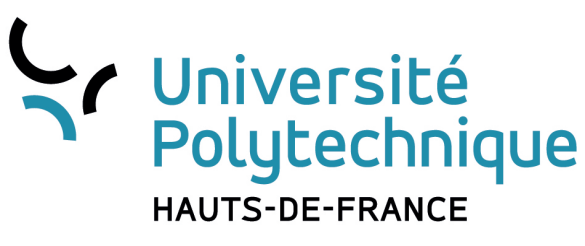


Apprentissage Statistique Automatique I

Méthodes linéaires pour la classification

Andrés F. López-Lopera
Université Polytechnique Hauts-de-France (UPHF)

Thèmes

1. Introduction

2. Régression linéaire

    Cadre binaire

    Cadre multi-classe

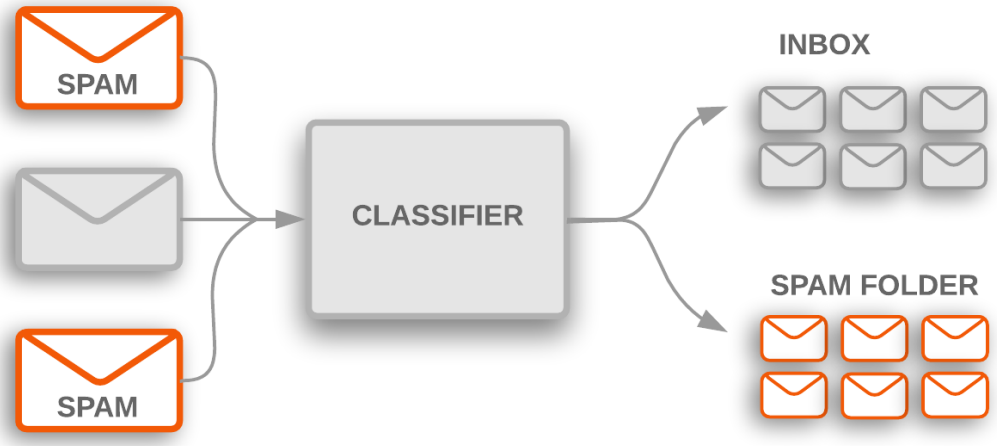
3. Régression logistique

4. Analyse discriminante linéaire

Introduction

Classification binaire

- Classer des emails (spams ou normaux)



[Google]

- Diagnostiquer des patients (malades ou sains)

- Classer des clients (fraudeurs ou bon payeurs)

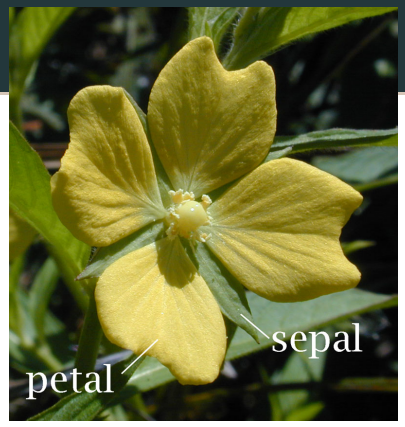
Classification multi-classe

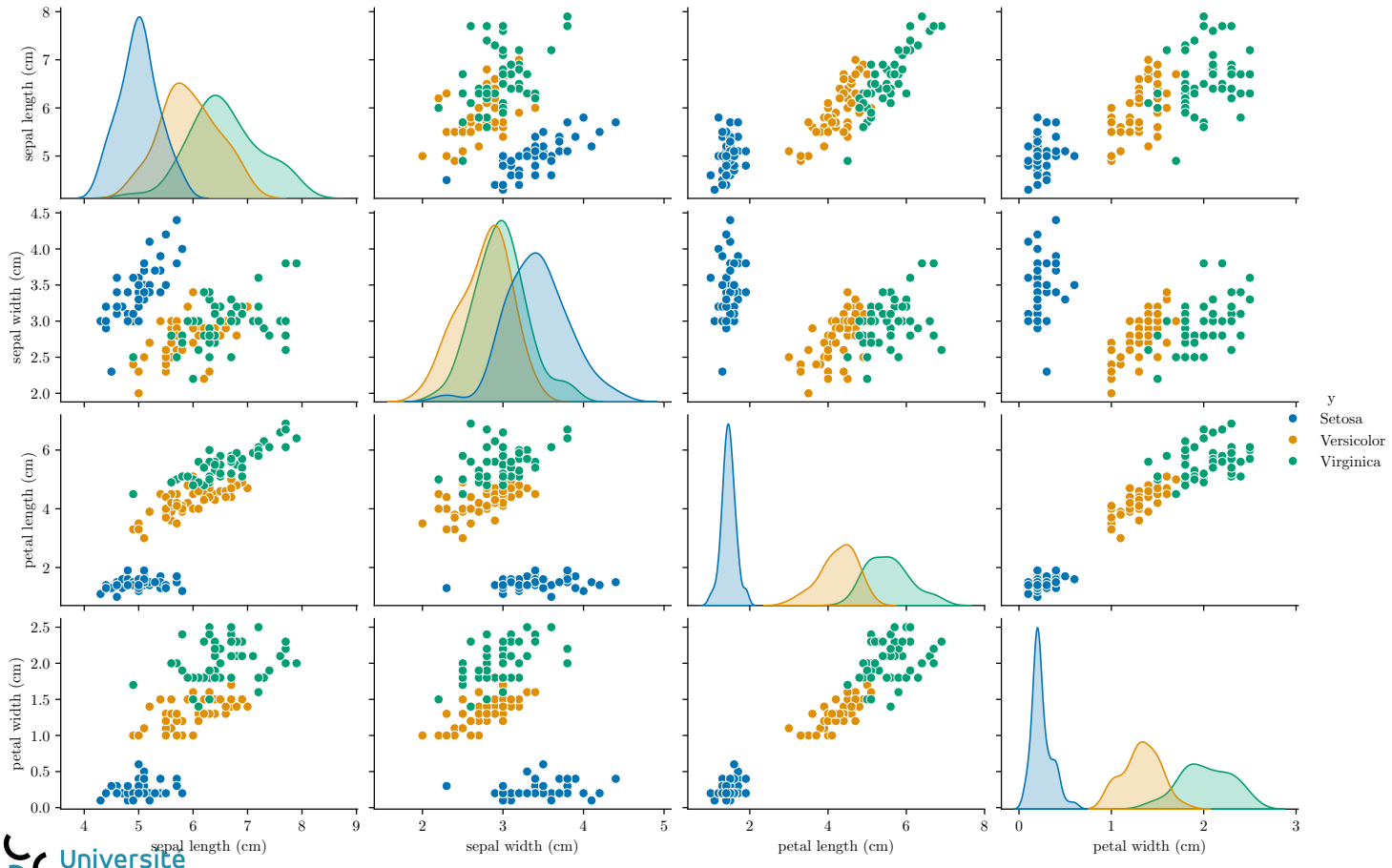
Cas de test - Iris (sklearn) [Pedregosa et al., 2011]

- 3 classes :

$$\kappa \in \{\texttt{setosa}, \texttt{versicolor}, \texttt{virginica}\}$$

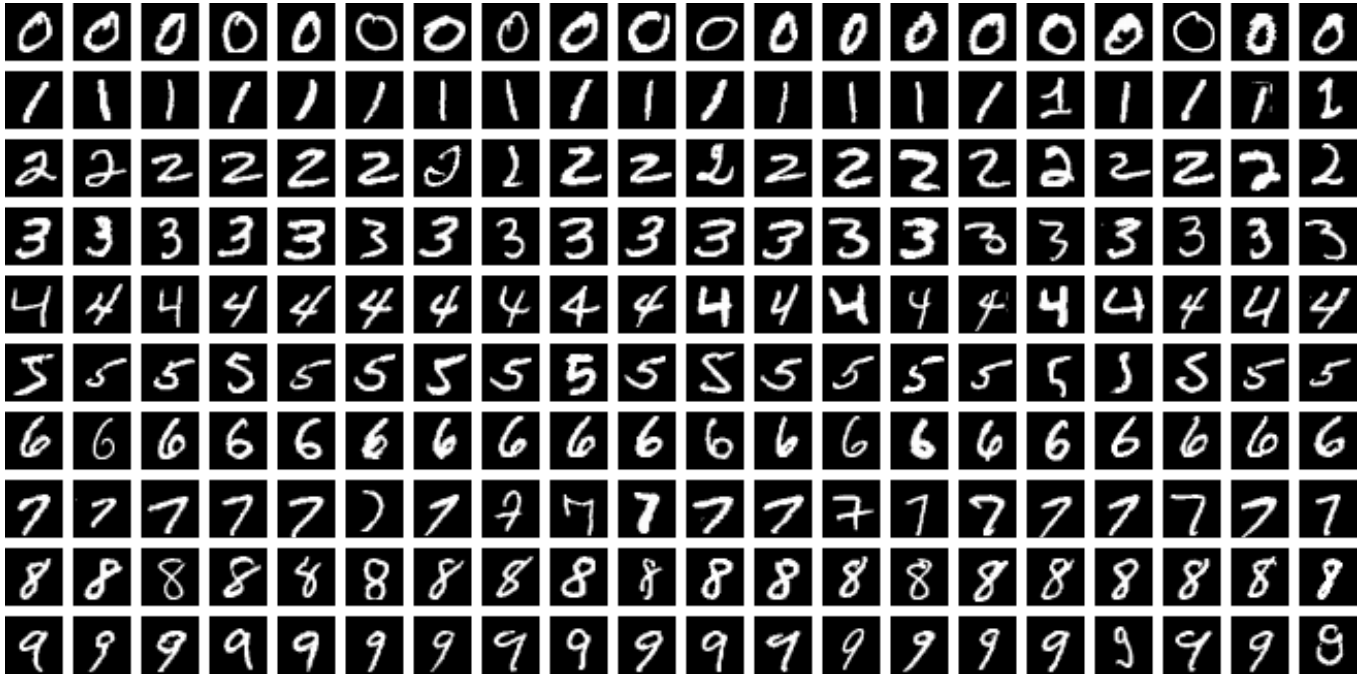
- 4 patterns : caractéristiques du sépale et du pétale





Classification multi-classe

- Classer des chiffres numérisés (80's/90's : scans de codes postaux, MNIST dataset)



- Classer des objets dans des images (2010's)
- Classer des textes par thème
- . . .

Problème de classification

· Soit la fonction cible

$$y : \quad \mathcal{X} \subseteq \mathbb{R}^d \to \mathcal{Y} \subseteq \mathbb{N}, \qquad x \mapsto y(x) \tag{1}$$

· Dans un cadre générale (multi-classe), $K \in \mathbb{N}$ représente le nombre de classes (disjointes !)

· On suppose que les classes sont indexées par

$$\mathcal{Y} = [\![0, K-1]\!] := \{0, \ldots, K-1\} \tag{2}$$

· On dénote $y \in [\![0, K-1]\!]^n$ pour les observations et $X = [x_1, \ldots, x_n]^\top \in \mathbb{R}^{n \times d}$ (n observations, d variables) pour les variables explicatives

Classifieur : c'est un estimateur

$$\eta_\theta : \begin{cases} \mathbb{R}^d & \mapsto [\![0, K-1]\!] \\ x & \to \eta_\theta(x) \end{cases} \tag{3}$$

Problème de classification

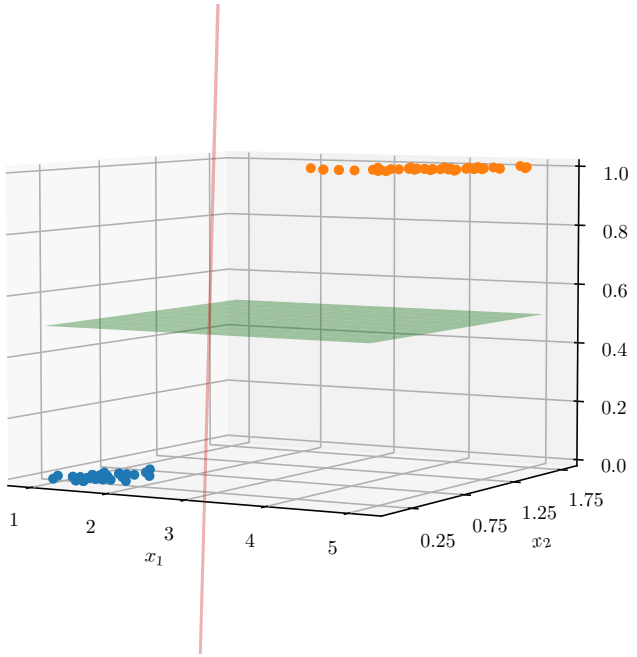
Objectif :

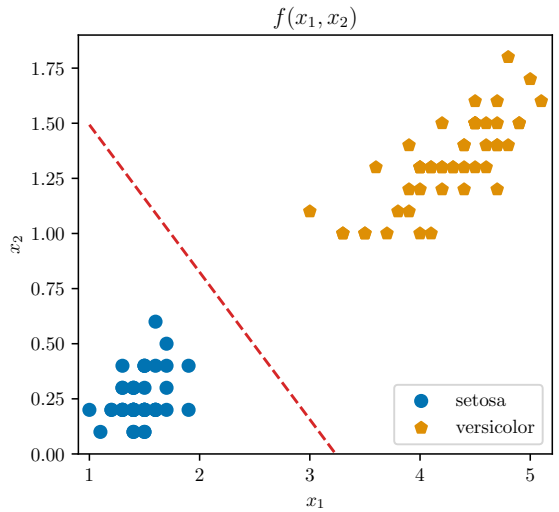
- Minimiser l'erreur : $\mathbb{P}(y \neq \eta_\theta(x)) = \mathbb{E}(\mathbb{1}_{y \neq \eta_\theta(x)})$

$\Leftrightarrow$ Maximiser la précision (*accuracy*) : $\mathbb{P}(y = \eta_\theta(x)) = \mathbb{E}(\mathbb{1}_{y = \eta_\theta(x)})$

· Ici, on considère des modèles linéaires, ceci signifie que les surfaces de décision (*decision surfaces*) sont des fonctions linéaires du x

· Elles sont donc définies par des hyperplans de dimension $d - 1$ dans un espace d'entrée de dimension d





● Classe 1 ● Classe 2 — surface de décision

Régression linéaire

Régression linéaire : cadre binaire

· Choisir comme classifieur

$$h_{\theta=(\beta_0,\beta)}(x) = \beta_0 + \sum_{j=1}^{d} \beta_j x_j = \beta_0 + \langle x, \beta \rangle, \tag{4}$$

où $x = [x_1, \ldots, x_d]^\top$ et $\beta = [\beta_1, \ldots, \beta_d]^\top$

· La règle de discrimination est

$$\eta_\theta(x) = \mathbb{1}_{h_\theta(x) \geq 0} := \begin{cases} 1 & \text{si } h_\theta(x) \geq 0, \\ 0 & \text{sinon,} \end{cases}$$

et dont la surface (ou borne) de décision est $h_\theta(x) = 0$ (hyperplan de dimension $d - 1$)

· Une règle possible de discrimination est donnée par

$$\eta_\theta(x) = \frac{1 + \text{sign}(h_\theta(x))}{2} \tag{5}$$

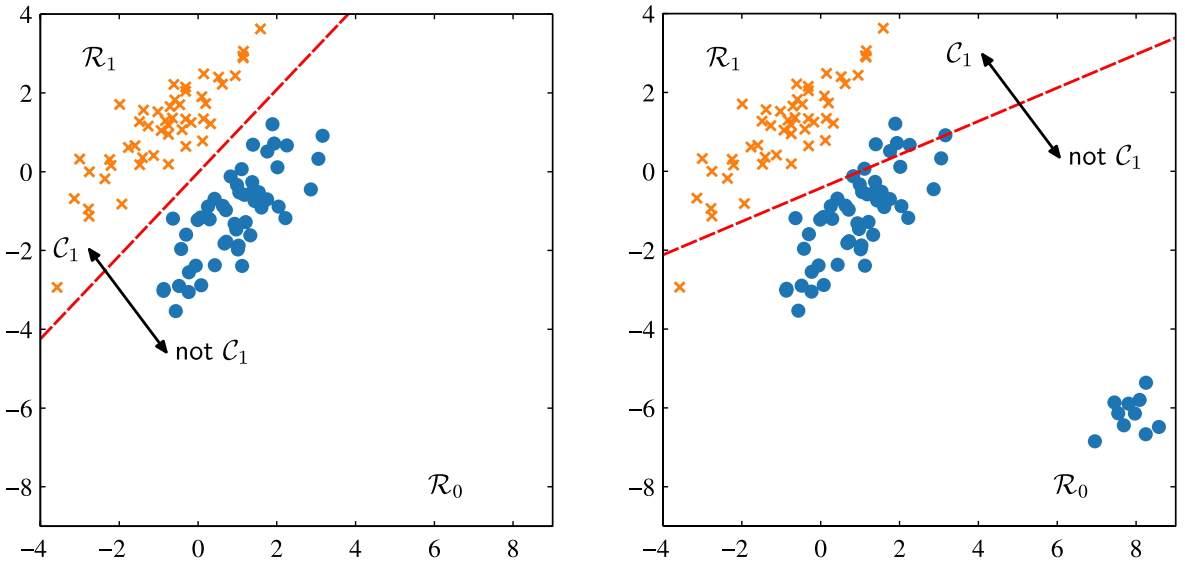
Rem : Il aussi possible de modéliser les classes $k \in \{-1, 1\}$. Dans ce cas, le classifieur est $\eta_\theta(x) = \text{sign}(h_\theta(x))$ (règle de discrimination plug-in)

Régression linéaire : cadre binaire

· La surface de décision définira une partition $\mathcal{X} = \mathcal{R}_1 \sqcup \mathcal{R}_2$ t.q.

$$\mathcal{R}_1 = \{x \in \mathcal{X} \mid h_\theta(x) \geq 0\},$$
$$\mathcal{R}_0 = \{x \in \mathcal{X} \mid h_\theta(x) < 0\}$$

· $\mathcal{R}_1$ et $\mathcal{R}_0$ représentent donc les régions de décision (*decision regions*)



· Comme pour la régression, le modèle linéaire est sensible aux *outliers*

Rem : les données dont les classes peuvent être séparées exactement par des surfaces de décision linéaires sont dits linéairement séparables

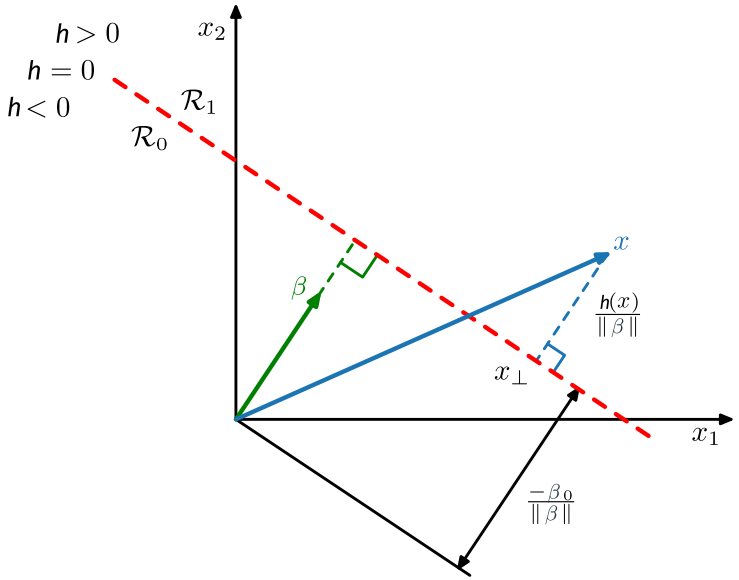
Régression linéaire : interprétation géométrique

· Supposons $x_A, x_B \in \mathcal{X}$ appartenant à la surface de décision

· Puisque $h(x_A) = h(x_B) = 0$, on a

$$h(x_A) - h(x_B) = \langle x_A - x_B, \beta \rangle = 0,$$

ce qui implique que β est orthogonal à la surface de décision



· Si $x \in \mathcal{X}$ est un point situé sur la surface de décision, alors $h(x) = 0$, et la distance normale entre l'origine et la surface de décision est donnée par :

$$\frac{\langle x, \beta \rangle}{\|\beta\|} = -\frac{\beta_0}{\|\beta\|}$$

· Considérons un point arbitraire $x \in \mathcal{X}$, et soit $x_\perp \in \mathcal{X}$ sa projection orthogonale sur la surface de décision, de sorte que :

$$x = x_\perp + r\frac{\beta}{\|\beta\|} \quad \Leftrightarrow \quad h(x) := \beta_0 + \langle x, \beta \rangle = \underbrace{\beta_0 + \langle x_\perp, \beta \rangle}_{=0} + r\frac{\langle \beta, \beta \rangle}{\|\beta\|} = r\|\beta\|$$

$$\Rightarrow r = \frac{h(x)}{\|\beta\|}$$

Méthode I : Traitement par régression linéaire brute

· Choisir comme classifieur $h_\beta(x) = \langle x, \beta \rangle$ (*i.e.*, faire le choix $y \approx X\beta$), ou plus exactement, la classe la plus proche

$$\eta_{\widehat{\beta}}(x) = \underset{k \in [\![0, K-1]\!]}{\operatorname{argmin}} \left| k - \langle \widehat{x}, \beta \rangle \right|, \tag{6}$$

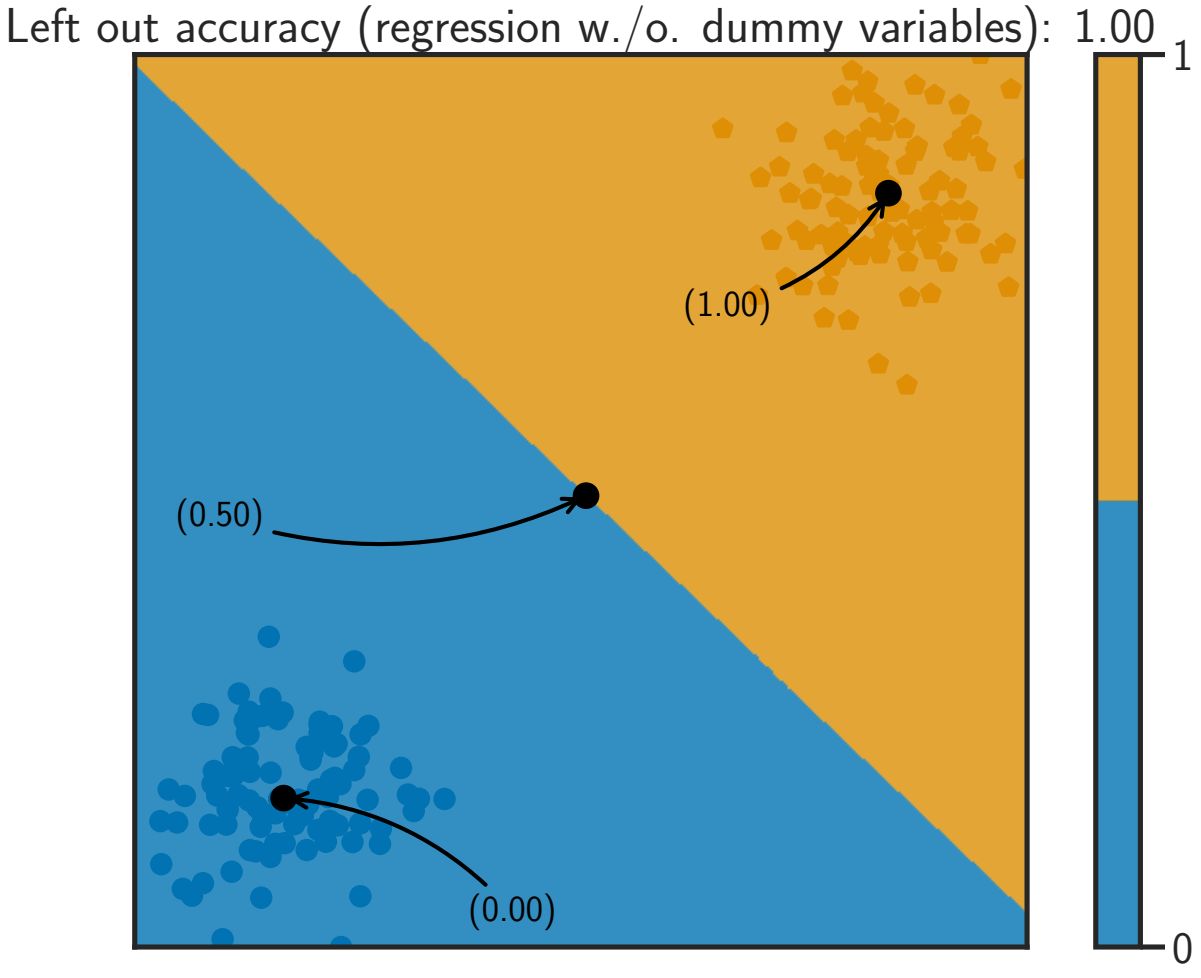
où les paramètres $\beta \in \mathbb{R}^p$ s'estiment par la méthode de moindres carrés:

$$\widehat{\beta} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \, \|y - X\beta\|^2,$$

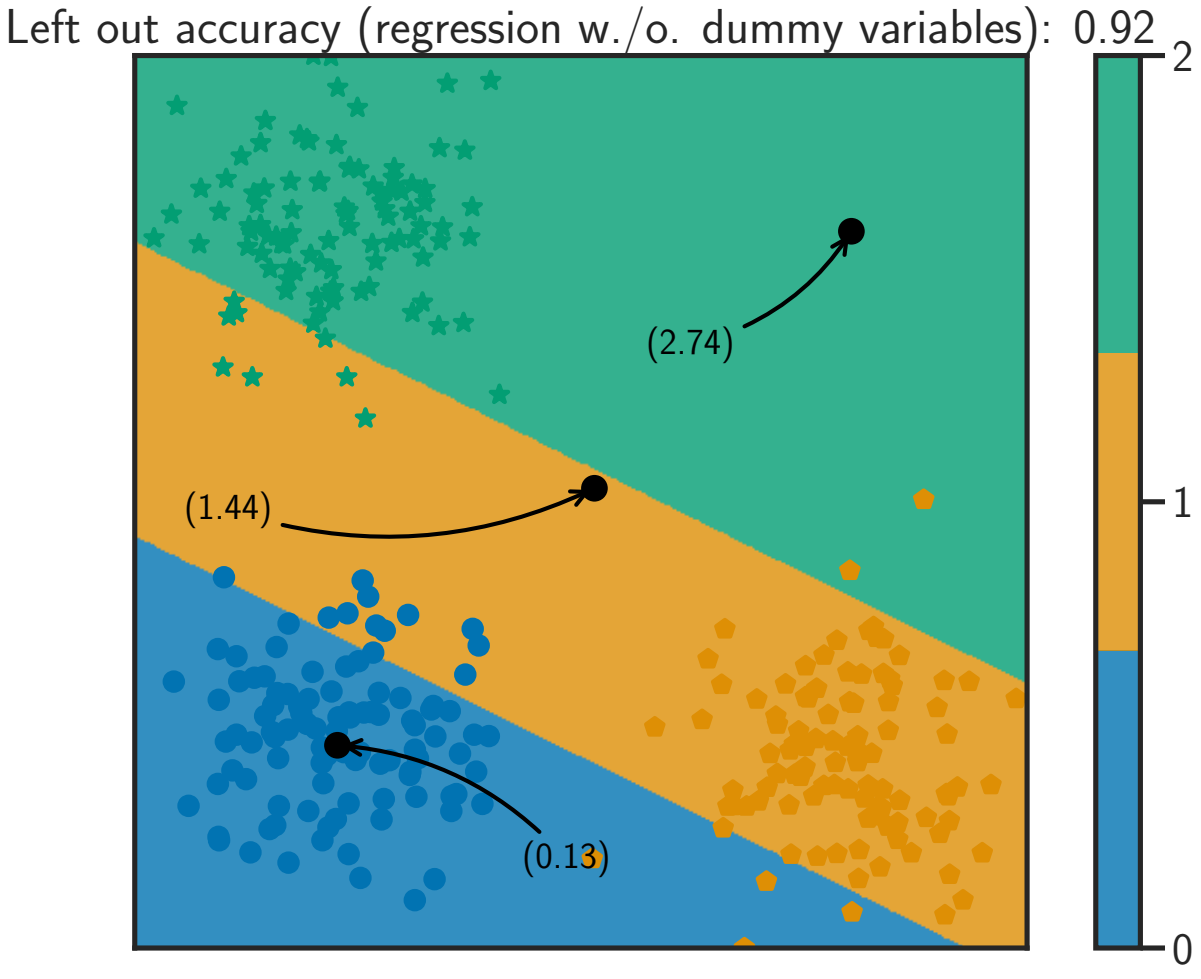
avec

$$y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \quad \beta = \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_d \end{bmatrix}, \quad X = \begin{bmatrix} 1 & x_{1,1} & \dots & x_{1,d} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & \dots & x_{n,d} \end{bmatrix}.$$

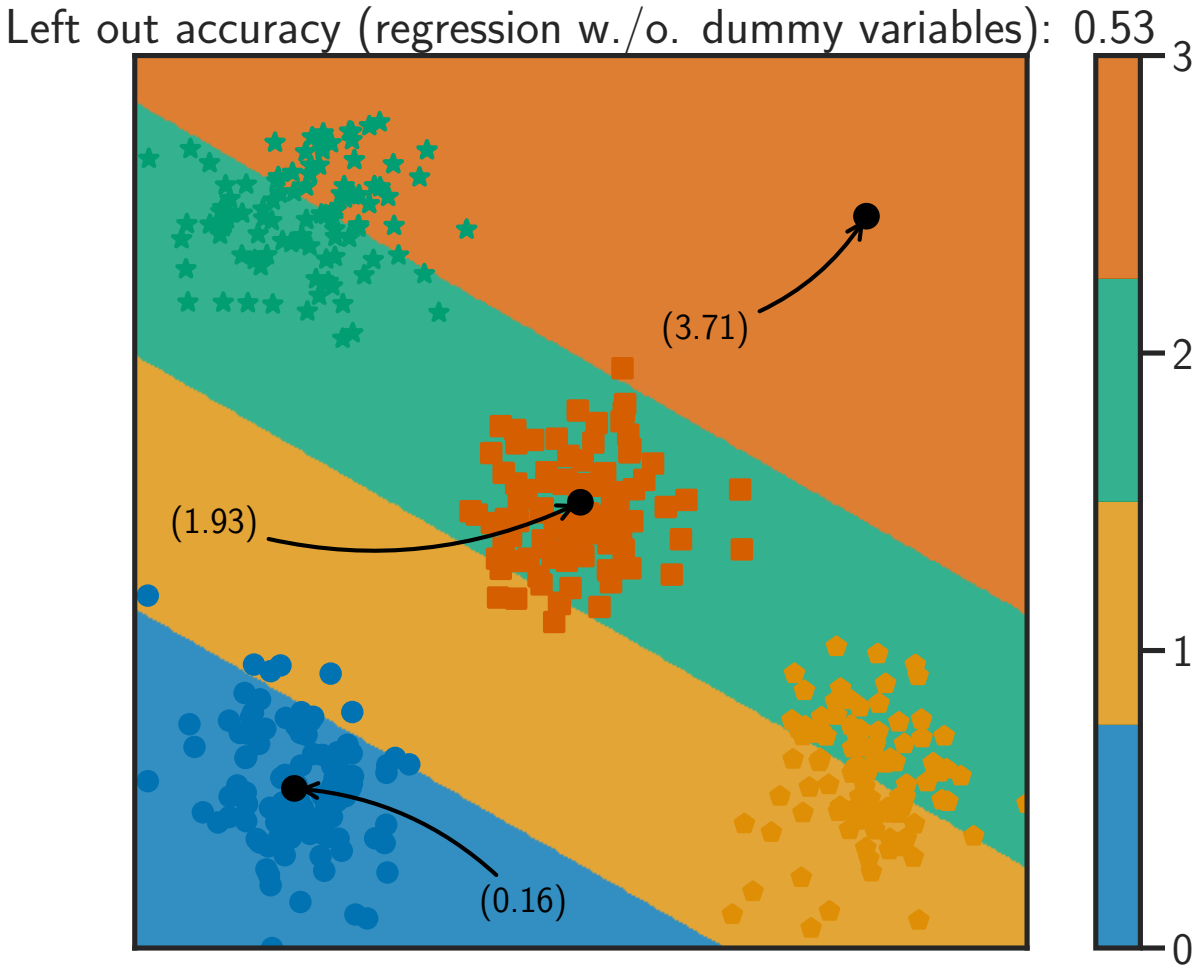
Un exemple avec $K = 2$



Un exemple avec $K = 3$



Un exemple avec $K = 4$



Régression linéaire : cadre multi-classe

Méthode II : Traitement par régression linéaire et classifieurs à deux classes

· On peut passer du cadre binaire au multi-classe pour **toute** méthode, *e.g.*, il suffit de tester les procédures suivantes [Bishop, 2006]

1) Procédure "un contre tous" (*one-versus-the-rest*)

- Créer un classifieur par classe, et produire un score (*e.g.*, une probabilité)
- Prédire la classe de score maximal
  Coût : K classifieurs

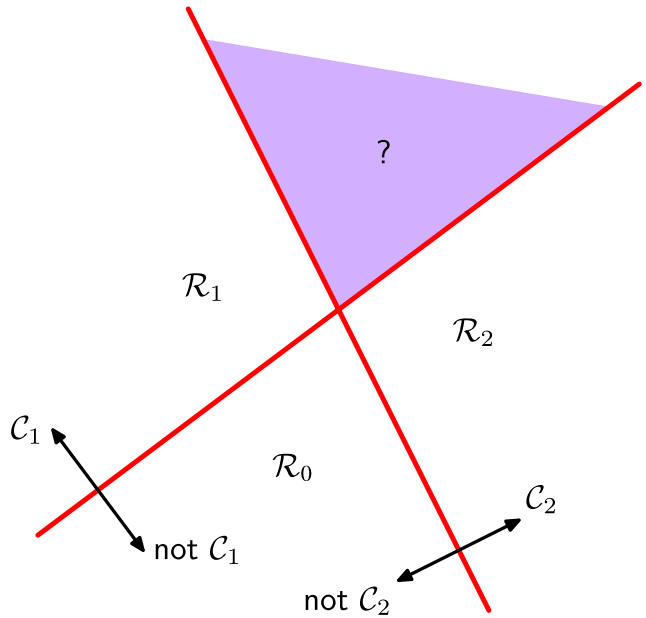
2) Procédure "un contre un" (*one-versus-one*)

- Calculer un classifieur pour toutes les $K(K-1)/2$ paires
- Prédire la classe qui gagne le plus de "duels"
  Coût : $K(K-1)/2$ classifieurs

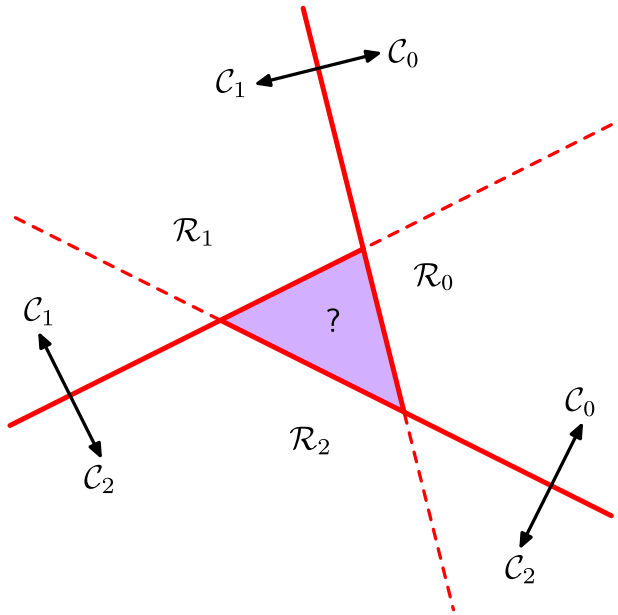
Rem : dans sklearn la descente par coordonnée avec LIBLINEAR utilise "one-versus-the-rest"

Régression linéaire : cadre multi-classe

· On peut cependant trouver des régions ambiguës



classifieur *“one-versus-the-rest”*



classifieur *“one-versus-one”*

Méthode III : Traitement par régression multi-classe et variables binaires

· On évite ces difficultés en considérant une seule règle à K classes constitué de K modèles linéaires de la forme suivante :

$$h_k(x) = \beta_0^{(k)} + \sum_{j=1}^{d} \beta_j^{(k)} x_j = \langle x, \beta^{(k)} \rangle \tag{7}$$

· Ces modèles s'utilisent pour estimer : $\mathbb{P}(y = 0|x), \ldots, \mathbb{P}(y = K-1|x)$

· On peut considérer des matrices d'indicatrice : $\mathbb{P}(y = k|x) = \mathbb{E}(Z^{(k)}|x)$ où

$$Z^{(k)} \in \mathbb{R}^n \quad Z_i^{(k)} = \mathbb{1}_{y_i=k} = \begin{cases} 1, \text{ si } y_i = k, \\ 0, \text{ sinon} \end{cases} \tag{8}$$

· Les paramètres $\beta^{(1)}, \ldots, \beta^{(K)}$ s'estiment par moindres carrées :

$$\widehat{B} := [\widehat{\beta}^{(1)}, \ldots, \widehat{\beta}^{(K)}] = \underset{B \in \mathbb{R}^{p \times K}}{\operatorname{argmin}} \frac{1}{2} \|Z - XB\|^2$$

Rem : on peut utiliser un solveur de moindres carrés multi-tâches

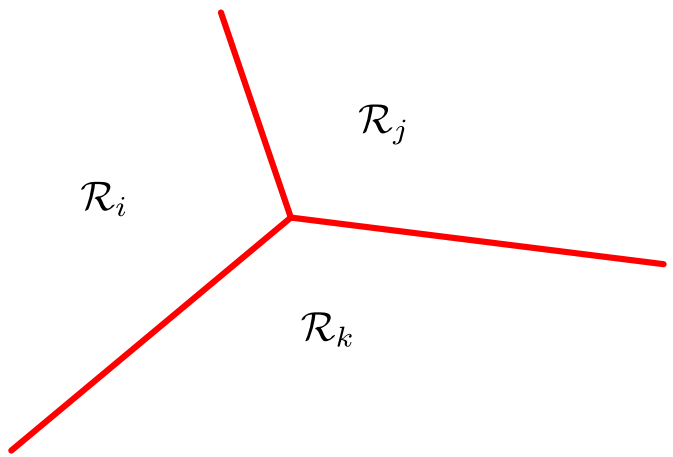
Régression linéaire : cadre multi-classe

· On choisit ensuite la probabilité estimée la plus grande, c'est-à-dire on associe la classe k si $h_k(x) > h_j(x)$ pour tout $j \neq k$, avec $j, k \in [\![0, K-1]\!]$

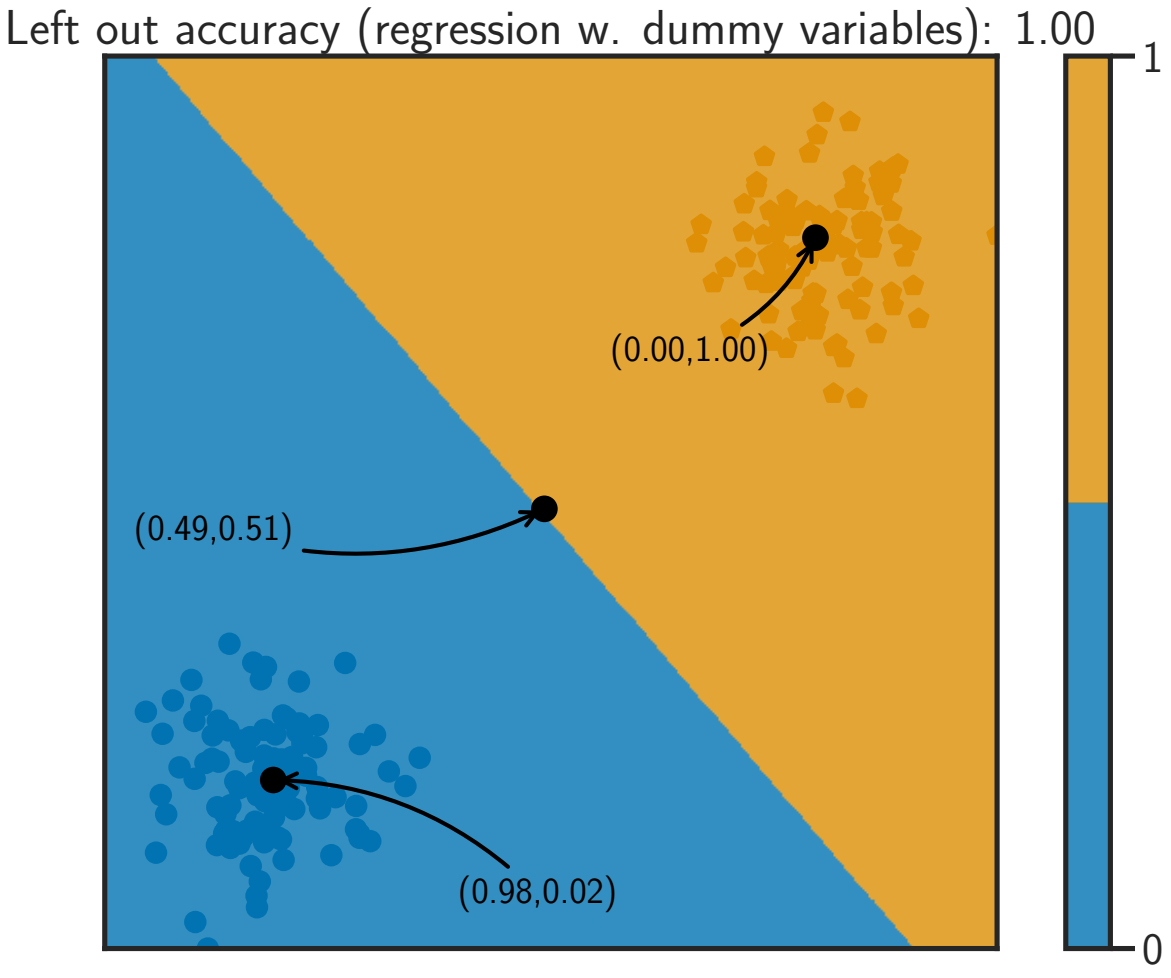
$$\eta_{\widehat{B}}(x) = \underset{k \in [\![0, K-1]\!]}{\operatorname{argmax}} \langle x, \widehat{\beta}^{(k)} \rangle$$

· La surface de décision entre les classes k et j est donc l'hyperplan de dimension $d-1$ qui satisfait $h_k(x) = h_j(x)$:

$$\langle x, \beta^{(k)} - \beta^{(j)} \rangle = 0$$

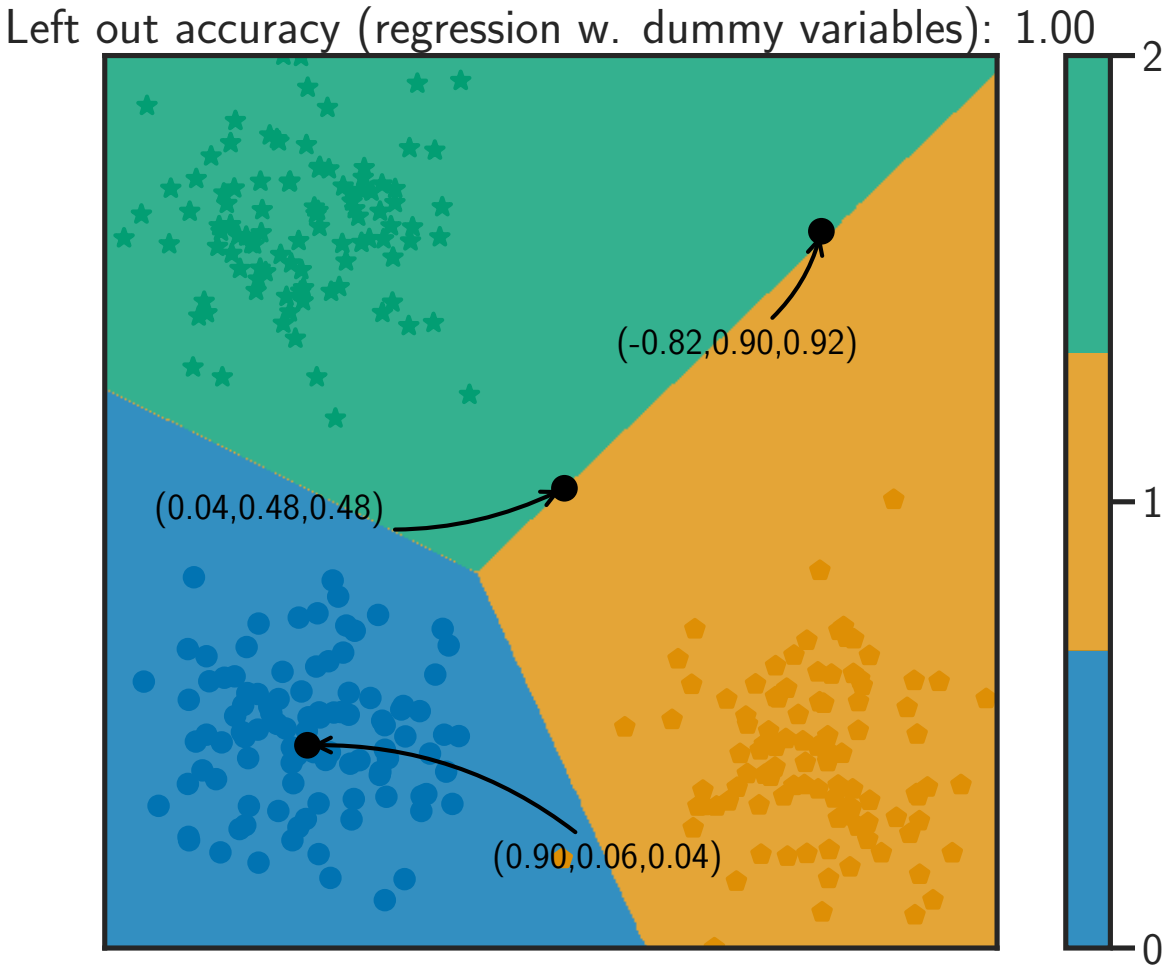


Un exemple avec $K = 2$



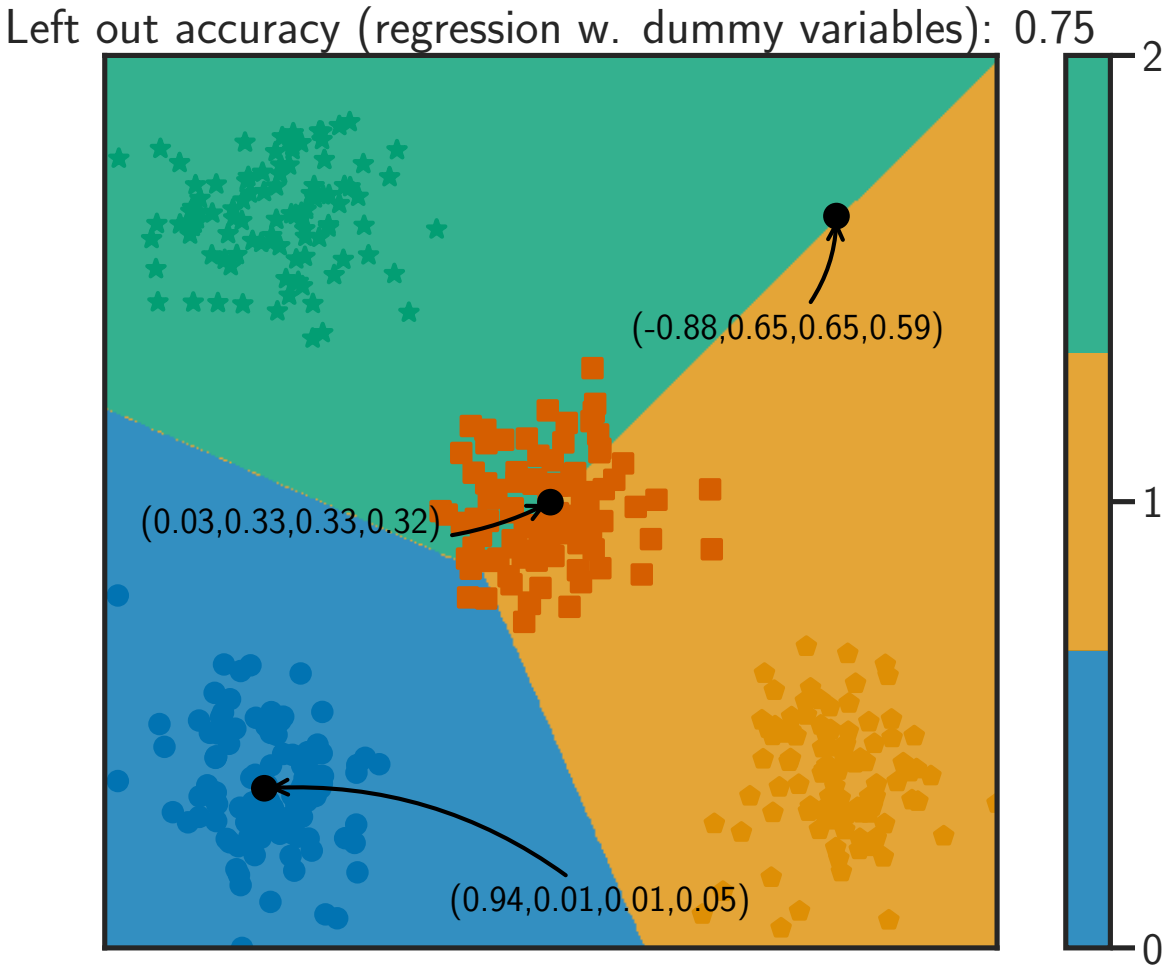
- Ok pour le cas binaire (à condition de la présence d'*outliers*)

Un exemple avec $K = 3$



- Les estimations $\langle x, \widehat{\beta}^{(k)} \rangle$ de $\mathbb{P}(y = k|x)$ peuvent être négatives

Un exemple avec $K = 4$



- Effet masque

Avantages / Inconvénients : prédiction linéaire et indicatrice

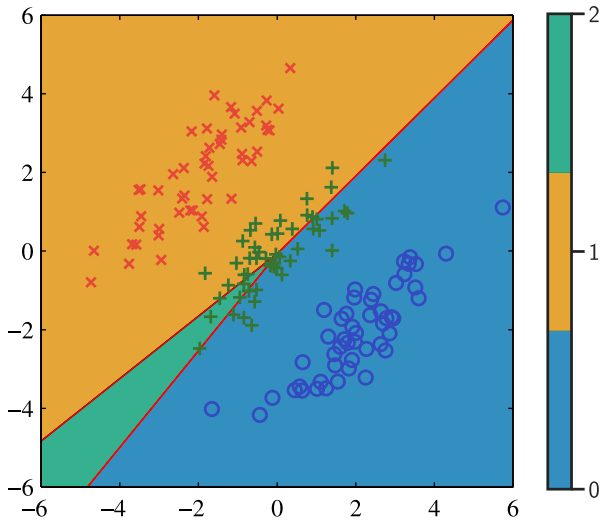
Avantages

- Simplicité : peu d'hypothèse sur le modèle
- Implémentable : facile avec un solveur de moindres carrés
- Réduction de la dimension : règle de discrimination de Fisher (*Fisher's linear discriminant*)

Inconvénients

- Les estimations $\langle x, \widehat{\beta}^{(k)} \rangle$ de $\mathbb{P}(y = k|x)$ peuvent être négatives
- Effet masque

Rem : Ne pas utiliser (sauf peut-être pour le cas binaire)



Régression logistique

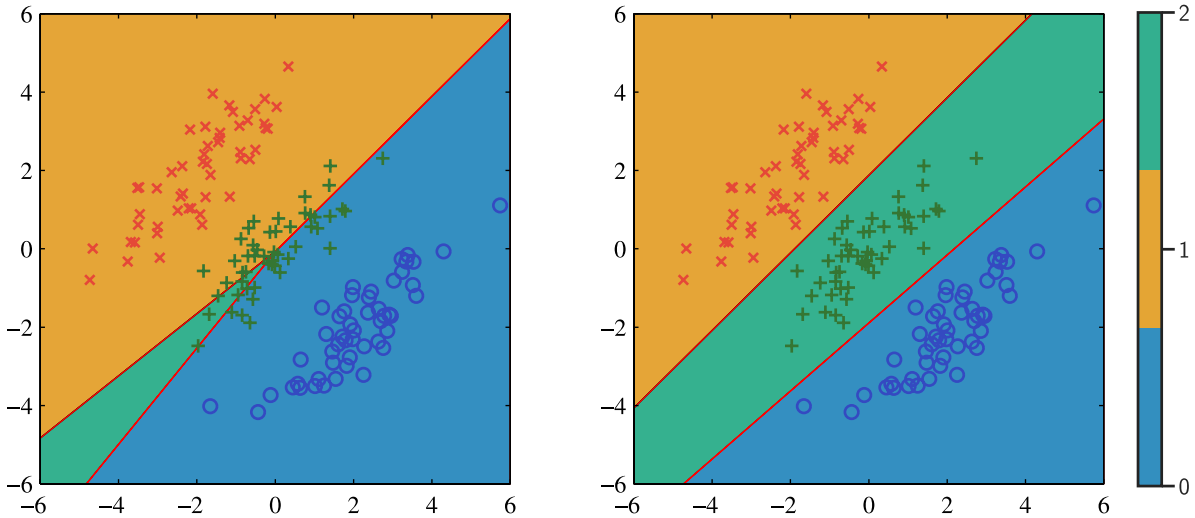
Motivation

· L'échec du classifieur de moindres carrés ne devrait pas surprendre

· L'estimation de paramètres se fait par maximum de vraisemblance sous l'hypothèse d'une distribution conditionnelle gaussienne

Problème : les vecteurs cibles binaires ont une distribution très éloignée de la gaussienne

Solution : adopter d'autres modèles (déterministes ou probabilistes) plus appropriés



Régression linéaire Régression logistique

Motivation

Rappel. Dans le cadre générale, on s'intéresse à estimer $\mathbb{P}(y = k|x)$ pour tout $k \in [\![0, K-1]\!]$

· Jusqu'à présent, on a essayé de les modéliser directement par une règle de discrimination linéaire

· Alternativement, on peut adopter une approche générative dans laquelle on modélise :

- les densités conditionnelles par classe $\mathbb{P}(x|y = k)$
- les probabilités a priori $\mathbb{P}(y = k)$

· Les probabilités a posteriori ses calculent à l'aide du théorème de Bayes

$$\mathbb{P}(y = k|x) = \frac{\mathbb{P}(x|y = k)\mathbb{P}(y = k)}{\mathbb{P}(x)} \tag{9}$$

Le cas gaussien binaire isotrope

· Supposons que les densités des observations des classes $k \in \{0, 1\}$ sont gaussiennes isotropes (de même variance $\sigma^2 > 0$) :

$$f_{X|y=k}(x) = \frac{1}{\sqrt{(2\pi)^d \sigma^n}} \exp\left(-\frac{\|x - \mu_k\|^2}{2\sigma^2}\right), \tag{10}$$

· Supposons également que $\mathbb{P}(y = 0) = \pi_0$

· Le log-ratio $\log\left(\frac{\mathbb{P}(y=1|x)}{\mathbb{P}(y=0|x)}\right)$ s'exprime comme suit **[exercice]** :

$$\begin{aligned}\log\left(\frac{\mathbb{P}(y=1|x)}{\mathbb{P}(y=0|x)}\right) &= \langle x, \frac{\mu_1 - \mu_0}{\sigma^2}\rangle + \frac{\|\mu_0\|^2 - \|\mu_1\|^2}{2\sigma^2} + \log\left(\frac{1-\pi_0}{\pi_0}\right) \\ &= \langle x, \beta\rangle + \beta_0\end{aligned}$$

· Ainsi, il est raisonnable de modéliser le log-ratio linéairement

· La règle de classification est alors :

$$\eta_\theta(x) = \mathbb{1}_{\mathbb{P}(y=1|x) \geq \mathbb{P}(y=0|x)} \quad \Leftrightarrow \quad \eta_\theta(x) = \mathbb{1}_{\langle x, \beta\rangle + \beta_0 \geq 0}$$

Lien avec la régression logistique

· De façon générale, supposons que

$$\log\left(\frac{\mathbb{P}(y=1|x)}{\mathbb{P}(y=0|x)}\right) = \langle x, \beta\rangle, \tag{11}$$

où $x = [1, x_1, \ldots, x_d]^\top \in \mathbb{R}^p$ et $\beta = [\beta_0, \ldots, \beta_d]^\top$

· Il est possible d'établir les probabilités a posteriori comme suit **[exercice]** :

$$\mathbb{P}(y=1|x) = \frac{\exp(\langle x, \beta\rangle)}{1+\exp(\langle x, \beta\rangle)} = \frac{1}{1+\exp(-\langle x, \beta\rangle)}, \tag{12}$$

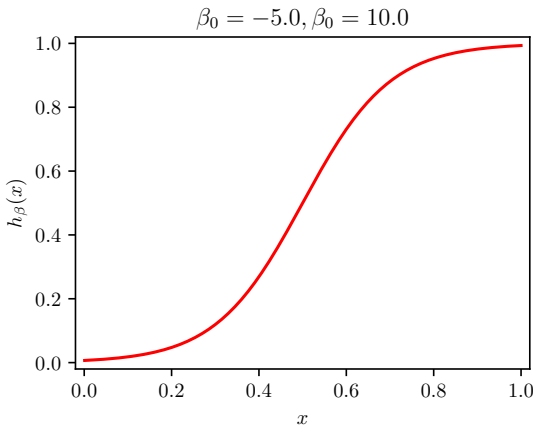
$$\mathbb{P}(y=0|x) = \frac{1}{1+\exp(\langle x, \beta\rangle)} = \frac{\exp(-\langle x, \beta\rangle)}{1+\exp(-\langle x, \beta\rangle)}.$$

Régression logistique binaire et à une seule variable

· Soit $y : \mathbb{R} \to \mathcal{Y}$ avec $\mathcal{Y} = \{0, 1\}$

· On cherche à créer une règle de discrimination dans la classe $\mathcal{F}_{\log}$ de fonctions type

$$h_\beta(x) = \frac{\exp(\beta_0 + \beta_1 x)}{1 + \exp(\beta_0 + \beta_1 x)} = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 x))}, \tag{13}$$



avec $\beta = (\beta_0, \beta_1) \in \mathbb{R}^2$ les paramètres du modèle

· Comme discuté pour le cas gaussien, $h_\beta(x)$ peut être interprété comme la probabilité de $y = 1|X = x$

$$h_\beta(x) := \mathbb{P}(y = 1|X = x)$$

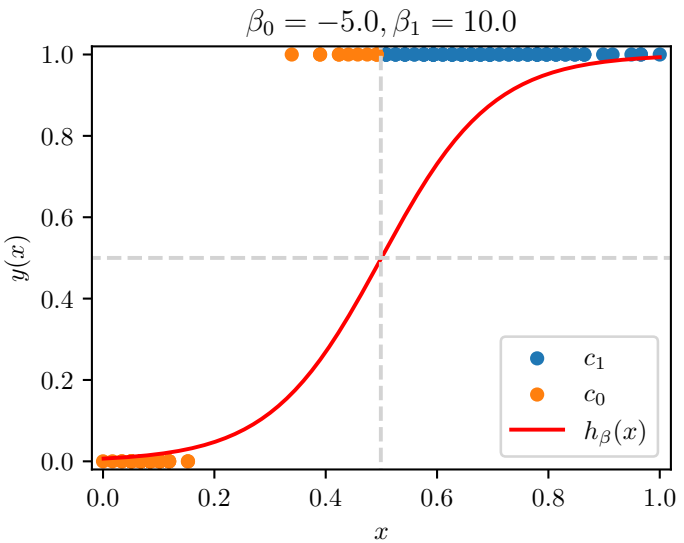
Régression logistique binaire et à une seule variable

· En supposant que la perte liée à une mauvaise classification de chaque classe est la même, alors la règle de discrimination est donnée par

$$\begin{aligned}\eta(x) &= \mathbb{1}_{\mathbb{P}(y=1|X=x)\geq\mathbb{P}(y=0|X=x)} \\ &= \mathbb{1}_{\mathbb{P}(y=1|X=x)\geq 1-\mathbb{P}(y=1|X=x)} \\ &= \mathbb{1}_{h_\beta(x)\geq 1-h_\beta(x)}\end{aligned}$$

· Étant estimé $\beta = (\beta_0, \beta_1)$, on peut ensuite prédire y pour un nouveau $x \in \mathbb{R}$

$$\eta(x) = \mathbb{1}_{h_\beta(x)\geq 0.5}$$



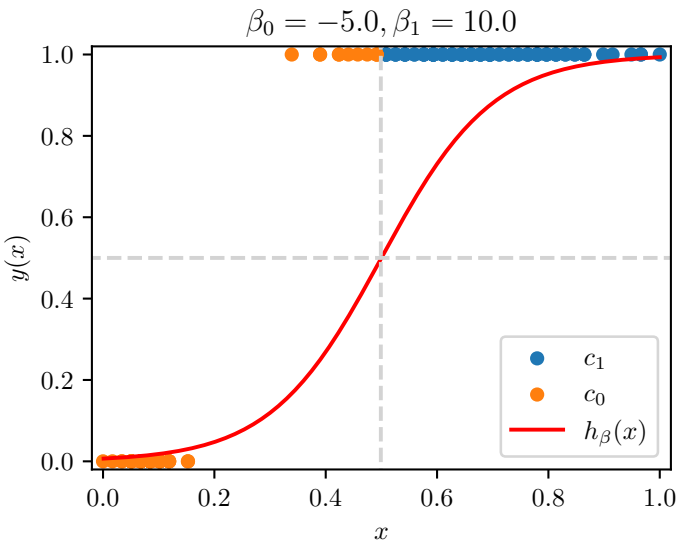
Régression logistique binaire et à une seule variable

· En supposant que la perte liée à une mauvaise classification de chaque classe est la même, alors la règle de discrimination est donnée par

$$\begin{aligned}\eta(x) &= \mathbb{1}_{\mathbb{P}(y=1|X=x)\geq\mathbb{P}(y=0|X=x)} \\ &= \mathbb{1}_{\mathbb{P}(y=1|X=x)\geq 1-\mathbb{P}(y=1|X=x)} \\ &= \mathbb{1}_{h_\beta(x)\geq 1-h_\beta(x)} = \mathbb{1}_{\log\left(\frac{h_\beta(x)}{1-h_\beta(x)}\right)\geq 0} = \mathbb{1}_{\beta_0+\beta_1 x\geq 0}\end{aligned}$$

· Étant estimé $\beta = (\beta_0, \beta_1)$, on peut ensuite prédire y pour un nouveau $x \in \mathbb{R}$

$$\begin{aligned}\eta(x) &= \mathbb{1}_{h_\beta(x)\geq 0.5} \\ &= \mathbb{1}_{\beta_0+\beta_1 x\geq 0}\end{aligned}$$

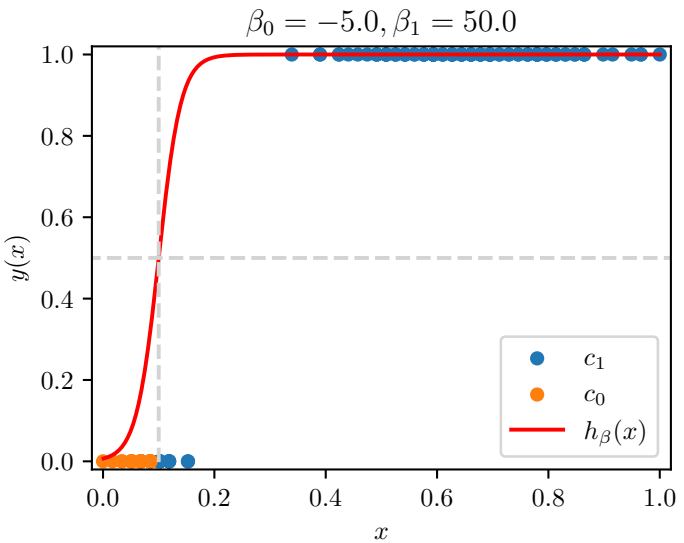


· En supposant que la perte liée à une mauvaise classification de chaque classe est la même, alors la règle de discrimination est donnée par

$$\begin{aligned}\eta(x) &= \mathbb{1}_{\mathbb{P}(y=1|X=x)\geq\mathbb{P}(y=0|X=x)} \\ &= \mathbb{1}_{\mathbb{P}(y=1|X=x)\geq 1-\mathbb{P}(y=1|X=x)} \\ &= \mathbb{1}_{h_\beta(x)\geq 1-h_\beta(x)} = \mathbb{1}_{\log\left(\frac{h_\beta(x)}{1-h_\beta(x)}\right)\geq 0} = \mathbb{1}_{\beta_0+\beta_1 x\geq 0}\end{aligned}$$

· Étant estimé $\beta = (\beta_0, \beta_1)$, on peut ensuite prédire y pour un nouveau $x \in \mathbb{R}$

$$\begin{aligned}\eta(x) &= \mathbb{1}_{h_\beta(x)\geq 0.5} \\ &= \mathbb{1}_{\beta_0+\beta_1 x\geq 0}\end{aligned}$$



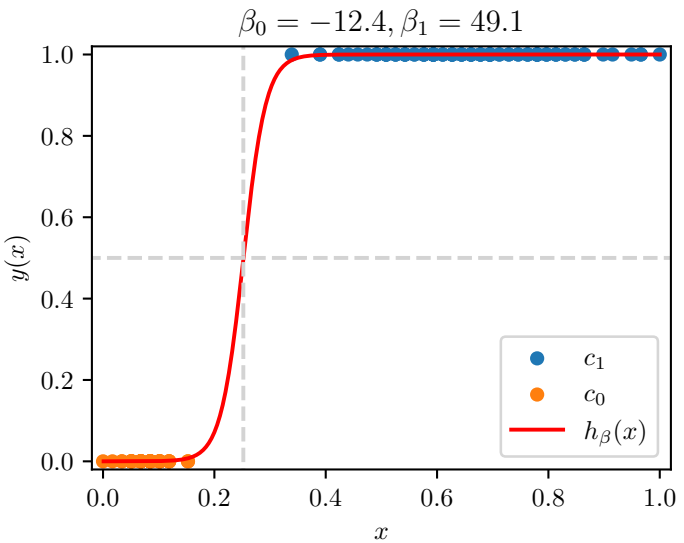
Régression logistique binaire et à une seule variable

· En supposant que la perte liée à une mauvaise classification de chaque classe est la même, alors la règle de discrimination est donnée par

$$\begin{aligned}\eta(x) &= \mathbb{1}_{\mathbb{P}(y=1|X=x)\geq\mathbb{P}(y=0|X=x)} \\ &= \mathbb{1}_{\mathbb{P}(y=1|X=x)\geq 1-\mathbb{P}(y=1|X=x)} \\ &= \mathbb{1}_{h_\beta(x)\geq 1-h_\beta(x)} = \mathbb{1}_{\log\left(\frac{h_\beta(x)}{1-h_\beta(x)}\right)\geq 0} = \mathbb{1}_{\beta_0+\beta_1 x\geq 0}\end{aligned}$$

· Étant estimé $\beta = (\beta_0, \beta_1)$, on peut ensuite prédire y pour un nouveau $x \in \mathbb{R}$

$$\begin{aligned}\eta(x) &= \mathbb{1}_{h_\beta(x)\geq 0.5} \\ &= \mathbb{1}_{\beta_0+\beta_1 x\geq 0}\end{aligned}$$



Généralisation à plusieurs co-variables

· Pour un problème à plusieurs co-variables $x \in \mathbb{R}^d$, on cherche à créer une règle dans la classe $\mathcal{F}_{\log}$ de fonctions type

$$h_\beta(x) = \frac{1}{1+\exp\left(-\left(\beta_0 + \sum_{j=1}^d \beta_j x_j\right)\right)} = \frac{1}{1+\exp\left(-\langle x, \beta\rangle\right)}, \tag{14}$$

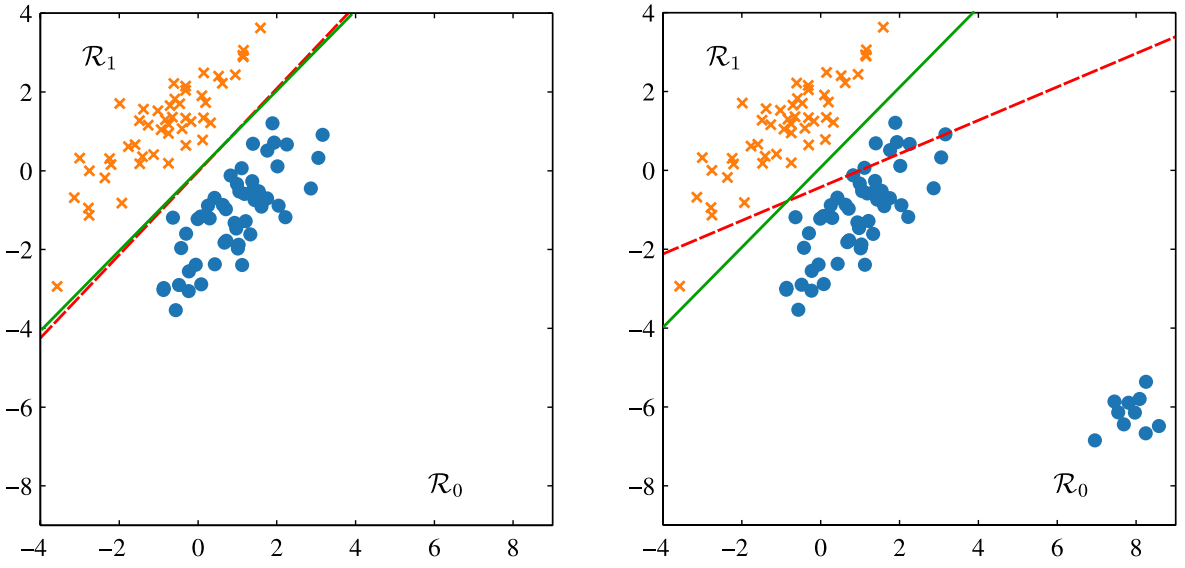
avec $x = (1, x_1, \ldots, x_d) \in \mathbb{R}^p$ et $\beta = (\beta_0, \ldots, \beta_d) \in \mathbb{R}^p$

· La règle de discrimination est donnée par

$$\eta(x) = \mathbb{1}\left(\log\left(\frac{h_\beta(x)}{1-h_\beta(x)}\right) > 0\right) = \mathbb{1}(\langle x, \beta\rangle \geq 0),$$

· On peut en conclure que la surface de décision est à nouveau donnée par un hyperplain en dimension $d-1$

Généralisation à plusieurs co-variables



— Régression linéaire
— Régression logistique

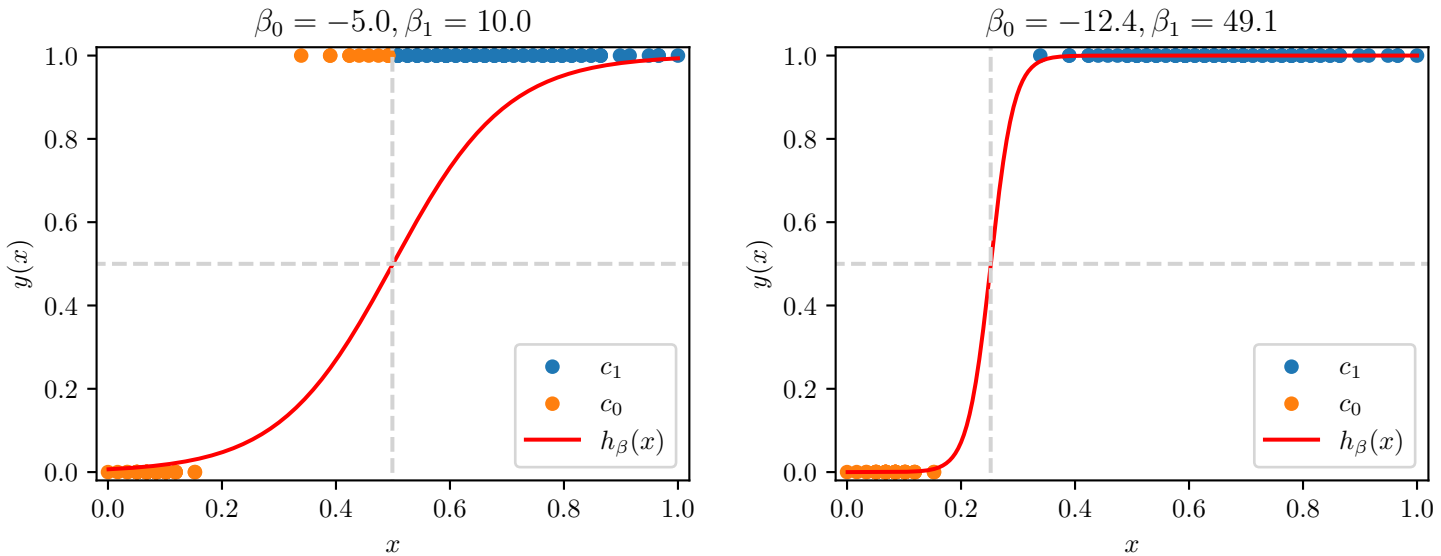
· La régression logistique est plus robuste

Influence des paramètres β

· En général, il y aura une incertitude quant à l'étiquette correcte

· Donc, nous devons prédire une distribution de probabilités sur les étiquettes, et non pas simplement décider de quel côté on se trouve de la limite de décision

- Le vecteur normal β décrit l'orientation de la surface de décision
- La norme $\|\beta\| = \sqrt{\sum_{j=1}^{d} \beta_d^2}$ contrôle la confiance dans les prédictions car elle a une influence sur la pente de la sigmoïde



Maximum de vraisemblance

· Pour estimer β, on cherche à maximiser la log-vraisemblance du modèle :

$$\mathcal{L}(\beta) = \log \prod_{i=1}^{n} \mathbb{P}_\beta(y = y_i | x = x_i) \tag{15}$$
$$= \sum_{i=1}^{n} \sum_{k=0}^{1} \mathbb{1}_{y_i=k} \log \mathbb{P}_\beta(y = k | x = x_i),$$

et donc

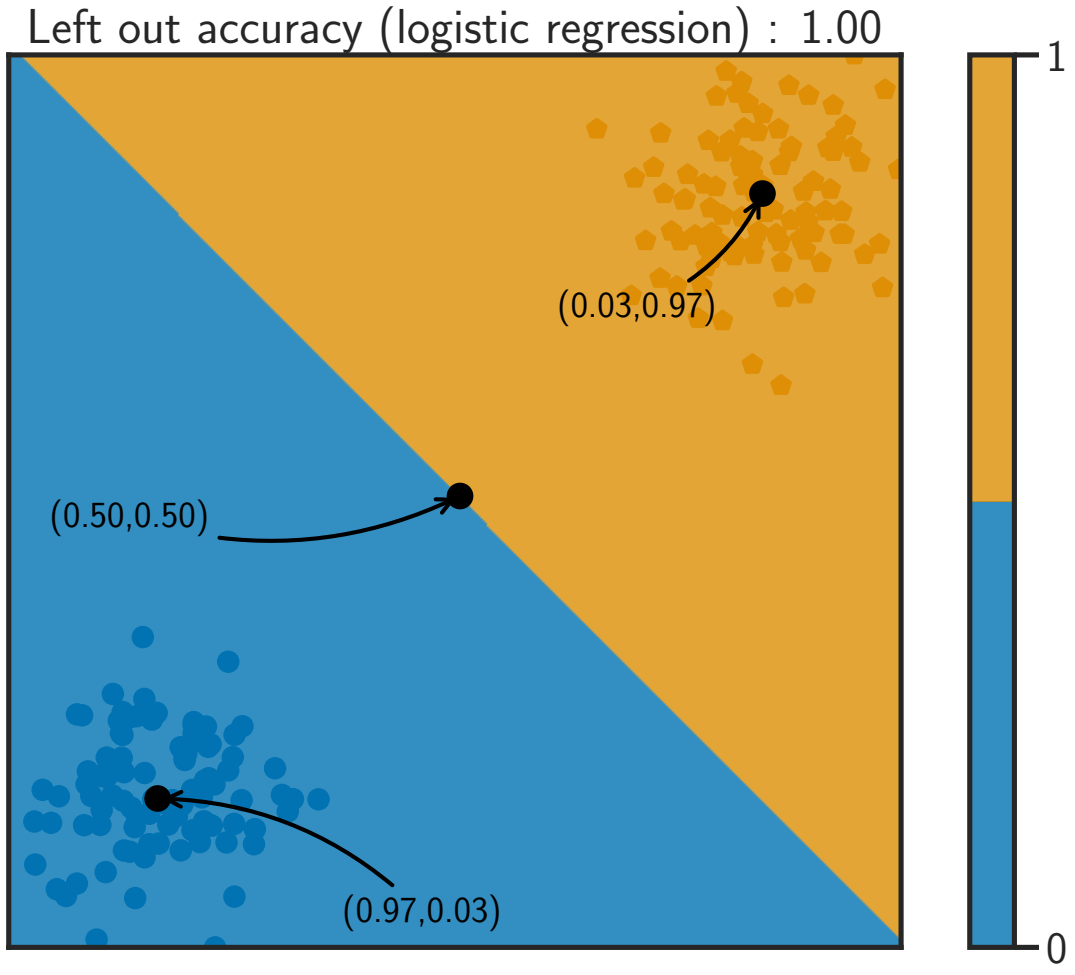
$$\widehat{\beta} = \underset{\beta \in \mathbb{R}^p}{\text{argmax}}\, \mathcal{L}(\beta) \quad \Leftrightarrow \quad \widehat{\beta} = \underset{\beta \in \mathbb{R}^p}{\text{argmin}}\, (-\mathcal{L}(\beta)) \tag{16}$$

· Ce problème d'optimisation n'a pas une solution close, alors un algorithme itératif doit être utilisé [Hastie et al., 2009]

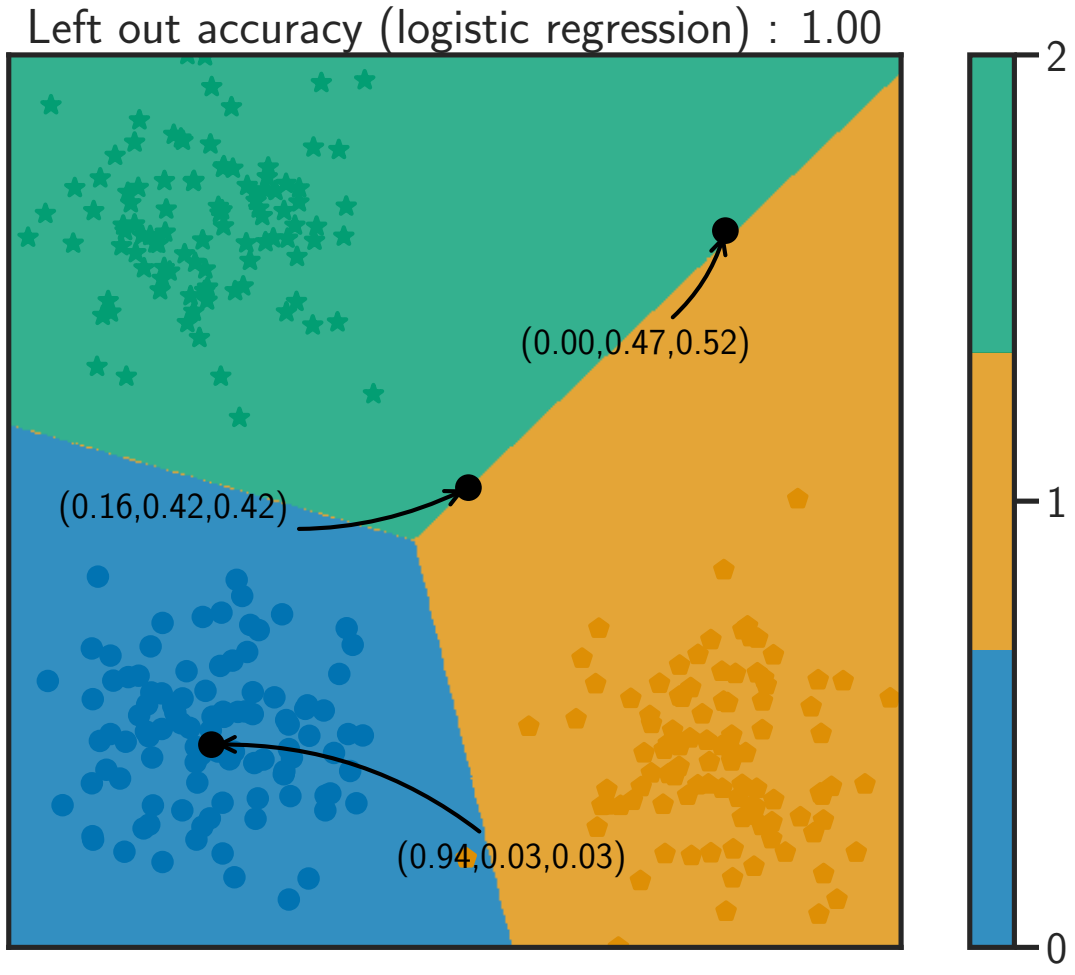
- Approche descente par coordonnées envisageable aussi (notamment si l'on régularise)
- Hessienne calculable : on peut appliquer la méthode de Newton

Packages : *cf.* sklearn en Python ou glm en R

Un exemple avec $K = 2$

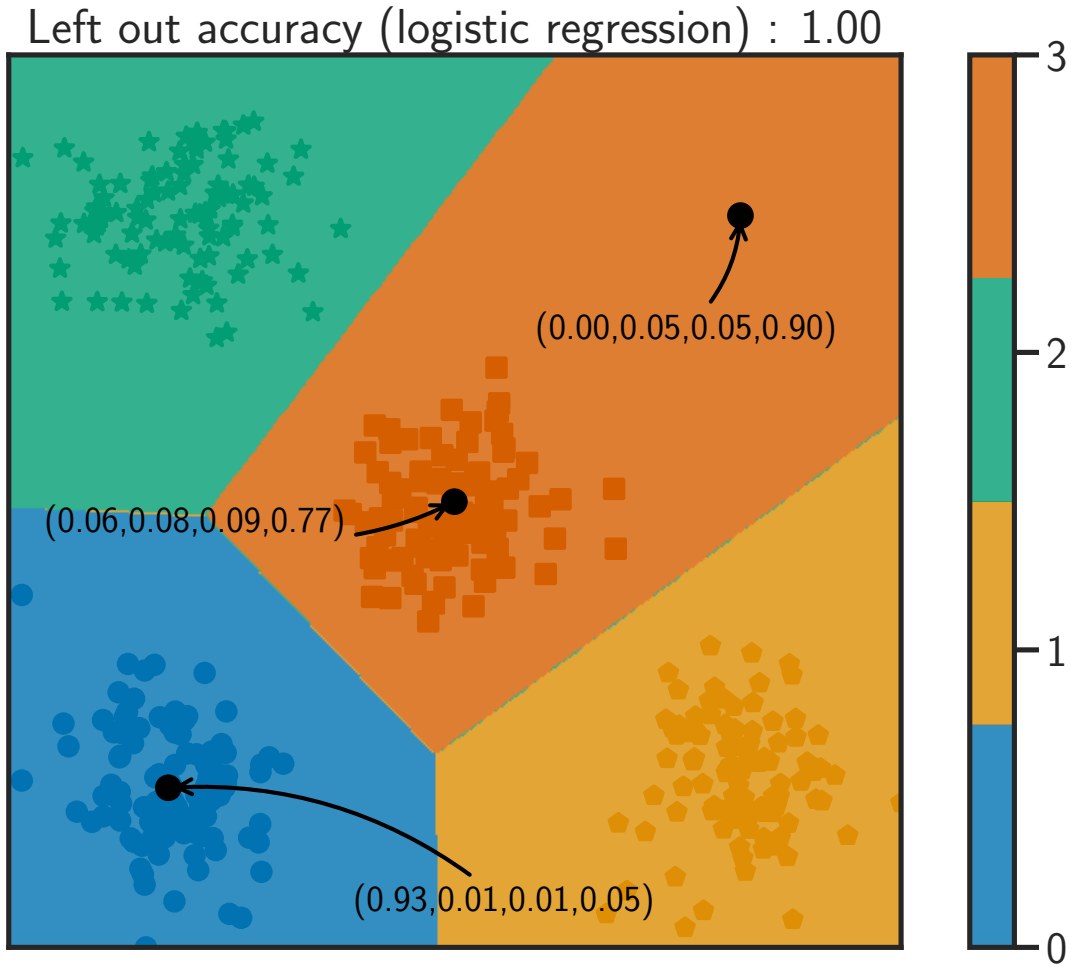


Un exemple avec $K = 3$



· On peut passer du cadre binaire au multi-classe en performant les procédures *one-versus-the-rest* ou *one-versus-one*

Un exemple avec $K = 4$



· On peut passer du cadre binaire au multi-classe en performant les procédures *one-versus-the-rest* ou *one-versus-one*

Régression logistique multi-classe

· Les log-ratios des probabilités conditionnelles des classes se modélisent à nouveau par des quantités linéaires :

$$\log\left(\frac{\mathbb{P}(y=k\,|x)}{\mathbb{P}(y=K-1|x)}\right)=\langle x,\beta^{(k)}\rangle, \text{ pour tout } k\in[\![0,K-2]\!], \beta^{(k)}\in\mathbb{R}^p$$

· On aura alors comme paramètres $B=[\beta^{(0)},\dots,\beta^{(K-1)},\underbrace{\beta^{(K-1)}}_{=0}]\in\mathbb{R}^{p\times K}$

· Sous cette hypothèse les surface de décision inter-classes sont des hyperplans en dimension $d-1$

$$\begin{cases}\text{si } \langle\beta^{(k)}-\beta^{(j)},x\rangle\geq 0, \text{ on préfère la classe } k \text{ à } j \text{ au point } x,\\ \text{sinon on préfère la classe } j \text{ à } k \text{ au point } x\end{cases}$$

pour tout $k,j\in[\![0,K-1]\!]$

Régression logistique multi-classe

· On peut alors estimer les probabilités conditionnelles **[exercice]** :

$$\mathbb{P}(y = k|x) \approx h_{B,k}(x) = \begin{cases} \dfrac{\exp\left(\langle x, \beta^{(k)}\rangle\right)}{1 + \sum_{j=0}^{K-2} \exp\left(\langle x, \beta^{(j)}\rangle\right)}, & \text{si } k \in \{0, \dots, K-2\} \\ \dfrac{1}{1 + \sum_{j=0}^{K-2} \exp\left(\langle x, \beta^{(j)}\rangle\right)}, & \text{si } k = K-1 \end{cases}$$

· Comme règle de classification, choisir la classe qui a la plus probable

$$\eta_B(x) = \underset{k \in [\![0, K-1]\!]}{\operatorname{argmax}} \; h_{B,k}(x)$$

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

Régression logistique multi-classe

Formulation possible : on note $B = [\beta_0, \dots, \beta_{K-1}] \in \mathbb{R}^{p \times K}$

$$\text{Pour } k \in [\![0, K-1]\!] : \ \mathbb{P}(y = k|x) = \frac{\exp\left(\langle x, \beta^{(k)} \rangle\right)}{\sum_{l=0}^{K-1} \exp\left(\langle x, \beta^{(j)} \rangle\right)}$$

Rem : on retrouve le précédent modèle en prenant $\beta^{(K-1)} = 0$

· La (log-)vraisemblance est comme suit

$$\begin{aligned}\mathcal{L}(B) &= \sum_{i=1}^{n} \sum_{k=0}^{K-1} \mathbb{1}_{\{y_i=k\}} \log(\mathbb{P}(y = k|x = x_i)) \\ &= \sum_{i=1}^{n} \sum_{k=0}^{K-1} \mathbb{1}_{\{y_i=k\}} \langle x_i, \beta^{(k)} \rangle - \log\left(\sum_{k=0}^{K-1} \exp(\langle x_i, \beta^{(k)} \rangle)\right)\end{aligned}$$

· On résout enfin

$$\widehat{B} \in \underset{B \in \mathbb{R}^{p \times K}}{\operatorname{argmin}} \left(-\mathcal{L}(B)\right)$$

Astuce numérique : régulariser le problème avec une pénalisation ℓ_1, ℓ_2

Cas de test - Iris (sklearn)

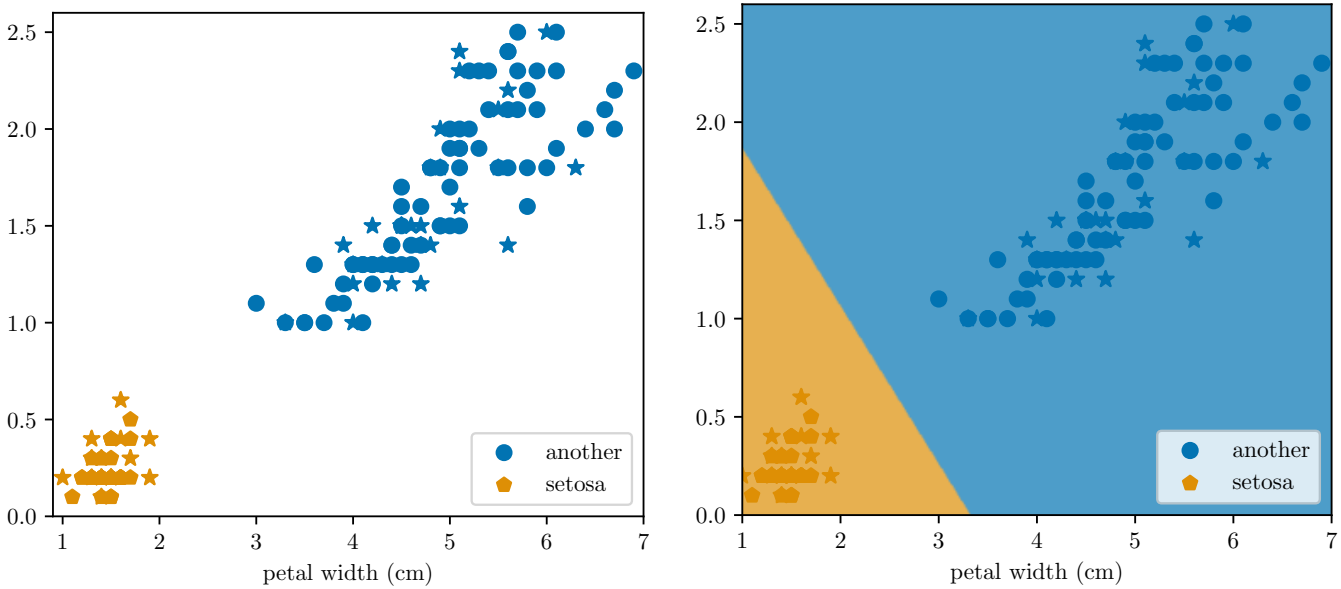
· Considérons :

- 2 caractéristiques :

$$(x_1, x_2) = (\text{longueur du sépale}, \text{largeur du pétale})$$

- 2 classes :

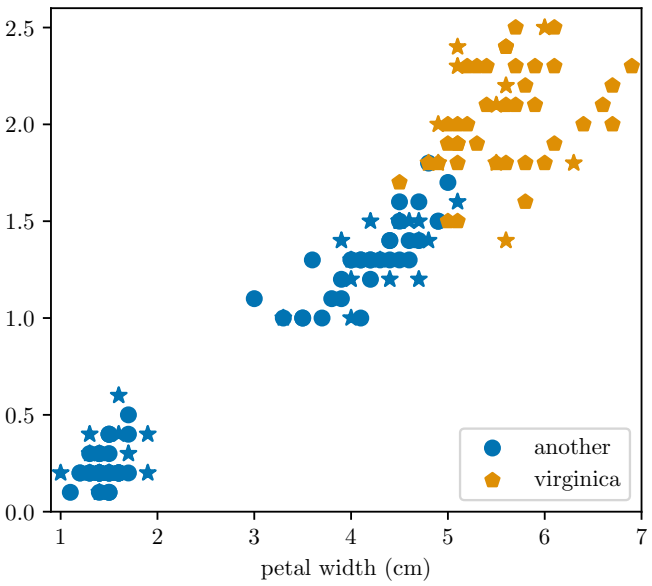
$$\kappa \in \{\text{Setosa}, \text{autre}\}$$

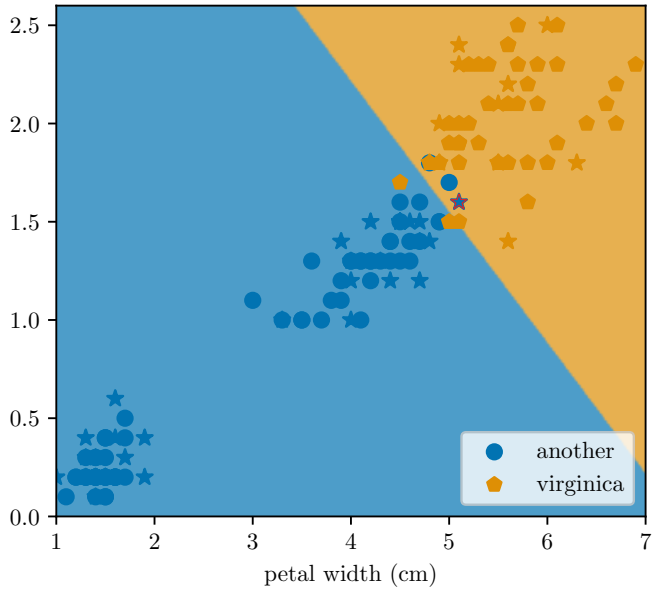


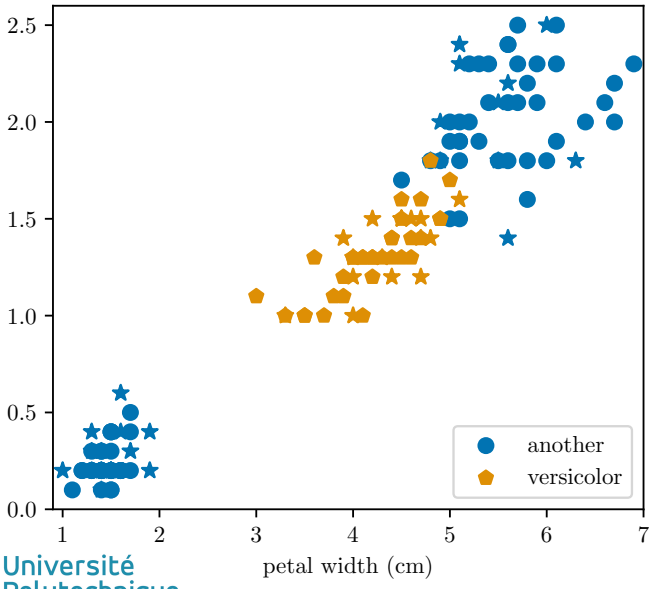
● : Données d'apprentissage

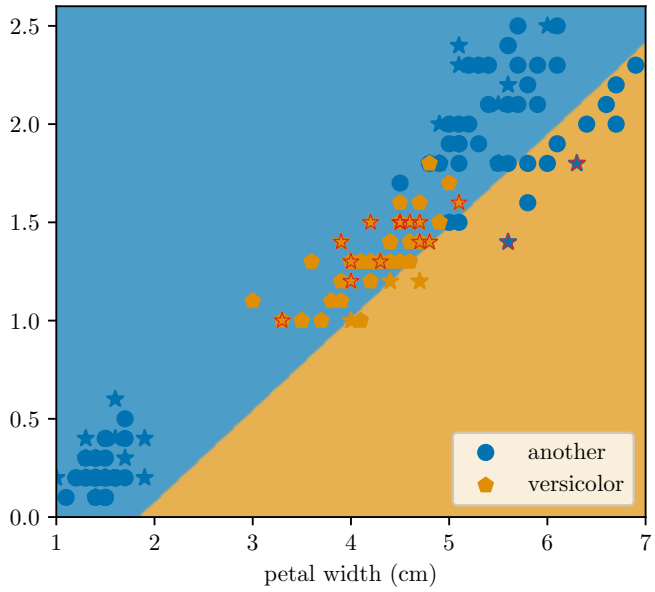
⋆ : Données à prédire

Cas de test - Iris (sklearn)





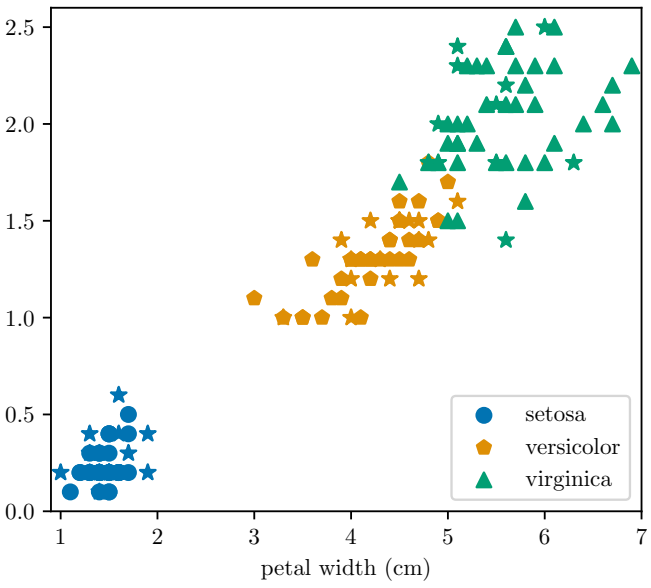


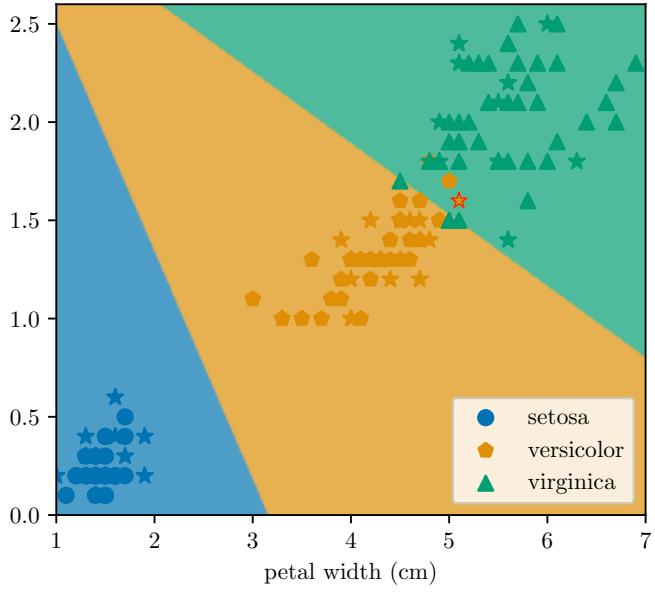


CERAMATHS | Université Polytechnique HAUTS-DE-FRANCE

● : Données d'apprentissage

Cas de test - Iris (sklearn)





● : Données d'apprentissage

⋆ : Données à prédire

Avantages / Inconvénients : régression logistique

Avantages

- Classification binaire plus facile
- Probabilités estimées bonnes
- Séparations inter-classes linéaires

Inconvénients

- Problème d'optimisation plus complexe (temps de calcul)
- Parfois géré par la technique du "un contre tous" et non par le cas logistique multinomial (surtout si K est petit)

Analyse discriminante linéaire

Le cas gaussien binaire isotrope

· Revenons au cas où les densités des observations des classes $k \in \{0, 1\}$ sont gaussiennes isotropes (de même variance $\sigma^2 > 0$) :

$$f_{X|y=k}(x) = \frac{1}{\sqrt{(2\pi)^d \sigma^n}} \exp\left(-\frac{\|x - \mu_k\|^2}{2\sigma^2}\right), \tag{17}$$

· Supposons $\mathbb{P}(y = 0) = \pi_0$ et $\mathbb{P}(y = 1) = \pi_1 = 1 - \pi_0$

· Le log-ratio $\log\left(\frac{\mathbb{P}(y=1|x)}{\mathbb{P}(y=0|x)}\right)$ s'exprime comme suit :

$$\log\left(\frac{\mathbb{P}(y=1|x)}{\mathbb{P}(y=0|x)}\right) = \langle x, \frac{\mu_1 - \mu_0}{\sigma^2}\rangle + \frac{\|\mu_0\|^2 - \|\mu_1\|^2}{2\sigma^2} + \log\left(\frac{\pi_1}{\pi_0}\right)$$

· En dénotant $\delta_k(x) = \langle x, \frac{\mu_k}{\sigma^2}\rangle - \frac{\|\mu_k\|^2}{2\sigma^2} + \log(\pi_k)$ pour $k \in \{0, 1\}$, on obtient

$$\log\left(\frac{\mathbb{P}(y=1|x)}{\mathbb{P}(y=0|x)}\right) = \delta_1(x) - \delta_0(x) = \langle x, \beta\rangle + \beta_0,$$

avec

$$\beta = \frac{\mu_1 - \mu_0}{\sigma^2}, \quad \beta_0 = \frac{\|\mu_0\|^2 - \|\mu_1\|^2}{2\sigma^2} + \log\left(\frac{\pi_1}{\pi_0}\right)$$

Analyse discriminante linéaire (*linear discriminant analysis, LDA*)

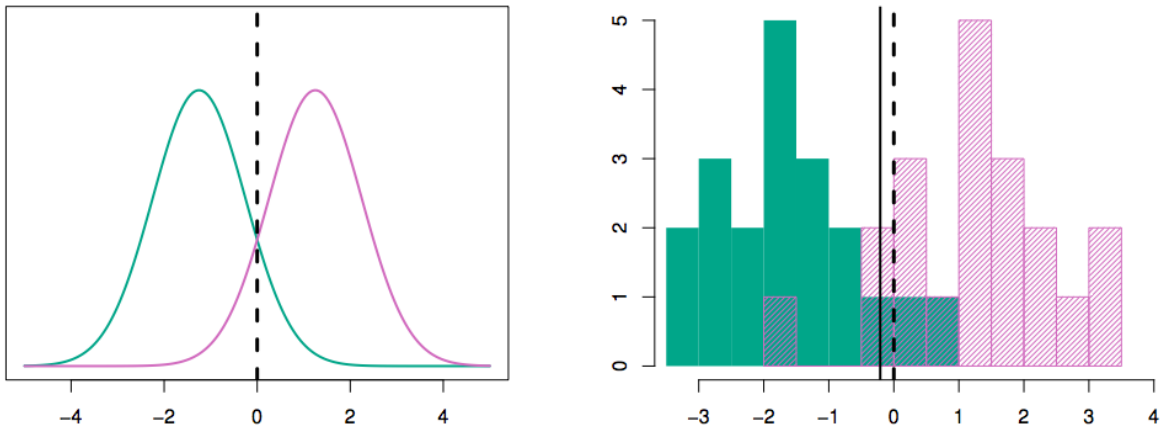
· LDA suppose que les densités des observations de chaque classe sont gaussiennes isotropes

· Dans ce cas, pour la règle de discrimination, il suffit alors de comparer les fonctions linéaires δ_k :

$$\eta_\theta(x) = \mathbb{1}_{\mathbb{P}(y=1|x) \geq \mathbb{P}(y=0|x)} \quad \Leftrightarrow \quad \eta_\theta(x) = \mathbb{1}_{\delta_1(x) \geq \delta_0(x)}$$

· Les fonctions δ_k s'appellent les scores de Fisher

· Notons que, pour le cas où $\pi_0 = \pi_1 = 0.5$, on peut vérifier que la surface de décision (linéaire) est en $x = \frac{1}{2}(\mu_0 + \mu_1)$ **[exercice]**



Exemple avec $\mu_1 = -\mu_0 = 1.5$ et $\sigma^2 = 1$

Analyse discriminante linéaire à plusieurs co-variables

· Les densités des observations des classes $k \in \{0, 1\}$ sont gaussiennes de même covariance $\Sigma \in \mathbb{R}^{d \times d}$:

$$f_{X|y=k}(x) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \exp\left(-\frac{1}{2}(x - \mu_k)^\top \Sigma^{-1}(x - \mu_k)\right), \tag{18}$$

· On suppose $\mathbb{P}(y = 0) = \pi_0$ et $\mathbb{P}(y = 1) = \pi_1 = 1 - \pi_0$

· Les scores de Fisher δ_k deviennent

$$\begin{aligned}\delta_k(x) &= \langle x, \Sigma^{-1}\mu_k \rangle - \frac{1}{2}\mu_k^\top \Sigma^{-1}\mu_k + \log(\pi_k) \\ &= \langle x, \Sigma^{-1}\mu_k \rangle - \frac{1}{2}\|\mu_k\|^2_{\Sigma^{-1}} + \log(\pi_k),\end{aligned}$$

qui est toujours une fonction linéaire en x

· Enfin

$$\log\left(\frac{\mathbb{P}(y = 1|x)}{\mathbb{P}(y = 0|x)}\right) = \delta_1(x) - \delta_0(x) = \langle x, \beta \rangle + \beta_0,$$

avec

$$\beta = \Sigma^{-1}(\mu_1 - \mu_0), \quad \beta_0 = \frac{1}{2}(\|\mu_0\|^2_{\Sigma^{-1}} - \|\mu_1\|^2_{\Sigma^{-1}}) + \log\left(\frac{\pi_1}{\pi_0}\right)$$

Estimateurs des paramètres

· Dans le cas $d = 1$, on a comme estimateurs **[exercice]**

$$\widehat{\pi}_k = \frac{n_k}{n}, \quad \widehat{\mu}_k = \sum_{i:y_i=k} \frac{x_i}{n_k},$$

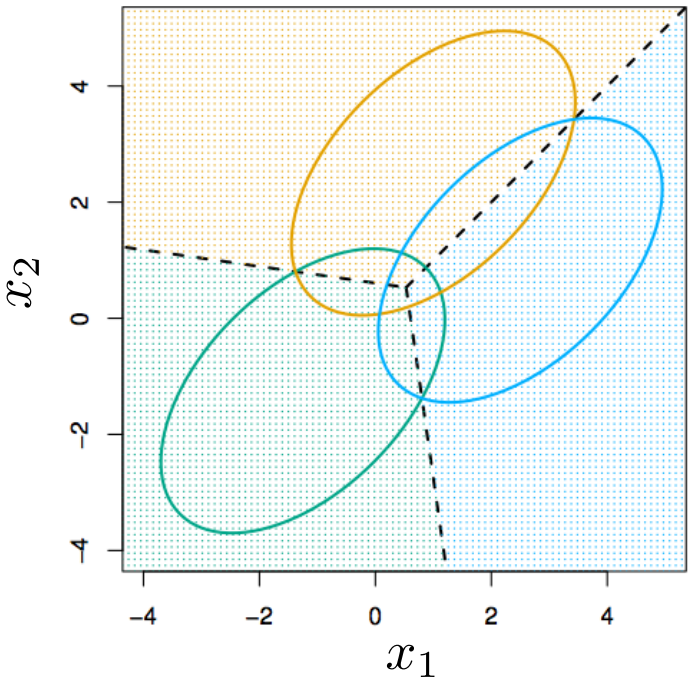
$$\widehat{\sigma}_k^2 = \sum_{i:y_i=k} \frac{(x_i - \widehat{\mu}_k)^2}{n_k - 1}, \quad \widehat{\sigma}^2 = \sum_{k=0}^{K-1} \frac{n_k - 1}{n - K} \widehat{\sigma}_k^2$$

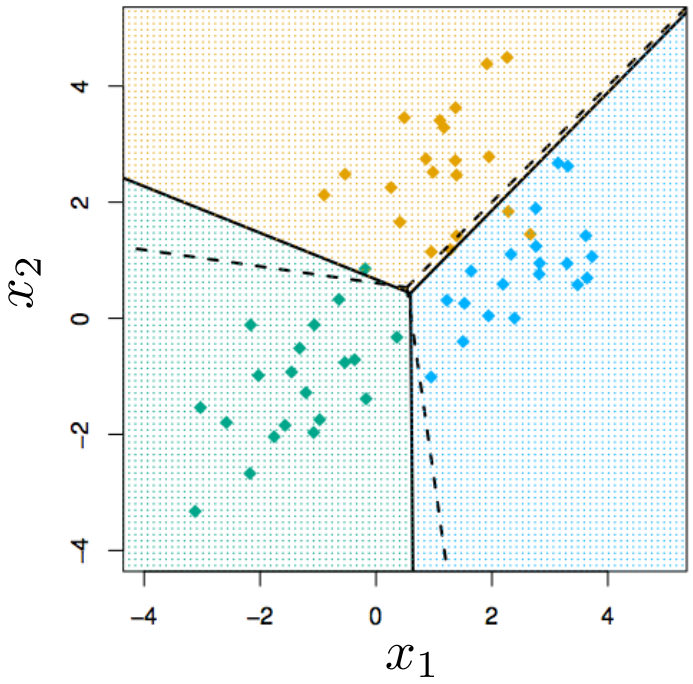
· Dans le cas $d > 1$, on a comme estimateurs **[exercice]**

$$\widehat{\pi}_k = \frac{n_k}{n}, \quad \widehat{\mu}_k = \sum_{i:y_i=k} \frac{x_i}{n_k},$$

$$\widehat{\Sigma}_k = \sum_{i:y_i=k} \frac{(x_i - \widehat{\mu}_k)(x_i - \widehat{\mu}_k)^\top}{n_k - 1}, \quad \widehat{\Sigma} = \sum_{k=0}^{K-1} \frac{n_k - 1}{n - K} \widehat{\Sigma}_k$$

Exemple simulé en dimension 2



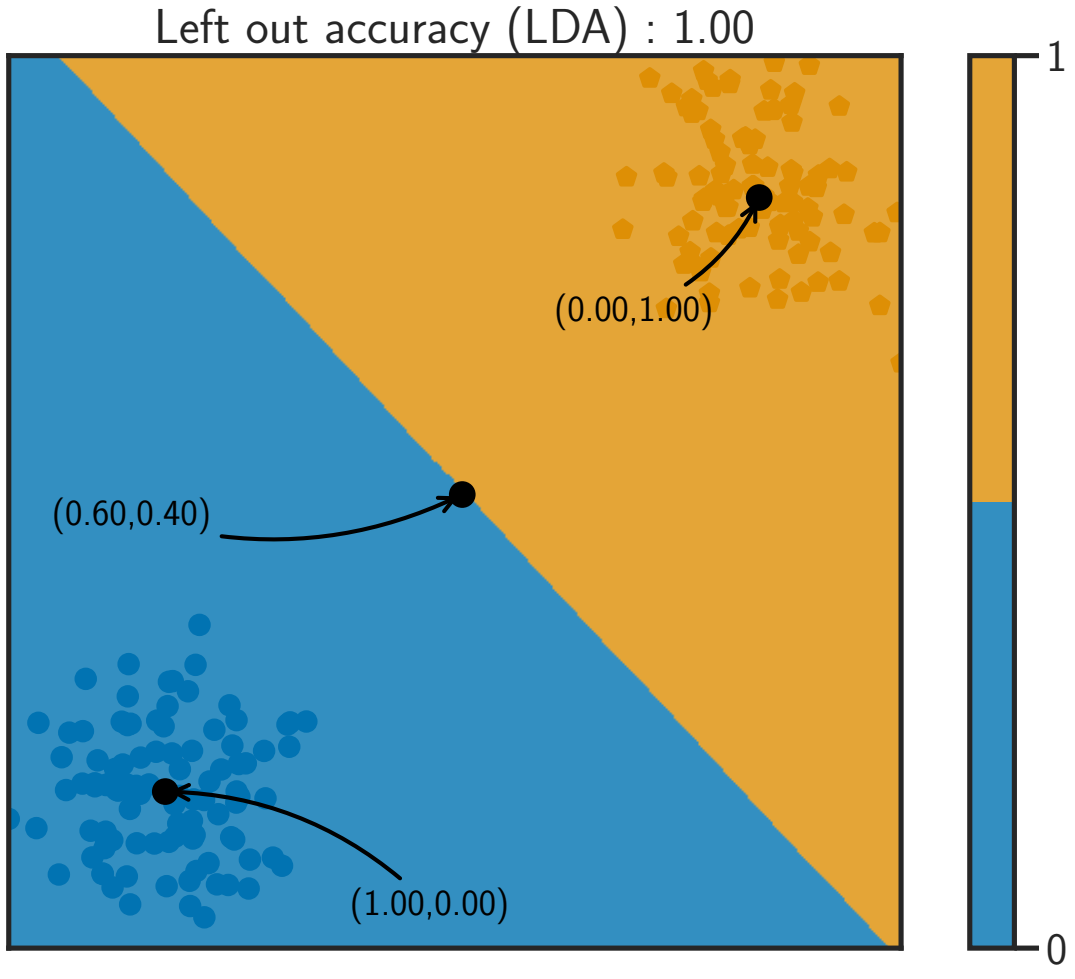


lignes pointillées : frontières théoriques
lignes pleines : frontières estimées par LDA

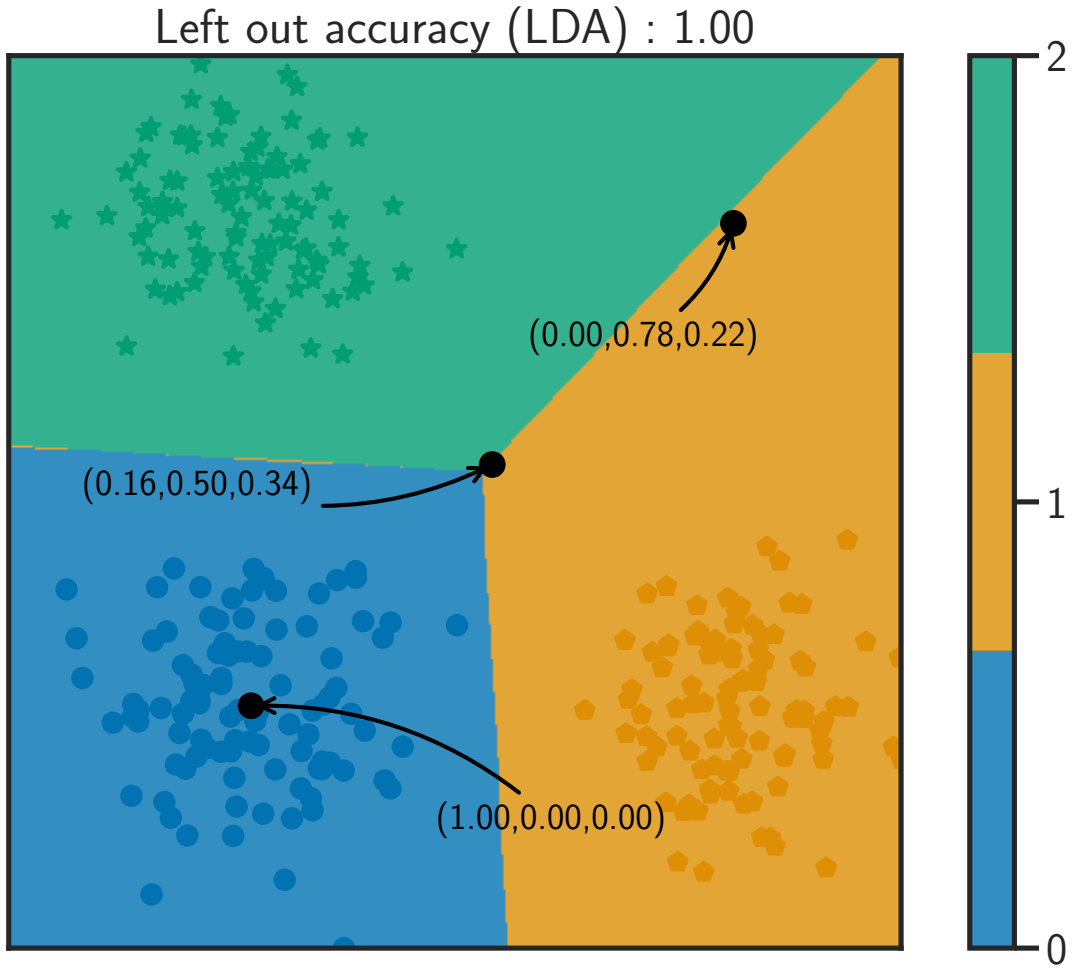
Exemple avec $\pi_k = \frac{1}{3}$ pour tout $k \in \{0, 1, 2\}$

[Hastie et al., 2009]

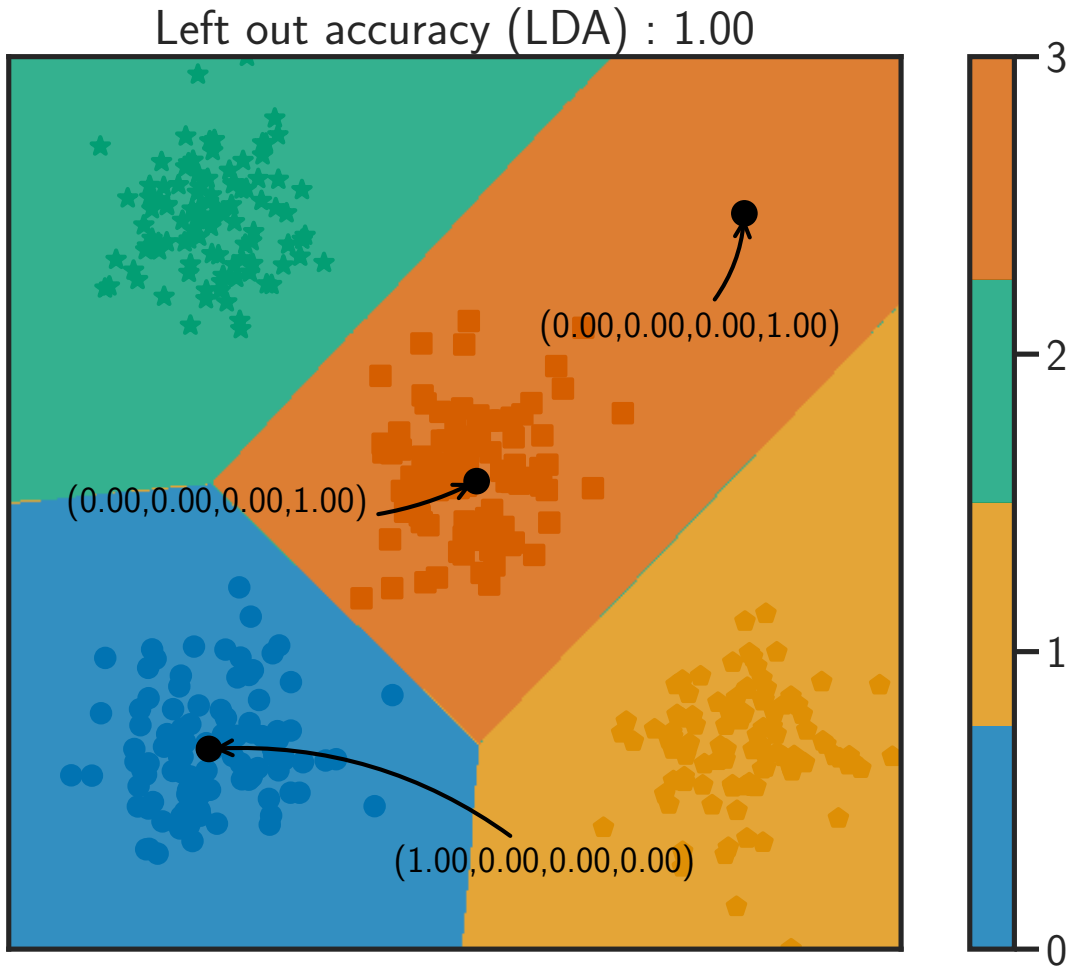
LDA : exemple avec $K = 2$



LDA : exemple avec $K = 3$



LDA : exemple avec $K = 4$



Pourquoi utiliser la LDA

- LDA est aussi populaire lorsqu'il y a plus de deux modalités pour y car elle permet de projeter les données dans des hyperplans séparent les groupes
- Lorsque les classes sont bien séparées, l'estimation des paramètres de la régression logistique devient instable. LDA ne souffre pas de ce problème
- Lorsque n est petit et la distribution de $\mathbb{P}(x|y = k)$ est à peu près gaussienne dans chaque classe, LDA est plus stable que la régression logistique

Autres formes d'analyse discriminante

Rappel. Les probabilités a posteriori ses calculent à l'aide de Bayes

$$\mathbb{P}(y = k|x) = \frac{\mathbb{P}(x|y = k)\mathbb{P}(y = k)}{\mathbb{P}(x)} \propto \pi_k \, f_{X|y=k}(x)$$

· En changeant la forme des densités des observations des classes $f_{X|y=k}(x)$, on obtient d'autres types de classifieurs

- Analyse discriminante quadratique (*quadratic discriminant analysis, QDA*) : des gaussiennes avec variances distinctes

$$f_{X|y=k}(x) = \frac{1}{\sqrt{(2\pi)^p |\Sigma_k|}} \exp\left(-\frac{1}{2}(x - \mu_k)^\top {\Sigma_k}^{-1}(x - \mu_k)\right),$$

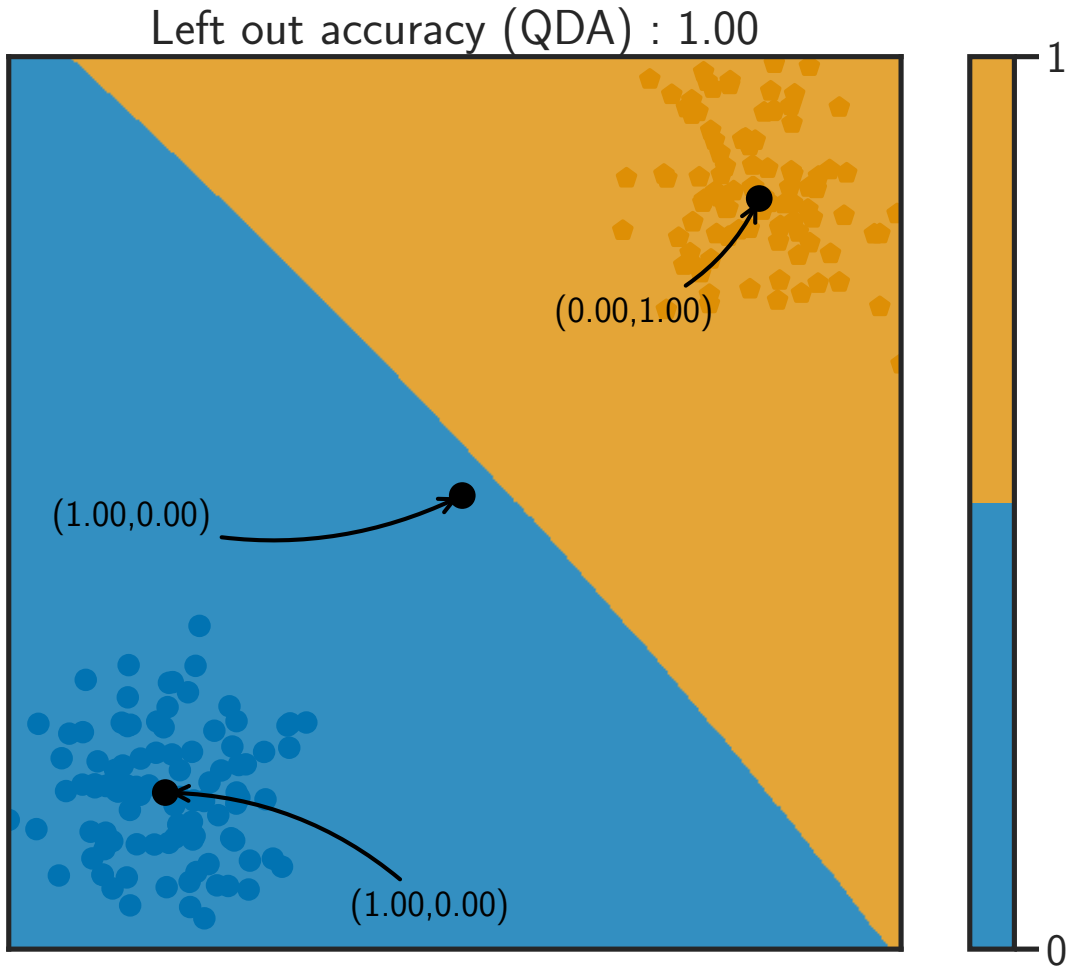
$$\delta_k(x) = -\frac{1}{2}(x - \mu_k)^\top {\Sigma_k}^{-1}(x - \mu_k) - \frac{1}{2}\log|\Sigma_k| + \log(\pi_k)$$

- Naïve Bayes : des densités d'observations qui se factorisent par co-variables

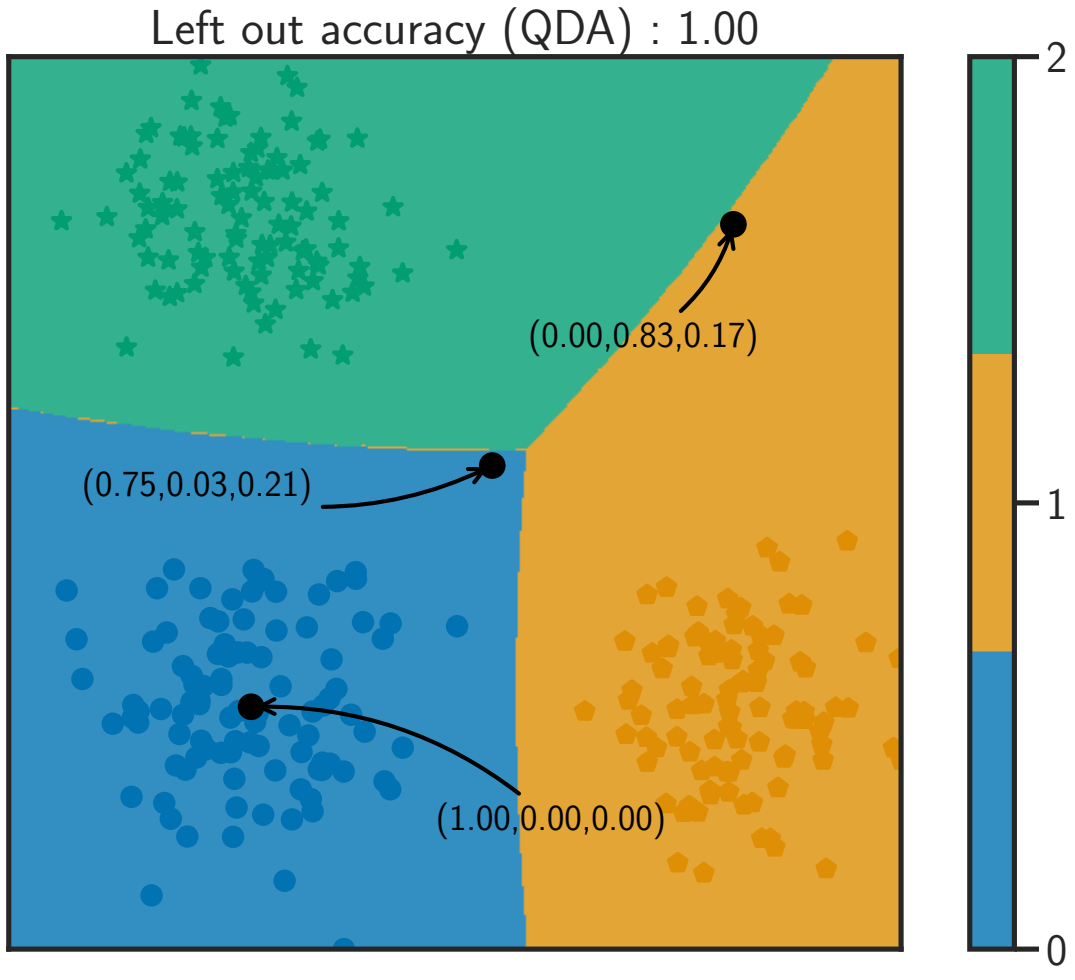
$$f_{X|y=k}(x) = \prod_{j=1}^{q} f_{X|y=k}(x_j),$$

$$\delta_k(x) = -\frac{1}{2}\sum_{j=1}^{p} \frac{(x_j - \mu_{k,j})^2}{\sigma_{k,j}^2} + \log(\pi_k)$$

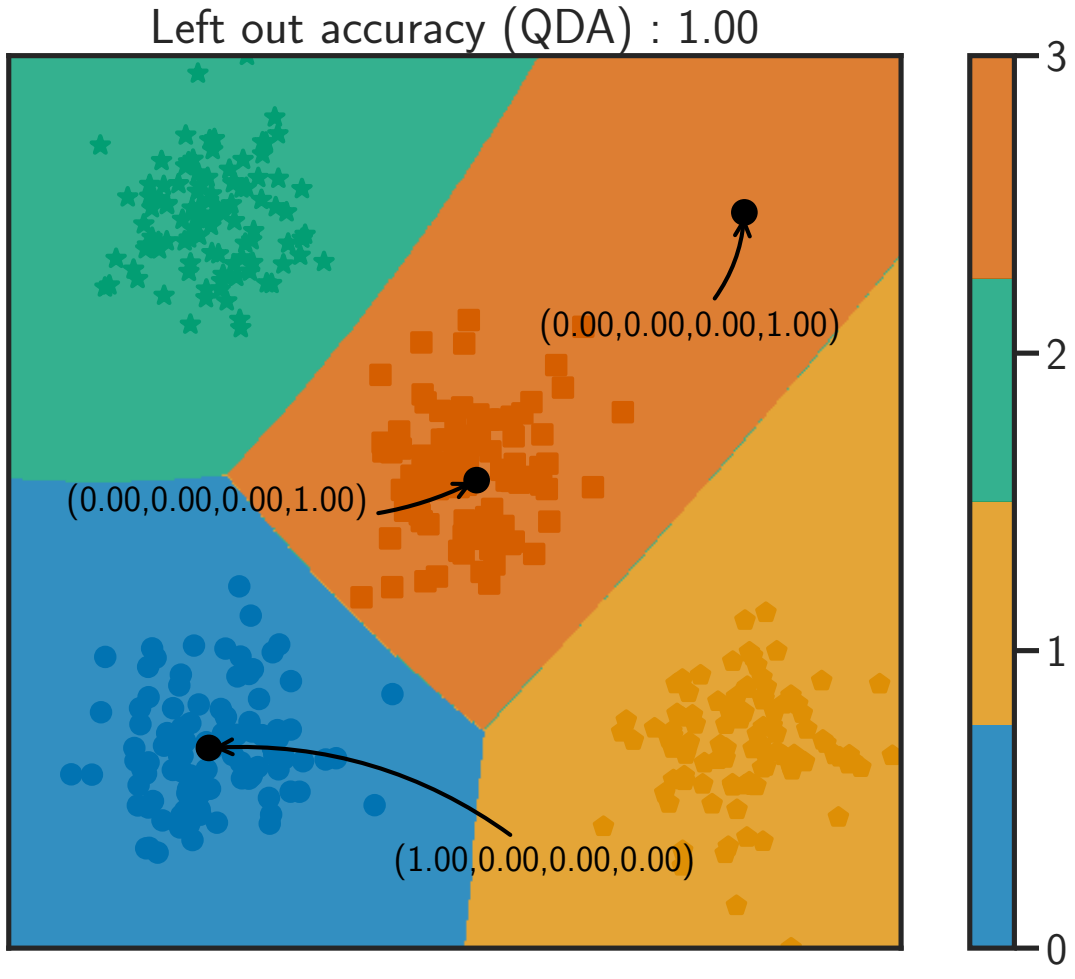
QDA : exemple avec $K = 2$



QDA : exemple avec $K = 3$



QDA : exemple avec $K = 4$



Conclusions

To take away...

Régression logistique

- Populaire pour la classification, surtout lorsqu'il n'y a que $K = 2$ classes

LDA

- Utile lorsque l'hypothèse gaussienne est raisonnable
- Lorsque $K > 2$, même lorsque n est petit
- Lorsque les classes sont bien séparées

Naïve Bayes

- Utile lorsque p est grand

A approfondir...

- En petite dimension ($p < 4$), et avec beaucoup d'observations, on peut faire de l'analyse discriminante non-paramétrique, en remplaçant les gaussiennes par des densités intra-classes estimées (*e.g.*, par une méthode à noyau)
- Autres méthodes (SVM, random forest, k plus proches voisins, etc) dans les cours suivants

Références

Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer, New York, second edition, 2009.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.