

CES DATA scientist

Apprentissage supervisé : théorie et algorithmes

Joseph Salmon

<http://josephsalmon.eu>

Télécom Paristech, Institut Mines-Télécom

Motivation – enjeux

- ▶ Point de départ : diverses méthodes usuelles et retour sur quelques éléments de statistiques
- ▶ Compréhension de méthodes classiques de classification
- ▶ Implémenter celles-ci en Python

Plan

Prérequis

Optimisation / statistiques / Algèbre
Probabilités

Cadre et notations

Contexte
Modèle
Du cadre binaire au cadre multi-classe

Quelques méthodes de classification

Prédiction linéaire et indicatrices
Analyse discriminante linéaire (LDA)
Analyse discriminante quadratique (QDA)
Bayésien Naïf
Régression logistique
K-nn

Sommaire

Prérequis

Optimisation / statistiques / Algèbre

Probabilités

Cadre et notations

Quelques méthodes de classification

Optimisation / statistiques

Optimisation/Analyse

- ▶ Projection sur un sous-espace
- ▶ Régression linéaire, moindre carrés :
$$Y = X\theta + \varepsilon, \quad \hat{\theta} = (X^\top X)^{-1} X^\top Y \text{ (quand } X^\top X \text{ est inversible)}$$
- ▶ Méthode de Newton

Algèbre

- ▶ Décomposition spectrale des matrices symétriques (carrées) :

$$\Sigma = UDU^\top \in \mathbb{R}^{p \times p}$$

pour une matrice orthonormale U (i.e., $U^\top U = \text{Id}_p$) et une matrice diagonale $D = \text{diag}(d_1, \dots, d_p)$

Sommaire

Prérequis

Optimisation / statistiques / Algèbre

Probabilités

Cadre et notations

Quelques méthodes de classification

Probabilités

- ▶ Les lois gaussiennes en dimension p quelconque ont comme densité la forme suivante :

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{p/2}\sqrt{|\Sigma|}} \exp\left\{-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right\} .$$

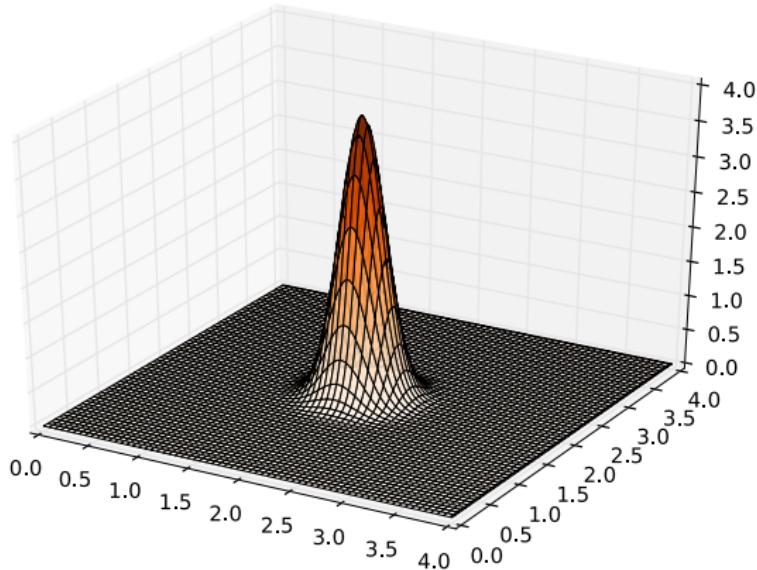
qui est gouvernée par deux paramètres :

- ▶ le vecteur d'espérance $\boldsymbol{\mu} \in \mathbb{R}^p$
- ▶ la matrice de covariance $\boldsymbol{\Sigma} \in \mathbb{R}^{p \times p}$ (symétrique)

Rem: $|\Sigma| = \det(\Sigma)$ est le produit des valeurs propres de Σ

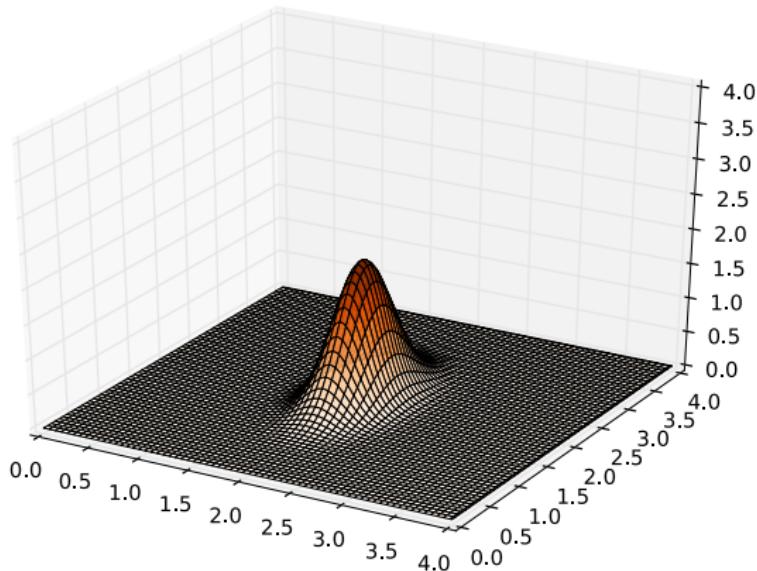
Lois gaussiennes

$$\Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} : \text{cas isotrope}$$



Lois gaussiennes

$$\Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} : \text{cas anisotrope}$$



Sommaire

Prérequis

Cadre et notations

Contexte

Modèle

Du cadre binaire au cadre multi-classe

Quelques méthodes de classification

La classification

Cadre binaire :

- ▶ Classer des patients entre malades / sains
- ▶ Classer des pourriels contre des emails normaux
- ▶ Classer bons payeurs (ou transactions normales) / mauvais payeurs (fraudes)
- ▶ Classer surfeur web entre futur acheteur ou non (contexte de la publicité en ligne)

Cadre multi-classe :

- ▶ Classer des chiffres numérisés (pour classer des codes postaux de courriers)
- ▶ Classer des espèces animales/végétales (e.g., iris)
- ▶ Classer des textes par thème (e.g., RCV20)

Exemples de variables explicatives

- ▶ cadre médical : âge, CPS, patrimoine génétique, examens, patrimoine génétique, antécédent, etc.
- ▶ cadre de la détection de pourriels : langue, niveau de langages, mots-clefs (e.g., discount, Nigeria, etc.)
- ▶ cadre de la publicité en ligne : historique de navigation, cookie, site web, âge, etc.

Numérisation des variables explicatives

Toutes les variables ne peuvent pas être utilisées telles quelles, il faut souvent faire un pré-traitement :

- ▶ Pour des variables continues pas de soucis (e.g., âge, températures, distances, etc.)
- ▶ pour des variables catégoriques il faut faire un traitement particulier

Solution 1 : “OneHotEncoder” : on code chaque modalité par une variable binaire.

Exemple : Hommes vs Femmes, variable homme, variable femme.

Limite : si l'on a beaucoup de modalités, on crée beaucoup de variables.

Solution 2 : on code chaque modalités par un nombre, mais il faut se méfier car la proximité ne veut alors rien dire...

Plus d'informations :

<http://fastml.com/converting-categorical-data-into-numbers-with-pandas-and-scikit-learn/>

“Feature engineering”

Création artificielle de variables explicatives :

- ▶ transformation simple de variables : terme de puissances de degré 2, 3, 4, . . . , log, exp etc.
- ▶ variables de types temporelles : cosinus/sinus avec des périodes adaptées
- ▶ interactions d'ordre supérieurs : par exemple les gens blonds avec des yeux bleus (catégories venant de deux variables explicatives couleur yeux / couleur cheveux)

Plus d'informations sur cette partie pré-traitement :

<http://scikit-learn.org/stable/modules/preprocessing.html>

Sommaire

Prérequis

Cadre et notations

Contexte

Modèle

Du cadre binaire au cadre multi-classe

Quelques méthodes de classification

Notations pour la classification multi-classes

Classes (en : *label*) : $\mathcal{Y} = \{0, 1, \dots, K - 1\}$ (K classes)

Espace des caractéristiques (en : *features*) : $\mathcal{X} \subset \mathbb{R}^p$

Observations : l'utilisateur reçoit un ensemble d'apprentissage composé de n couples $(X_i, Y_i) \sim (X, Y)$ supposés *i.i.d.*, où

- ▶ les $X_i \in \mathcal{X}$ correspondent aux **variables explicatives**
- ▶ $Y_i \in \mathcal{Y}$ correspondent aux **étiquettes**.

Formalisme vectoriel :

- ▶ Vecteur des étiquettes : $\mathbf{Y} = (Y_1, \dots, Y_n)^\top \in \mathbb{R}^n$
- ▶ Matrice des caractéristiques/attributs :
 $\mathbf{X} = (X_1, \dots, X_n)^\top \in \mathbb{R}^{n \times p}$

Exemple: Modèle binaire quand $K = 2$. Parfois utile de prendre $\mathcal{Y} = \{-1, 1\}$ (notamment quand on travaille avec des hyperplans)

Classification supervisée

Objectif

Être capable pour un nouvel élément $X_{n+1} \in \mathcal{X}$ d'estimer sa classe Y_{n+1} par une quantité $\hat{Y}_{n+1} \in \mathcal{Y}$

Sommaire

Prérequis

Cadre et notations

Contexte

Modèle

Du cadre binaire au cadre multi-classe

Quelques méthodes de classification

De deux à plusieurs classes

On peut passer du cadre binaire au multi-classe pour toute méthode, e.g., il suffit de tester :

- ▶ “un contre tous” (en : **One-vs.-all**) : créer un classifieur par classe, et produire un score (par exemple une probabilité). Choisir alors la classe avec le score maximum.
- ▶ “un contre un” (en : **One-vs.-one**) : on calcule un classifieur pour toutes les $K(K - 1)/2$ paires. Pour la prédiction on calcule tous les choix possibles et l'on prend la classe qui a reçu le plus de votes.

Outils de diagnostique

Autre type d'information : "Matrice de confusion"

Estimer les quantités $\mathbb{P}(\hat{Y}_{n+1} = k | Y = k')$ pour tout k, k'
c'est-à-dire que l'on veut estimer pour toutes les classes la
probabilités d'estimer une autre classe.

Cela permet de dépister les erreurs courantes (e.g., les chiffres 7 et 1 peuvent être très souvent confondus, 1 et 8 rarement)

Voir aussi : Courbe ROC, AUC, F1, etc.

Rem: Il y a de nombreuses méthodes de mesure d'erreurs qui peuvent être utiles dans divers contextes.

Leur énumération est donnée par exemple ici :

http://scikit-learn.org/stable/modules/model_evaluation.html

Sommaire

Prérequis

Cadre et notations

Quelques méthodes de classification

Prédiction linéaire et indicatrices

Analyse discriminante linéaire (LDA)

Analyse discriminante quadratique (QDA)

Bayésien Naïf

Régression logistique

K-nn

Prédiction linéaire et indicatrices

Idée naïve : utiliser un outils de prédiction linéaire pour faire de la classification

Simple, mais ne marche pas (on va le voir quand même pour s'en convaincre)

Prédiction linéaire et indicatrices

Idée simple : utiliser une méthode de régression pour estimer $\mathbb{P}(Y = 0|X = x), \mathbb{P}(Y = 1|X = x), \dots, \mathbb{P}(Y = K - 1|X = x)$ et choisir la classe qui donne le plus grande probabilité.

Rem: $\mathbb{P}(Y_{n+1} = k|X = X_{n+1}) = \mathbb{E}(Z^{(k)}|X = X_{n+1})$

Solution possible : résoudre K problèmes de régression, $k = 0, \dots, K - 1$ on définit $Z^{(k)} \in \mathbb{R}^n$ le vecteur de coordonnées

$$Z_i^{(k)} = \mathbb{1}_{Y_i=k} = \begin{cases} 1 & \text{si } Y_i = k, \\ 0 & \text{sinon.} \end{cases}$$

Estimateur des moindres carrés $\boxed{\theta^{(k)} = \arg \min_{\theta \in \mathbb{R}^p} \|\mathbf{X}\theta - Z^{(k)}\|^2}$

Prédiction (conditionnelle) $Y_{n+1}^{(k)} = X_{n+1}\theta^{(k)}$

Le classifieur final

$$\hat{Y}_{n+1} = \arg \max_{k \in \{0, \dots, K-1\}} Y_{n+1}^{(k)} \quad (\text{ex-aequo départagés au hasard})$$

Exo: Retrouver la formule de l'estimateur des moindres carrés

Prédiction linéaire et indicatrice

Seconde interprétation :

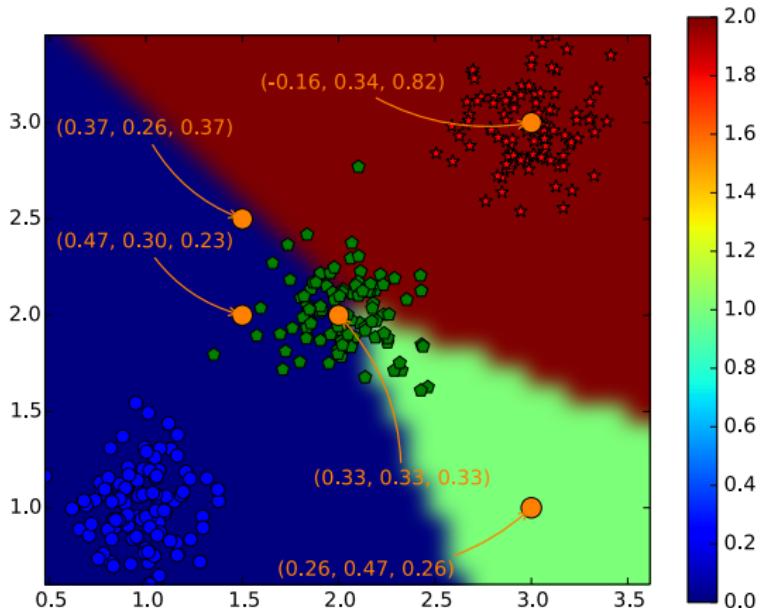
$$e_k = (0, \dots, 0, 1, 0, \dots, 0)^\top \in \mathbb{R}^K \text{ (1 en k-ième place)}$$

$$\arg \min_{M \in \mathbb{R}^{K \times d}} \left(\sum_{i=1}^n \|e_{Y_i} - X_i^\top M\|^2 \right)$$

puis

$$\hat{Y}_{n+1} = \arg \min_{k \in \{0, \dots, K-1\}} \|e_k - X_{n+1}^\top M\|^2$$

Un exemple : prédition linéaire et indicatrices



Données avec trois classes : les triplets indiquent les probabilités estimées des classes 0, 1 et 2 aux points indiqués

Avantages / Inconvénients : prédition linéaire et indicatrice

Avantages

- ▶ Simple : sans hypothèse de modèle (encore que)
- ▶ Implémentable facilement avec un solveur de moindres-carrés
- ▶ $\sum_{k=0}^{K-1} Y_{n+1}^{(k)} = 1$ si la matrice des caractéristiques contient la variable constante (*i.e.*, une colonne de 1)

Exo: Prouver ce point (utiliser de projection sur les colonnes)

Inconvénients

- ▶ les estimations $Y_{n+1}^{(k)}$ de $\mathbb{E}(Z^{(k)}|X = X_{n+1})$ n'ont pas de raison d'être positives, et peuvent donc ne pas l'être !
- ▶ Solution ? : $\theta^{(k)} = \arg \min_{\theta \in \mathbb{R}^p} \|\mathbf{X}\theta - Z^{(k)}\|^2$ ne résoud pas le problème, pour un nouveau pt la prédition peut être négative
- ▶ effet masque

TP: Faire la partie TP associée

Sommaire

Prérequis

Cadre et notations

Quelques méthodes de classification

Prédiction linéaire et indicatrices

Analyse discriminante linéaire (LDA)

Analyse discriminante quadratique (QDA)

Bayésien Naïf

Régression logistique

K-nn

Analyse discriminante linéaire (I)

Hypothèse de mélange gaussien

Pour tout k , la loi conditionnelle de X sachant $Y = k$ est gaussienne $\mathcal{N}_p(\mu_k, \Sigma_k)$ (on note f_k leur densités respectives)

- ▶ Vecteurs des centres de classes : $\mu_k \in \mathbb{R}^p$
- ▶ Matrices de covariance : Σ_k sont symétriques de taille $p \times p$
- ▶ Probabilités de la classe k : $\pi_k = \mathbb{P}\{Y = k\}$

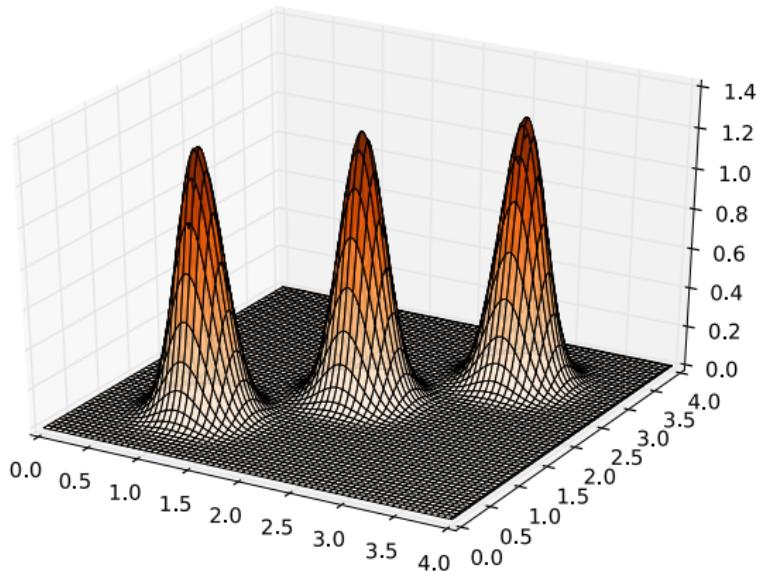
Mélanges : on tire avec probabilité π_k une étiquette $Y = k$, qui indique si X est tiré selon la loi f_k . La densité du mélange est donc

$$f(\mathbf{x}) = \sum_{k=0}^{K-1} \pi_k f_k(\mathbf{x})$$

Rappel : la densité p -dimensionnelle de la loi $\mathcal{N}_p(\mu_k, \Sigma_k)$ est

$$f_k(\mathbf{x}) = \frac{1}{(2\pi)^{p/2} \sqrt{|\Sigma_k|}} \exp \left\{ -\frac{1}{2} (\mathbf{x} - \mu_k)^\top \Sigma_k^{-1} (\mathbf{x} - \mu_k) \right\} .$$

Exemple de mélange ($K = 3, p = 2$)



Analyse discriminante linéaire (II)

La formule de Bayes donne :

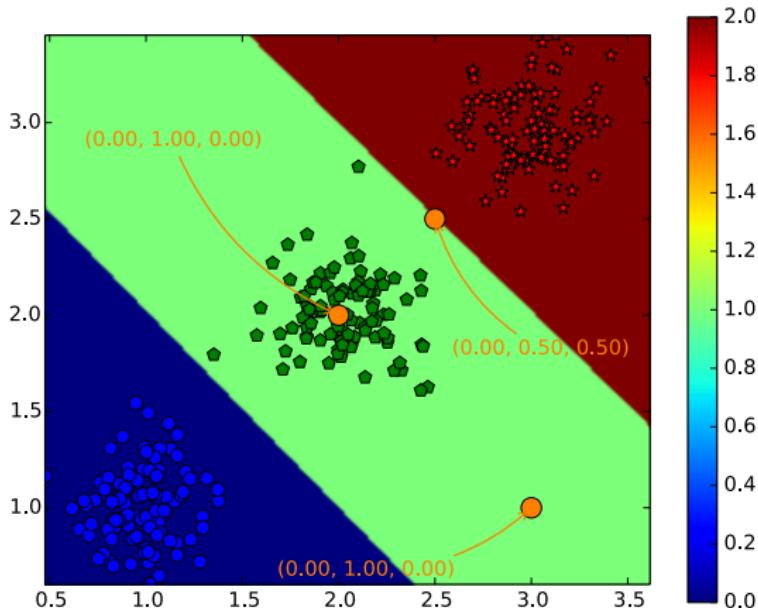
$$\mathbb{P}(Y = k | X = \mathbf{x}) = \frac{\pi_k f_k(\mathbf{x})}{\sum_{k'=0}^{K-1} \pi_{k'} f_{k'}(\mathbf{x})}$$

Rapport de log-vraisemblance sous l'hypothèse de **covariances identiques** $\Sigma_k = \Sigma, \quad \forall k \in \{0, \dots, K-1\}$

$$\begin{aligned} \log \left(\frac{\mathbb{P}(Y = k | X = \mathbf{x})}{\mathbb{P}(Y = l | X = \mathbf{x})} \right) &= \mathbf{x}^\top \Sigma^{-1} (\mu_k - \mu_l) + \frac{1}{2} (\mu_l^\top \Sigma^{-1} \mu_l - \mu_k^\top \Sigma^{-1} \mu_k) \\ &\quad + \log \left(\frac{\pi_k}{\pi_l} \right) \end{aligned}$$

Les séparatrices sont **linéaires** (affines) : on affecte \mathbf{x} à la classe k si $\mathbf{x}^\top \Sigma^{-1} (\mu_l - \mu_k) + \frac{1}{2} (\mu_k^\top \Sigma^{-1} \mu_k - \mu_l^\top \Sigma^{-1} \mu_l) + \log \left(\frac{\pi_k}{\pi_l} \right) > 0$

Un exemple : LDA



Données avec trois classes : les triplets indiquent les probabilités estimées des classes 0, 1 et 2 aux points indiqués

Analyse discriminante linéaire (II)

Classifieur par LDA pour un nouveau point X_{n+1}

$$\hat{Y}_{n+1}^{\text{LDA}} = \arg \max_{k \in \{0, \dots, K-1\}} \left(-X_{n+1}^\top \Sigma^{-1} \mu_k + \frac{1}{2} (\mu_k^\top \Sigma^{-1} \mu_k) + \log(\pi_k) \right)$$

En pratique, remplacer les quantités théoriques (inconnues) π_k, μ_k et Σ par les contreparties empiriques :

$$\hat{\pi}_k = \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{\{Y_i=k\}} \text{ et } n_k = \sum_{i=1}^n \mathbb{1}_{\{Y_i=k\}}$$

$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i=1}^n X_i \mathbb{1}_{\{Y_i=k\}}$$

$$\hat{\Sigma} = \frac{1}{n} \sum_{k=0}^{K-1} n_k \times \frac{1}{n_k} \sum_{i=1}^n \mathbb{1}_{\{Y_i=k\}} (X_i - \hat{\mu}_k)(X_i - \hat{\mu}_k)^\top$$

Rem: noter que le fait que $\hat{\Sigma}$ soit inversible n'est en rien garantie !

Autre interprétation

$$\hat{Y}_{n+1}^{\text{LDA}} = \arg \min_{k \in \{0, \dots, K-1\}} \frac{1}{2} (X_{n+1} - \hat{\mu}_k)^\top \hat{\Sigma}^{-1} (X_{n+1} - \hat{\mu}_k) - \log(\hat{\pi}_k)$$

Interprétation

Centrer et réduire les données ("sphériser") :

$$\begin{aligned}(X_{n+1} - \hat{\mu}_k)^\top \hat{\Sigma}^{-1} (X_{n+1} - \hat{\mu}_k) &= \|\hat{\Sigma}^{-1/2} (X_{n+1} - \hat{\mu}_k)\|^2 \\ &= \|\tilde{X}_{n+1} - \tilde{\mu}_k\|^2\end{aligned}$$

Avec la décomposition spectrale $\hat{\Sigma} = UDU^\top$ et

$$\tilde{X}_{n+1} = D^{-1/2} U^\top \hat{X}_{n+1}, \quad \tilde{\mu}_k = D^{-1/2} U^\top \hat{\mu}_k$$

Rem: On parle de sphérisation car si $\text{Var}(\mathbf{x}) = \hat{\Sigma}$ alors
 $\text{Var}(D^{-1/2} U^\top \mathbf{x}) = D^{-1/2} U^\top \hat{\Sigma} (D^{-1/2} U^\top)^\top = \text{Id}_p$

Accélération quand $K < p$ (peu de classes)

$H = \text{vect}(\tilde{\mu}_0, \dots, \tilde{\mu}_{K-1})$ vérifie ($\dim H \leq K$) (K vecteurs de \mathbb{R}^p).

Projetons sur H avec Π_H et utilisons Pythagore

$$\begin{aligned}\|\tilde{X}_{n+1} - \tilde{\mu}_k\|^2 &= \|\Pi_H(\tilde{X}_{n+1}) + \Pi_{H^\perp}(\tilde{X}_{n+1}) - \tilde{\mu}_k\|^2 \\ &= \|\Pi_H(\tilde{X}_{n+1}) - \tilde{\mu}_k\|^2 + \|\Pi_{H^\perp}(\tilde{X}_{n+1})\|^2\end{aligned}$$

Rem: le dernier terme ne dépend pas de k (omis dans l'optim. en k), et en general $K \ll n, p$

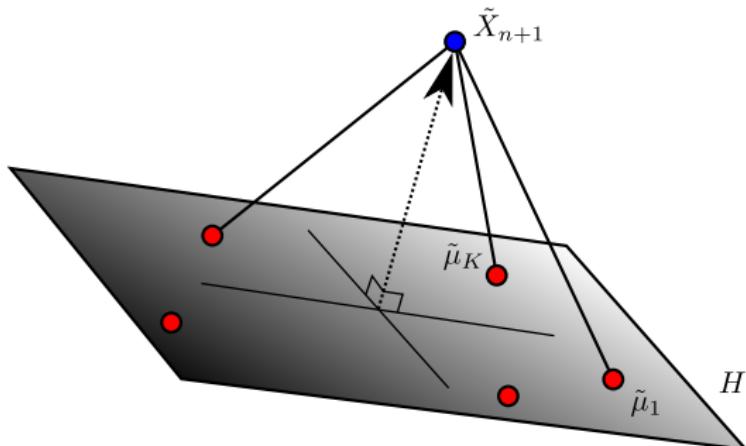


FIGURE: $K = 5, \dim(H) = 2, p = 3$

Accélération quand $K < p$ (peu de classes)

$H = \text{vect}(\tilde{\mu}_0, \dots, \tilde{\mu}_{K-1})$ vérifie ($\dim H \leq K$) (K vecteurs de \mathbb{R}^p).

Projetons sur H avec Π_H et utilisons Pythagore

$$\begin{aligned}\|\tilde{X}_{n+1} - \tilde{\mu}_k\|^2 &= \|\Pi_H(\tilde{X}_{n+1}) + \Pi_{H^\perp}(\tilde{X}_{n+1}) - \tilde{\mu}_k\|^2 \\ &= \|\Pi_H(\tilde{X}_{n+1}) - \tilde{\mu}_k\|^2 + \|\Pi_{H^\perp}(\tilde{X}_{n+1})\|^2\end{aligned}$$

Rem: le dernier terme ne dépend pas de k (omis dans l'optim. en k), et en general $K \ll n, p$

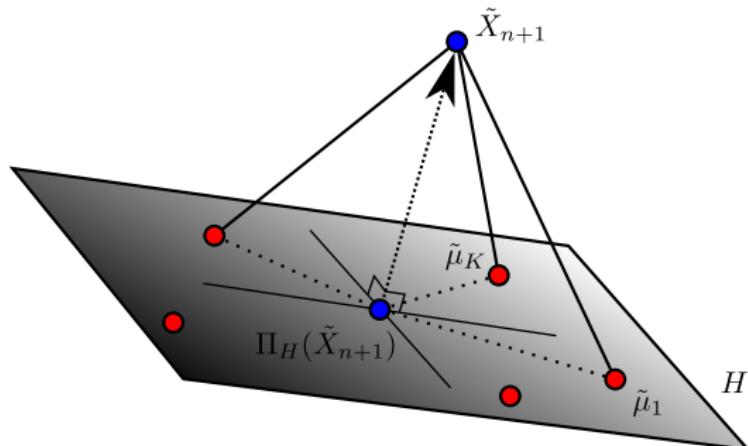


FIGURE: $K = 5, \dim(H) = 2, p = 3$

Accélération quand $K < p$ (peu de classes)

$H = \text{vect}(\tilde{\mu}_0, \dots, \tilde{\mu}_{K-1})$ vérifie ($\dim H \leq K$) (K vecteurs de \mathbb{R}^p).

Projetons sur H avec Π_H et utilisons Pythagore

$$\begin{aligned}\|\tilde{X}_{n+1} - \tilde{\mu}_k\|^2 &= \|\Pi_H(\tilde{X}_{n+1}) + \Pi_{H^\perp}(\tilde{X}_{n+1}) - \tilde{\mu}_k\|^2 \\ &= \|\Pi_H(\tilde{X}_{n+1}) - \tilde{\mu}_k\|^2 + \|\Pi_{H^\perp}(\tilde{X}_{n+1})\|^2\end{aligned}$$

Rem: le dernier terme ne dépend pas de k (omis dans l'optim. en k), et en general $K \ll n, p$

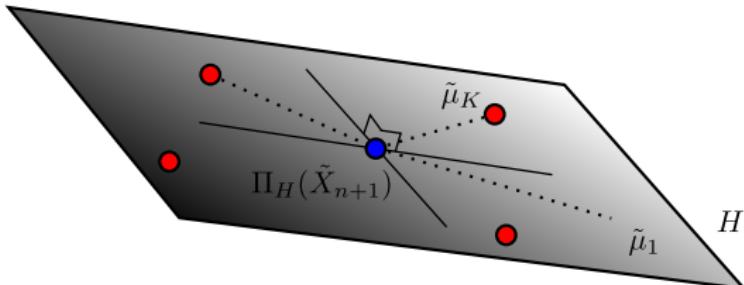


FIGURE: $K = 5, \dim(H) = 2, p = 3$

Algorithme récapitulatif

1. Calculer pour tout k , $\hat{\pi}_k, \hat{\mu}_k$ puis $\hat{\Sigma}$
2. Projeter les observations X_1, \dots, X_{n+1} sur l'espace $H = \text{vect}(\tilde{\mu}_0, \dots, \tilde{\mu}_{K-1})$ avec Π_H
3. Trouver

$$\arg \min_{k \in \{0, \dots, K-1\}} \left(\|\Pi_H(\tilde{X}_{n+1}) - \tilde{\mu}_k\|^2 - \log(\hat{\pi}_k) \right)$$

Exo: Implémenter cette fonction sous Python soit en utilisant la projection sur H , soit en utilisant un solveur de moindre-carrés
Rappel : si $\tilde{M} = [\tilde{\mu}_0, \dots, \tilde{\mu}_{K-1}]$, la projection sur $\text{vect}(\tilde{\mu}_0, \dots, \tilde{\mu}_{K-1})$ s'écrit $\tilde{M}(\tilde{M}^\top \tilde{M})^{-1} \tilde{M}^\top$, quand \tilde{M} inversible.

Code source “évolué” de scikit-learn :

<https://github.com/scikit-learn/scikit-learn/blob/master/sklearn/lda.py>

Avantages / Inconvénients : LDA

Avantages

- ▶ optimal pour des gaussiennes
- ▶ pas d'effet masque

Inconvénients

- ▶ Inversion de la covariance : besoin éventuellement de régulariser $\hat{\Sigma}$
- ▶ Les points très loin des frontières ont tous la même influence
- ▶ Robustesse aux hypothèses gaussiennes...

TP: Faire la partie TP associée

Sommaire

Prérequis

Cadre et notations

Quelques méthodes de classification

Prédiction linéaire et indicatrices

Analyse discriminante linéaire (LDA)

Analyse discriminante quadratique (QDA)

Bayésien Naïf

Régression logistique

K-nn

Analyse discriminante quadratique (I)

Hypothèse provisoire

Pour tout k , la loi conditionnelle de X sachant $Y = k$ est gaussienne $\mathcal{N}_p(\mu_k, \Sigma_k)$ (on note f_k leur densités respectives)

Hypothèse identique que sous le cadre LDA :

$$f(\mathbf{x}) = \sum_{k=0}^{K-1} \pi_k f_k(\mathbf{x})$$

Cette fois on ne fait plus l'hypothèse : $\forall k = 0, \dots, K-1, \hat{\Sigma}_k = \Sigma$

Analyse discriminante quadratique (II)

Rapport de log-vraisemblance : sans supposer que

$$\Sigma_k^{-1} = \Sigma^{-1}, \quad \forall k \in \{0, \dots, K-1\}$$

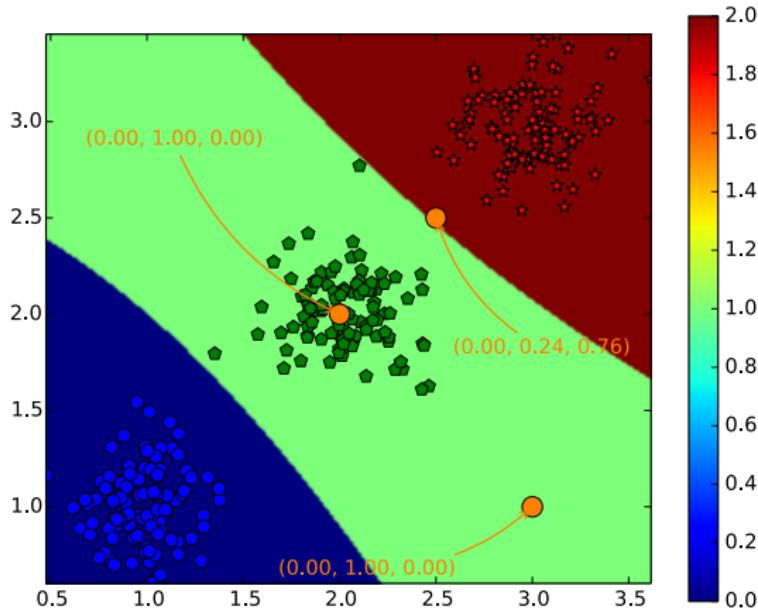
$$\begin{aligned} \log \left(\frac{\mathbb{P}(Y=k|X=\mathbf{x})}{\mathbb{P}(Y=l|X=\mathbf{x})} \right) &= \mathbf{x}^\top \Sigma^{-1} (\mu_k - \mu_l) + \frac{1}{2} (\mu_l^\top \Sigma^{-1} \mu_l - \mu_k^\top \Sigma^{-1} \mu_k) \\ &\quad + \log \left(\frac{\pi_k}{\pi_l} \right) \\ &\quad - \boxed{\log(|\Sigma_k|/|\Sigma_l|) + \mathbf{x}^\top (\Sigma_k^{-1} - \Sigma_l^{-1}) \mathbf{x}} \end{aligned}$$

Nouvelle règle :

$$\begin{aligned} \hat{Y}_{n+1}^{\text{QDA}} &= \\ \arg \min_{k \in \{0, \dots, K-1\}} & \left[\frac{(X_{n+1} - \hat{\mu}_k)^\top \hat{\Sigma}_k^{-1} (X_{n+1} - \hat{\mu}_k)}{2} - \log(\hat{\pi}_k) + \log(|\hat{\Sigma}_k|) \right] \end{aligned}$$

Rem: les séparatrices sont **quadratiques**

Un exemple : QDA



Données avec trois classes : les triplets indiquent les probabilités estimées des classes 0, 1 et 2 aux points indiqués

Avantages / Inconvénients : QDA

Avantages

- ▶ Modèle plus riche/flexible (complexité plus grande)

Inconvénients

- ▶ plus lourd à calculer
- ▶ Les séparatrices ne sont plus linéaires

TP: Faire la partie TP associé

Sommaire

Prérequis

Cadre et notations

Quelques méthodes de classification

Prédiction linéaire et indicatrices

Analyse discriminante linéaire (LDA)

Analyse discriminante quadratique (QDA)

Bayésien Naïf

Régression logistique

K-nn

Bayésien Naïf gaussien (I)

Retour sur le Bayésien Naïf gaussien :

$$\mathbb{P}(Y = k | X = \mathbf{x}) = \frac{\pi_k f_k(\mathbf{x})}{\sum_{k'=0}^{K-1} \pi_{k'} f_{k'}(\mathbf{x})}$$

Supposons que cette fois les densité de chaque classe sont indépendantes sur toutes les dimensions :

$$f_k(\mathbf{x}) = \frac{1}{(2\pi)^{p/2} \sqrt{|\Sigma_k|}} \exp \left\{ -\frac{1}{2} (\mathbf{x} - \mu_k)^\top \Sigma_k^{-1} (\mathbf{x} - \mu_k) \right\} .$$

avec $\Sigma_k = \begin{bmatrix} \sigma_{1,k}^2 & 0 & \cdots & 0 \\ 0 & \sigma_{2,k}^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_{p,k}^2 \end{bmatrix}$

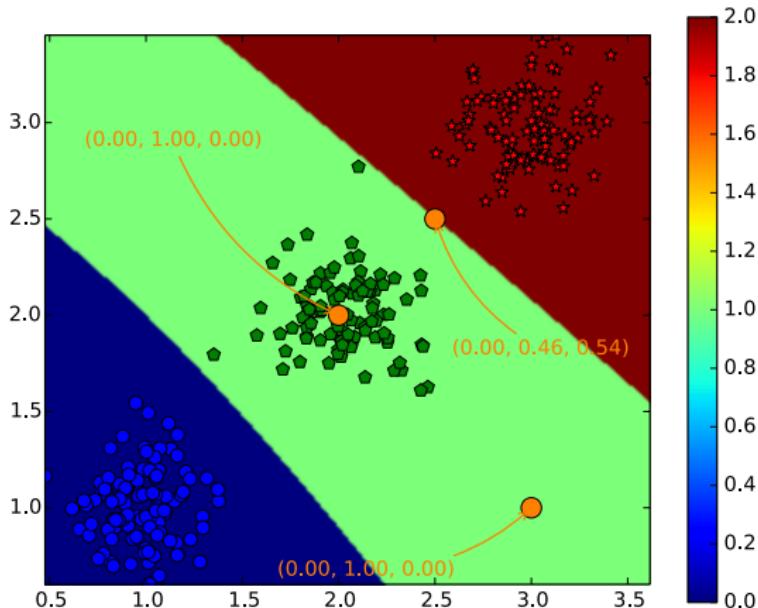
Règle Naïve de Bayes (gaussien)

Nouvelle règle :

$$\hat{Y}_{n+1}^{\text{NB}} = \arg \min_{k \in \{0, \dots, K-1\}} \left[\frac{(X_{n+1} - \hat{\mu}_k)^\top \Sigma_k^{-1} (X_{n+1} - \hat{\mu}_k)}{2} - \log(\hat{\pi}_k) + \log(|\hat{\Sigma}_k|) \right]$$

Rem: les séparatrices sont **quadratiques**

Un exemple : Bayésien Naïf



Données avec trois classes : les triplets indiquent les probabilités estimées des classes 0, 1 et 2 aux points indiqués

Avantages / Inconvénients : Bayésien Naïf

Avantages

- ▶ En pratique cela facilite les calculs : il ne faut plus calculer des matrices de covariance mais simplement les variances sur les p directions (par “nuage de points” correspondant aux classes)
- ▶ inversion facile des matrices diagonales !
- ▶ rapide pour de la grande dimension (e.g., très grands textes / spams)

Inconvénients

- ▶ les séparatrices ne sont plus linéaires
- ▶ connu pour avoir des probabilités estimées mauvaises
- ▶ “renforcement de rumeur” attention à ne pas rajouter des variables très corrélées (besoin d'un pré-écrémage)

Sommaire

Prérequis

Cadre et notations

Quelques méthodes de classification

Prédiction linéaire et indicatrices

Analyse discriminante linéaire (LDA)

Analyse discriminante quadratique (QDA)

Bayésien Naïf

Régression logistique

K-nn

Régression logistique : cas binaire

On suppose $K = 2$ et l'on souhaite modéliser les probabilités conditionnelles des classes, ou plutôt leur log-ratio, par des quantités linéaires (affines) :

$$\log \left(\frac{\mathbb{P}(Y = 0 | X = \mathbf{x})}{\mathbb{P}(Y = 1 | X = \mathbf{x})} \right) = \alpha + \langle \beta, \mathbf{x} \rangle$$

où $\alpha \in \mathbb{R}$ et $\beta \in \mathbb{R}^p$

Sous une telle hypothèse la séparatrice est **linéaire**, la règle étant simplement :

$$\alpha + \langle \beta, \mathbf{x} \rangle > 0$$

signifiant que l'on préfère la classe 0 à la classe 1 pour le point \mathbf{x}

Régression logistique (0')

On peut alors estimer les probabilités conditionnelles facilement :

$$\mathbb{P}(Y = 0 | X = \mathbf{x}) = \frac{\exp(\alpha + \langle \beta, \mathbf{x} \rangle)}{1 + \exp(\alpha + \langle \beta, \mathbf{x} \rangle)}$$

$$\mathbb{P}(Y = 1 | X = \mathbf{x}) = \frac{1}{1 + \exp(\alpha + \langle \beta, \mathbf{x} \rangle)}$$

Régression logistique : numériquement

“Maximisation de la (log-)vraisemblance”

Notant la (log-)vraisemblance la fonction ℓ des paramètres :

$$\begin{aligned}\ell(\alpha, \beta) &= \sum_{i=1}^n \log(\mathbb{P}(Y = Y_i | X = X_i, \alpha, \beta)) \\ &= \sum_{i=1}^n \sum_{k=0}^1 \mathbb{1}_{\{Y_i=k\}} \log(\mathbb{P}(Y = k | X = X_i, \alpha, \beta))\end{aligned}$$

on cherche alors une solution

$$(\hat{\alpha}, \hat{\beta}) \in \arg \max_{\alpha, \beta} \ell(\alpha, \beta)$$

Exo: montrer la formule qui suit :

$$\ell(\alpha, \beta) = \sum_{i=1}^n \left(Y_i(\alpha + \langle \beta, X_i \rangle) - \log[1 + \exp(\alpha + \langle \beta, X_i \rangle)] \right)$$

Régression logistique et méthode Newton

La Hessienne est calculable : on peut donc appliquer la méthode de Newton

Ceux intéressés par les détails techniques, peuvent trouver les calculs de la Hessienne [Hastie et al. \(2009, page 120\)](#)

<http://www-stat.stanford.edu/~tibs/ElemStatLearn/>

Régression logistique (I)

On tente ici de modéliser les probabilités conditionnelles des classes, ou plutôt leur log-ratio, par des quantités linéaires (affines) :

$$\log \left(\frac{\mathbb{P}(Y = k | X = \mathbf{x})}{\mathbb{P}(Y = K - 1 | X = \mathbf{x})} \right) = \alpha_k + \langle \beta_k, \mathbf{x} \rangle$$

où $\alpha_k \in \mathbb{R}$ et $\beta_k \in \mathbb{R}^p$ pour tout $k \in \{0, \dots, K - 1\}$

Sous une telle hypothèse les séparatrices sont **linéaires**, la règle étant simplement :

$$\alpha_k + \langle \beta_k, \mathbf{x} \rangle > 0$$

signifiant que l'on préfère la classe k à la classe $K - 1$ pour le point \mathbf{x}

Régression logistique (II)

On peut alors estimer les probabilités conditionnelles facilement :

Pour $k = 0, \dots, K - 2$:

$$\mathbb{P}(Y = k | X = \mathbf{x}) = \frac{\exp(\alpha_k + \langle \beta_k, \mathbf{x} \rangle)}{1 + \sum_{l=0}^{K-2} \exp(\alpha_l + \langle \beta_l, \mathbf{x} \rangle)},$$

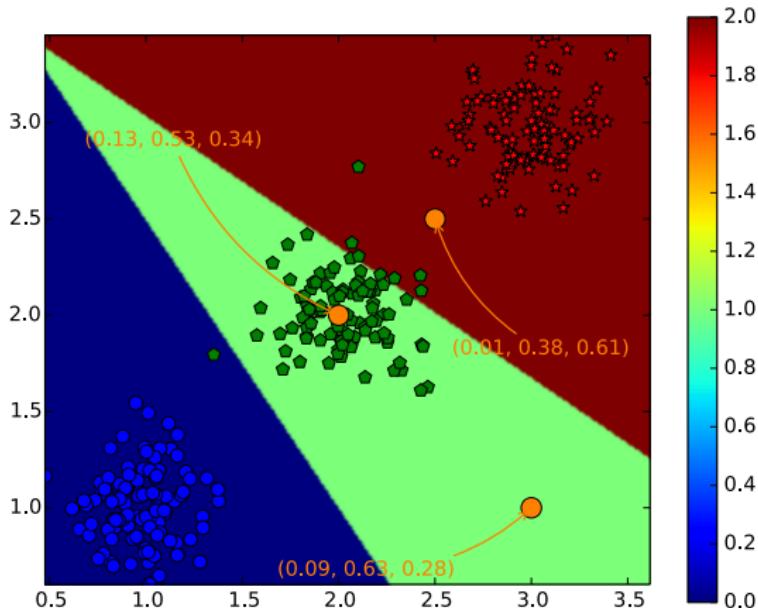
Pour $k = K - 1$:

$$\mathbb{P}(Y = K - 1 | X = \mathbf{x}) = \frac{1}{1 + \sum_{l=0}^{K-2} \exp(\alpha_l + \langle \beta_l, \mathbf{x} \rangle)}$$

Règle : on choisit la classe qui a la plus grande probabilité

Rem: Numériquement : le problème devient beaucoup plus dur (à écrire et à traiter) qu'en binaire, cf. [Hastie et al. \(2009\)](#)

Un exemple : régression logistique



Données avec trois classes : les triplets indiquent les probabilités estimées des classes 0, 1 et 2 aux points indiqués

Avantages / Inconvénients : régression logistique

Avantages

- ▶ connu pour avoir des probabilités estimées bonnes
- ▶ séparations linéaires

Inconvénients

- ▶ Classification binaire plus facile
- ▶ Problème d'optimisation plus complexe (temps de calcul)
- ▶ En pratique le cadre multi-classe est parfois géré par la technique du “un contre tous” et non par le cas logistique multinomiale (surtout si K est petit)

TP: Faire la partie TP associé

Sommaire

Prérequis

Cadre et notations

Quelques méthodes de classification

Prédiction linéaire et indicatrices

Analyse discriminante linéaire (LDA)

Analyse discriminante quadratique (QDA)

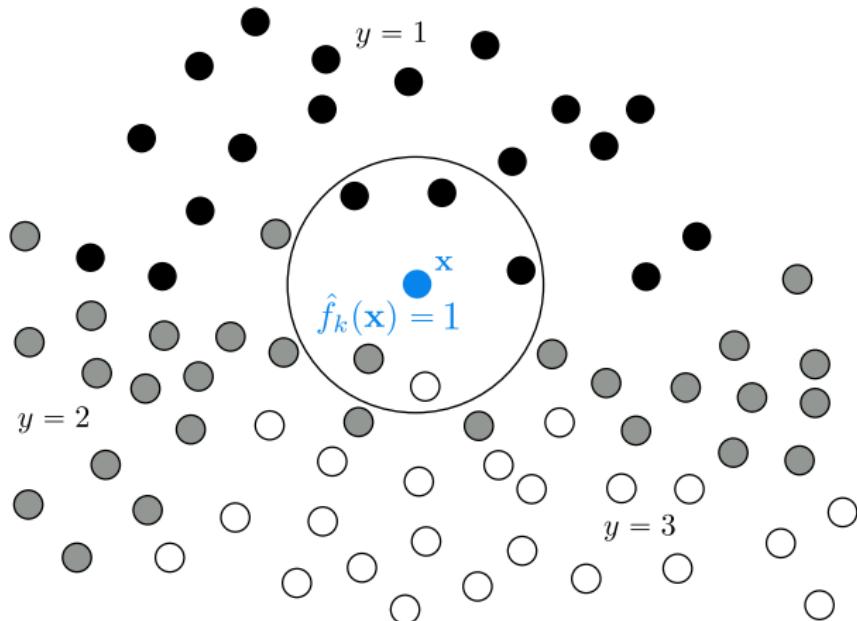
Bayésien Naïf

Régression logistique

K-nn

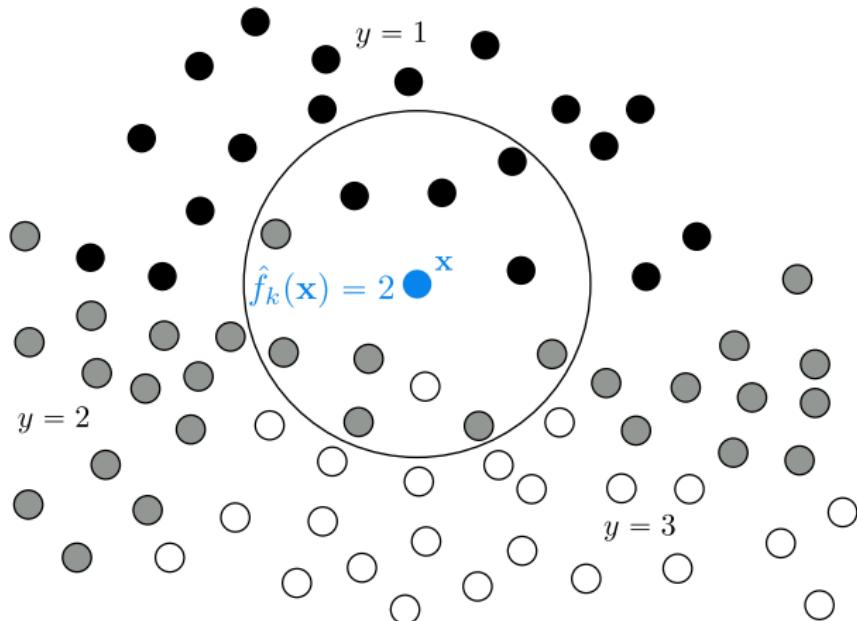
K-nn

Méthode des k -plus proches voisins pour des valeurs du paramètres $k = 5$ et $k = 11$. pour $K = 3$ classes noir ($y = 1$), gris ($y = 2$), blanc ($y = 3$).

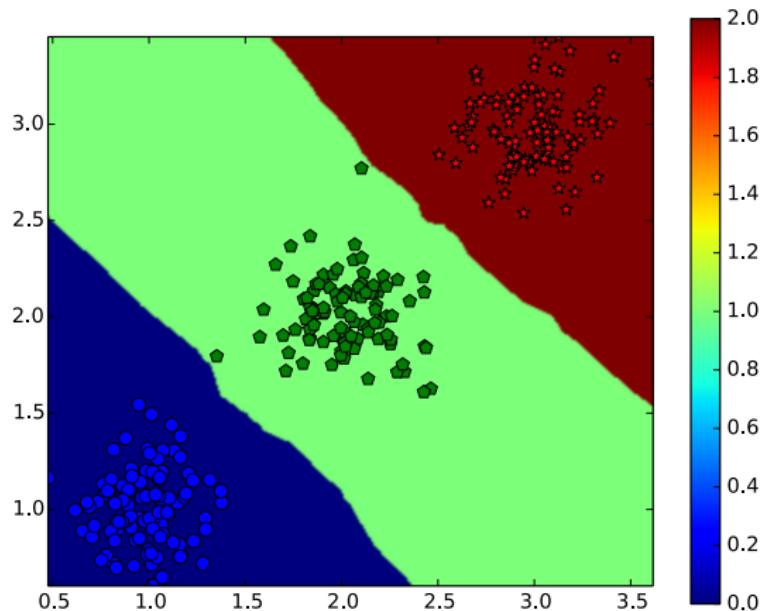


K-nn

Méthode des k -plus proches voisins pour des valeurs du paramètres $k = 5$ et $k = 11$. pour $K = 3$ classes noir ($y = 1$), gris ($y = 2$), blanc ($y = 3$).



Un exemple : K-nn



Données avec trois classes : ici on pas directement accès aux probabilités estimées des classes

Avantages / Inconvénients : K-nn

Avantages

- ▶ séparations non-convexe en général
- ▶ s'adapte avec tout type de distance

Inconvénients

- ▶ Multi-classe par défaut.
- ▶ temps de calcul peut-être long (calculer toutes les distances deux à deux, est-ce vraiment utile ?)

Références I

Bonus : feuille de route simplifiée pour le machine learning :
http://scikit-learn.org/stable/tutorial/machine_learning_map/

Aspects numériques sur LDA/QDA :

<http://www.stat.cmu.edu/~ryantibs/datamining/>

Divers :

- ▶ T. Hastie, R. Tibshirani, and J. Friedman.

The elements of statistical learning.

Springer Series in Statistics. Springer, New York, second edition, 2009.

<http://www-stat.stanford.edu/~tibs/ElemStatLearn/>.