1s

3ème méthode :

Algorithme de référence : k-d tree

Complexité	Précision	Temps
O(nlog(n))	70%	0.01 s

Choix du nombre de voisins

Le meilleur K sera celui qui fournira la meilleure précision :

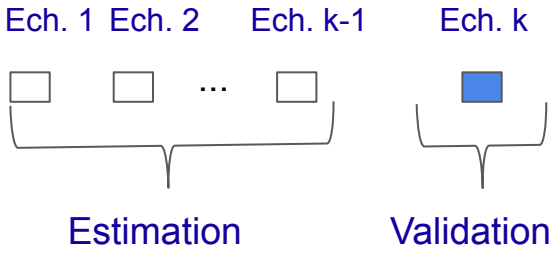
Rappel :

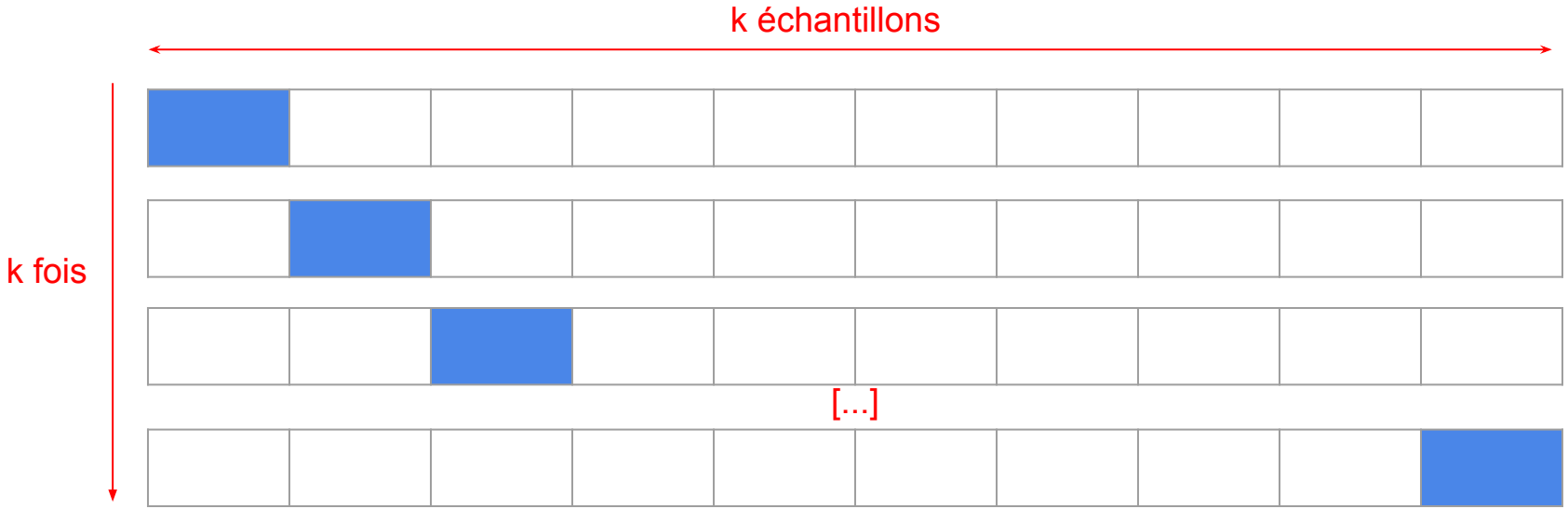
décision \ état de la nature	1	0
1	VP	FP
0	FN	VN

$$P = \frac{VP + VN}{VP + FP + FN + VN}$$

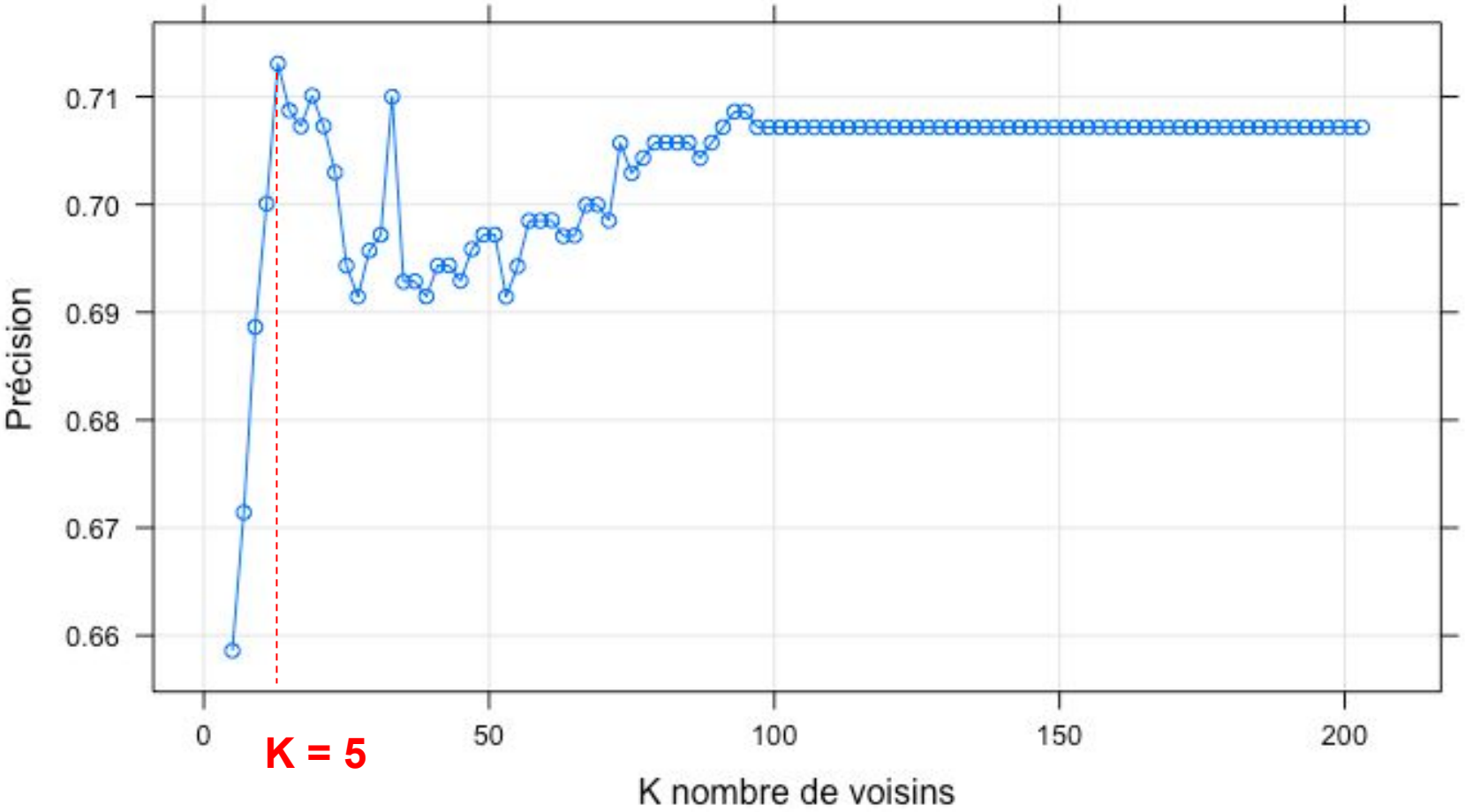
Calcul de la précision sur X_test => **surapprentissage !**

Validation croisée k-fold





Choix du nombre de voisins



Conclusion

Avantages :

- Fonctionnement facile à comprendre
- Facile à implémenter
- La phase d'apprentissage est très rapide
- Se généralise bien aux problèmes de classification multiclasses
- Méthode locale -> robuste aux valeurs extrêmes

Inconvénients :

- La phase de validation est **coûteuse en temps**
- Sensible au **déséquilibre des classes**
- Méthode non paramétrique -> **Fléau de la dimension**