Introduction à l'apprentissage automatique

Séance 3

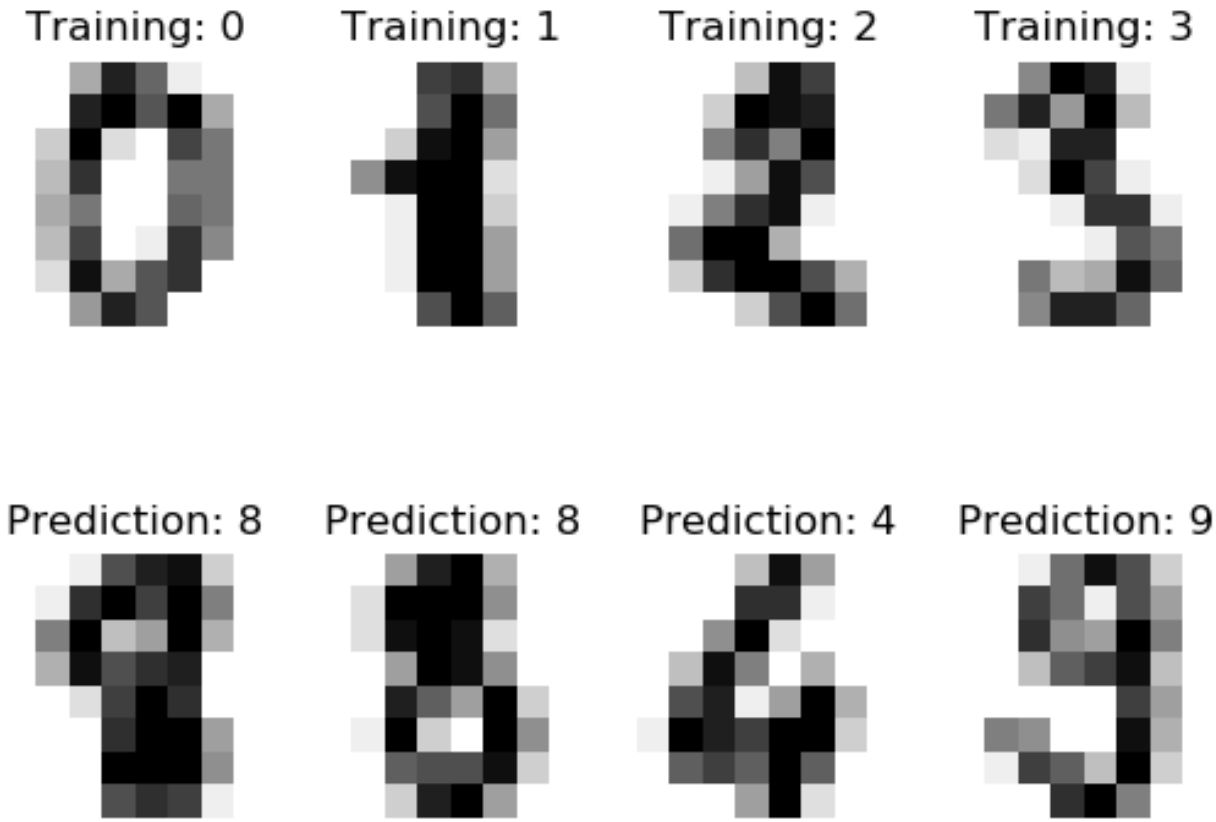
Théorie statistique de la décision et applications

Frédéric Sur

Plan

1. Classification et décision statistique
   - Éléments de théorie statistique de la décision
   - Le ≪ meilleur ≫ classifieur : classifieur de Bayes
2. Mise en œuvre du classifieur de Bayes
   - Classifieur naïf de Bayes
   - Classification aux plus proches voisins
   - Régression logistique
   - GMM et partitionnement
3. Point méthodologique : apprentissage, validation, test
4. Conclusion

Classification supervisée



Aujourd'hui : classification supervisée

Cadre : théorie statistique de la décision

Notations

K classes : $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K$
classes = partition de l'ensemble des observations possibles
Exemple : $\mathcal{C}_1$ =1, $\mathcal{C}_2$ =2, ..., $\mathcal{C}_{10}$ =0

Base d'apprentissage : N observations $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathbb{R}^d$
(Ex. : $\boldsymbol{x} \in \mathbb{R}^{256}$ = vecteur des niveaux de gris d'une imagette 16×16)
étiquetées $y_1, \ldots, y_N \in \{\mathcal{C}_1, \ldots, \mathcal{C}_K\}$

Question du jour : prédiction de la classe d'une nouvelle observation $\boldsymbol{x}$?

Soit f un classifieur : $f(\boldsymbol{x}) = \mathcal{C}_1$ ou $f(\boldsymbol{x}) = \mathcal{C}_2$, etc.
$\rightarrow$ f est une fonction sur l'ensemble des entrées possibles qui prédit la classe d'appartenance

Problème : le classifieur peut faire des erreurs
Exemple : $f(\boldsymbol{x}) = \mathcal{C}_1$ alors que $y = \mathcal{C}_2$

Formalisation probabiliste

Probabilités :

- $p(\mathcal{C}_k)$ *probabilité a priori (prior)*, $\sum_k p(\mathcal{C}_k) = 1$

$\rightarrow$ ce qu'on suppose sans connaître d'observations

Exemple OCR : $p(\text{a}) = 0.07$, $p(\text{b}) = 0.01$, $p(\text{c}) = 0.03\ldots$

- $p(\boldsymbol{x}|\mathcal{C}_k)$ *proba. conditionnelle* (en fait, *densité* de la proba - *vraisemblance*)

$\rightarrow$ probabilité pour qu'une obs. tirée dans la classe $\mathcal{C}_k$ vaille $\boldsymbol{x}$

Exemple : $p(\boldsymbol{x}|\mathcal{C}_k) = \frac{1}{(2\pi)^{d/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(\boldsymbol{x}-\mu_k)^T \Sigma_k^{-1}(\boldsymbol{x}-\mu_k)}$

- $p(\mathcal{C}_k|\boldsymbol{x})$ *probabilité (vraisemblance) a posteriori*

$\rightarrow$ probabilité de la classe $\mathcal{C}_k$ étant donnée l'observation $\boldsymbol{x}$

Théorème de Bayes :

$$\forall\, 1 \leq k \leq K,\ p(\mathcal{C}_k|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\mathcal{C}_k)p(\mathcal{C}_k)}{p(\boldsymbol{x})}$$

$$\text{et } p(\boldsymbol{x}) = \sum_{k=1}^{K} p(\boldsymbol{x}, \mathcal{C}_k) = \sum_{k=1}^{K} p(\boldsymbol{x}|\mathcal{C}_k)p(\mathcal{C}_k)$$

Erreur de classification

Pour simplifier, $K = 2$ dans la suite

Le classifieur f définit une partition de l'ensemble des observations possibles en régions $\mathcal{R}_i$ telles que :

$$\mathcal{R}_i = \{\boldsymbol{x}, f(\boldsymbol{x}) = \mathcal{C}_i\}$$

Problème : les partitions $(\mathcal{C}_i)$ et $(\mathcal{R}_i)$ ne coïncident pas
(à cause des erreurs de classification)

Proposition : calcul de la proportion moyenne d'erreur **théorique**

$$\begin{aligned} E_{\text{err}} &= E_{\mathbf{X},Y}\left(1_{f(\boldsymbol{x})\neq y}\right) = \iint 1_{f(\boldsymbol{x})\neq y} p(\boldsymbol{x}, y)\mathrm{d}\boldsymbol{x}\mathrm{d}y \\ &= \int 1_{f(x)\neq \mathcal{C}_1} p(\boldsymbol{x}, \mathcal{C}_1)\mathrm{d}x + \int 1_{f(\boldsymbol{x})\neq \mathcal{C}_2} p(\boldsymbol{x}, \mathcal{C}_2)\mathrm{d}\boldsymbol{x} \\ &= \int_{\mathcal{R}_2} p\left(\boldsymbol{x}, \mathcal{C}_1\right)\mathrm{d}\boldsymbol{x} + \int_{\mathcal{R}_1} p\left(\boldsymbol{x}, \mathcal{C}_2\right)\mathrm{d}\boldsymbol{x} \end{aligned}$$

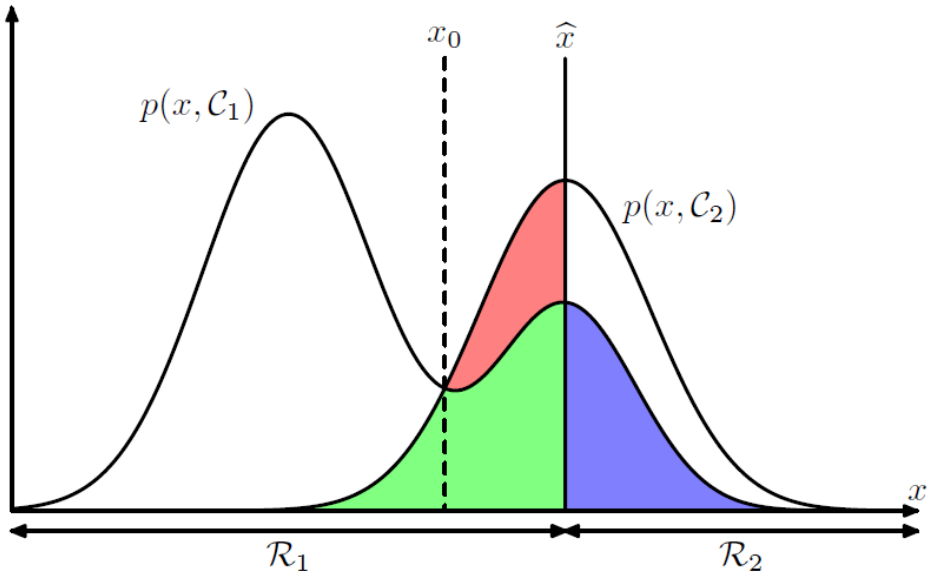
Question : existe-t-il un classifieur f minimisant E_{err} ?

Minimisation de l'erreur moyenne

$$E_{\text{err}} = \int_{\mathcal{R}_2} p(\boldsymbol{x}, \mathcal{C}_1)\, \mathrm{d}\boldsymbol{x} + \int_{\mathcal{R}_1} p(\boldsymbol{x}, \mathcal{C}_2)\, \mathrm{d}\boldsymbol{x}$$

Illustration : $x \in \mathbb{R}$, $\mathcal{R}_1 = \{x \in \mathbb{R}, x < \hat{x}\}$, $\mathcal{R}_2 = \{x \in \mathbb{R}, x > \hat{x}\}$

Question : comment fixer $\hat{x}$ de manière à minimiser E_{err} ?



$E_{\text{err}} =$ rouge + vert + bleu

Or vert + bleu = Cste
lorsque $\hat{x}$ varie

$\rightarrow$ minimum atteint pour $\hat{x} = x_0$ et alors :
$\mathcal{R}_1 = \{x,\ p(x, \mathcal{C}_1) > p(x, \mathcal{C}_2)\}$ et $\mathcal{R}_2 = \{x,\ p(x, \mathcal{C}_2) > p(x, \mathcal{C}_1)\}$

Illustration : C. Bishop, *Pattern Recognition and Machine Learning*, Springer 2006

Classifieur de Bayes

$\rightarrow$ ce raisonnement se généralise à $\boldsymbol{x} \in \mathbb{R}^d$, et $\mathcal{R}_i \neq$ intervalles

Conséquence : la règle de classification minimisant la **proportion moyenne d'erreurs** est

$$\begin{aligned} f(\boldsymbol{x}) = \text{argmax}_{\mathcal{C}_k}\, p(\boldsymbol{x}, \mathcal{C}_k) &= \text{argmax}_{\mathcal{C}_k}\, p(\boldsymbol{x})\, p(\mathcal{C}_k|\boldsymbol{x}) \\ &= \text{argmax}_{\mathcal{C}_k}\, p(\mathcal{C}_k)\, p(\boldsymbol{x}|\mathcal{C}_k) \end{aligned}$$

Classifieur de Bayes – maximum a posteriori (MAP)

$$f(\boldsymbol{x}) = \text{argmax}_{\mathcal{C}_k}\, p(\mathcal{C}_k|\boldsymbol{x}) = \text{argmax}_{\mathcal{C}_k}\, p(\mathcal{C}_k)\, p(\boldsymbol{x}|\mathcal{C}_k)$$

Problème : on ne connaît généralement pas les $p(\mathcal{C}_k)$ et $p(\boldsymbol{x}|\mathcal{C}_k)$...

En pratique ? (1)

Question : comment estimer $p(\mathcal{C}_k)$ et $p(\boldsymbol{x}|\mathcal{C}_k)$ à partir du jeu de données disponible (ensemble d'observations classifiées) ?

$p(\mathcal{C}_k)$:

- information connue a priori
  exemple : OCR
- ou fréquence estimée à partir de la base d'observations
  exemple : $p(\mathcal{C}_k) = \dfrac{\#\{x_i \in \mathcal{C}_k,\ 1 \leq i \leq N\}}{N}$
  où $\#$ *désigne le cardinal d'un ensemble*
- ou, dans le cas où les $p(\mathcal{C}_k)$ sont égaux :
  le classifieur de Bayes se simplifie en $f(\boldsymbol{x}) = \text{argmax}_k\, p(\boldsymbol{x}|\mathcal{C}_k)$
  $\rightarrow$ règle du maximum de vraisemblance (ML)

En pratique ? (2)

$p(\boldsymbol{x}|\mathcal{C}_k)$: toute une partie de l'apprentissage non-supervisé concerne l'*estimation de densités de probabilité*

si $\boldsymbol{x} \in \mathbb{R}^d$ avec $d \ll$ grand $\gg$: attention, *curse of dimensionality !*
(voir discussion poly : estimation d'une matrice de covariance en grande dimension)

$\rightarrow$ en pratique, on a intérêt à réduire la dimension d / le nombre de paramètres du modèles...
(ex. : matrice de covariance en dimension d : $d(d+1)/2$ paramètres)

$\rightarrow$ des hypothèses simplificatrices sont nécessaires pour mettre en œuvre le classifieur de Bayes

Plan

1. Classification et décision statistique
   - Éléments de théorie statistique de la décision
   - Le ≪ meilleur ≫ classifieur : classifieur de Bayes
2. Mise en œuvre du classifieur de Bayes
   - Classifieur naïf de Bayes
   - Classification aux plus proches voisins
   - Régression logistique
   - GMM et partitionnement
3. Point méthodologique : apprentissage, validation, test
4. Conclusion

Classifieur naïf de Bayes

Une manière de battre la malédiction de la dimensionnalité. . .

Si $\boldsymbol{x} = (x^1, x^2, \ldots, x^d) \in \mathbb{R}^d$, on suppose les composantes *conditionnellement statistiquement indépendantes*

Donc : $p(\boldsymbol{x}|\mathcal{C}_k) = \prod_{i=1}^{d} p(x^i|\mathcal{C}_k)$

Gros avantage : plutôt qu'estimer la distribution $p(\boldsymbol{x}|\mathcal{C}_k)$ sur $\mathbb{R}^d$, on estime les d distributions $p(x^i|\mathcal{C}_k)$ sur $\mathbb{R}$.

Classifieur naïf de Bayes

$$f(\boldsymbol{x}) \;=\; \text{argmax}_{\mathcal{C}_k}\, p(\mathcal{C}_k) \prod_{i=1}^{d} p(x^i|\mathcal{C}_k)$$

Exemple : classifieur naïf gaussien

$\rightarrow$ on suppose les distributions $p(x^i|\mathcal{C}_k)$ gaussiennes

$\rightarrow$ deux paramètres (μ_k, σ_k) pour chaque gaussienne

Estimateur de distribution aux plus proches voisins

On cherche à estimer une distribution de probabilité ϕ à partir de M observations $\boldsymbol{x}_i$

Estimateur des P plus proches voisins (K-NN) :

$$\phi(\boldsymbol{x}) = \frac{P}{M\, V_P(\boldsymbol{x})}$$

où $V_P(\boldsymbol{x})$: volume d'une boule B contenant les P p.p.v. de $\boldsymbol{x}$

$\rightarrow$ hypothèse : ϕ constant sur B
Donc il faudrait une boule « pas trop grosse »
(malédiction dimensionnalité ?)

$\rightarrow$ en fait, utilisé pour la classification supervisée...

Classification aux P plus proches voisins

Base de données : N observations $x_1, \ldots, x_N$ et classes associées parmi $\mathcal{C}_1, \ldots, \mathcal{C}_K$.
$\rightarrow$ N_1 observations dans $\mathcal{C}_1, \ldots,$ N_K dans $\mathcal{C}_K$, t.q. $\sum_k N_k = N$

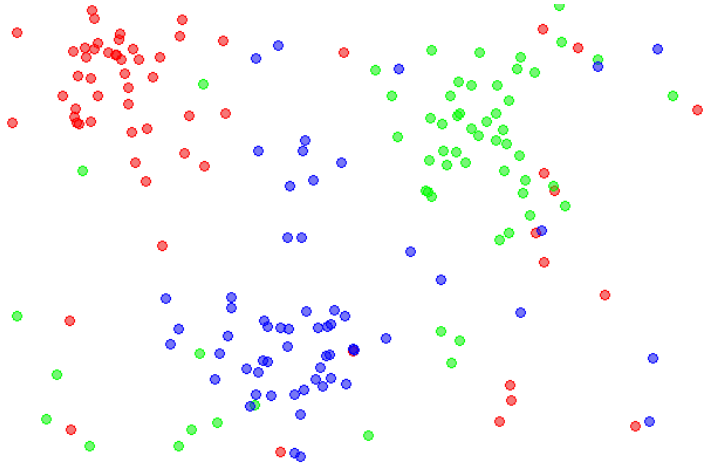
Problème : étant donnée une nouvelle observation $\boldsymbol{x}$, comment prédire sa classe ?

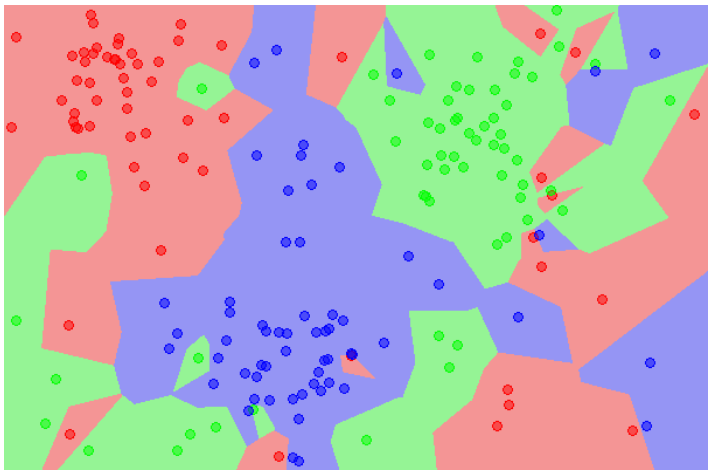
Parmi les P p.p.v. de $\boldsymbol{x}$: P_1 dans $\mathcal{C}_1, \ldots,$ P_K dans C_K $(\sum_k P_k = P)$
Par estimation aux P-p.p.v. : $p(\boldsymbol{x}|\mathcal{C}_k) = \frac{P_k}{N_k V_P(\boldsymbol{x})}$
De plus : $p(\mathcal{C}_k) = \frac{N_k}{N}$

Classifieur MAP :
$\text{argmax}_k p(\mathcal{C}_k)p(\boldsymbol{x}|\mathcal{C}_k) = \text{argmax}_k P_k/(NV_P(\boldsymbol{x})) = \text{argmax}_k P_k$

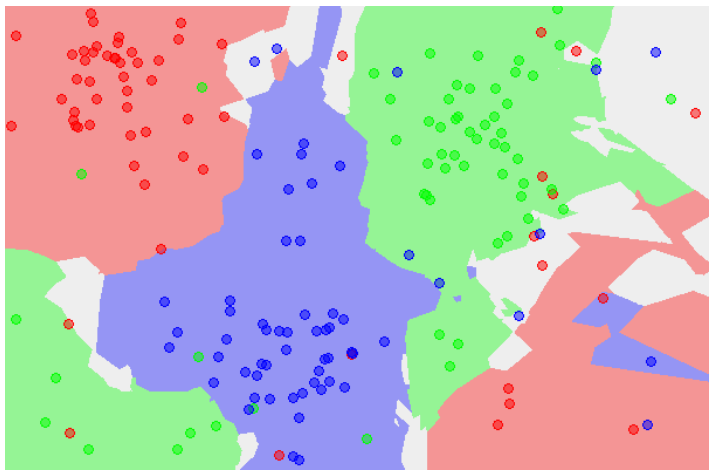
Définition : le *classifieur aux P-p.p.v.* affecte une nouvelle observation à la classe majoritaire parmi les P observations les plus proches
$\rightarrow$ implémente le classifieur de Bayes sous hypothèses (très) simplificatrices

Exemple





1-p.p.v.



5-p.p.v.

K augmente $\rightarrow$ effet de régularisation

$+$: dépendance moins grande à la base de données

$-$: on s'écarte des hypothèses permettant d'approcher le classif. de Bayes

La régression logistique

Dans le cas bi-classe :

$$p(\mathcal{C}_1|\boldsymbol{x}) = \frac{p(\mathcal{C}_1)p(\boldsymbol{x}|\mathcal{C}_1)}{p(\mathcal{C}_1)p(\boldsymbol{x}|\mathcal{C}_1) + p(\mathcal{C}_2)p(\boldsymbol{x}|\mathcal{C}_2)} = \frac{1}{1 + \frac{p(\mathcal{C}_2)p(\boldsymbol{x}|\mathcal{C}_2)}{p(\mathcal{C}_1)p(\boldsymbol{x}|\mathcal{C}_1)}}$$

Avec $f(\boldsymbol{x}) = \log\left(\frac{p(\boldsymbol{x}|\mathcal{C}_1)}{p(\boldsymbol{x}|\mathcal{C}_2)}\right) + \log\left(\frac{p(\mathcal{C}_1)}{p(\mathcal{C}_2)}\right)$: $p(\mathcal{C}_1|\boldsymbol{x}) = \frac{1}{1 + e^{-f(\boldsymbol{x})}}$

Définition : fonction logistique (ou sigmoïde) : $\sigma(t) = \frac{1}{1+e^{-t}}$

Hypothèse simplificatrice : $f(\boldsymbol{x}) = \beta_0 + \boldsymbol{\beta}_1 \cdot \boldsymbol{x}$

$\rightarrow$ c'est aussi une manière de contrer la malédiction de la dimensionnalité, en réduisant le nombre de paramètres à estimer

$\rightarrow$ régression logistique : estimation de β_0 et $\boldsymbol{\beta}_1$

Classifieur MAP : classifieur de la régression logistique

$\boldsymbol{x}$ dans $\mathcal{C}_1$ ssi $p(\mathcal{C}_1|\boldsymbol{x}) > 1/2 \iff f(\boldsymbol{x}) > 0$

(séparation des deux classes par un hyperplan)

Illustration

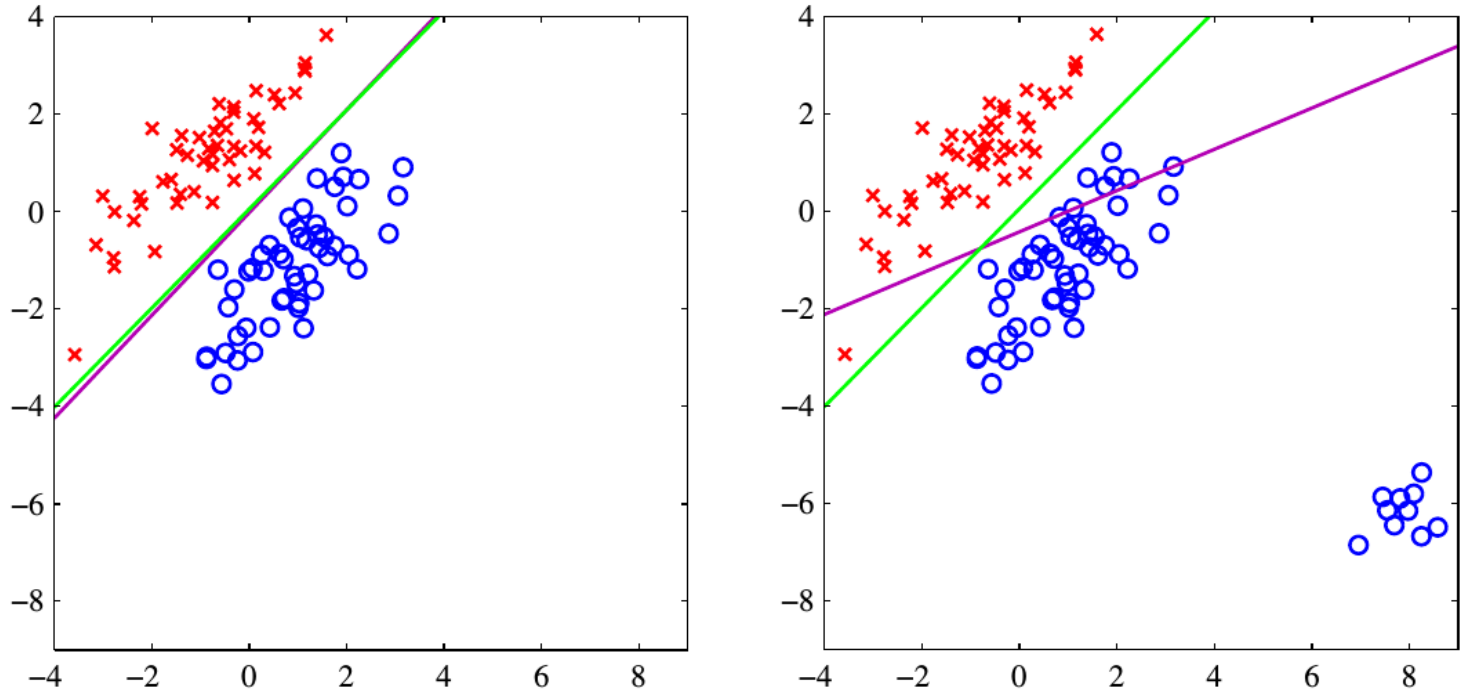


Figure 4.4 The left plot shows data from two classes, denoted by red crosses and blue circles, together with the decision boundary found by least squares (magenta curve) and also by the logistic regression model (green curve), which is discussed later in Section 4.3.2. The right-hand plot shows the corresponding results obtained when extra data points are added at the bottom left of the diagram, showing that least squares is highly sensitive to outliers, unlike logistic regression.

Illustration : C. Bishop, *Pattern Recognition and Machine Learning*, Springer 2006

→ l'estimation du classifieur de la régression logistique n'est pas sensible aux points "trop faciles à classer"

→ on reviendra sur cette propriété en séance 4 (SVM)

Partitionnement par mélange de gaussiennes

Estimation par EM des paramètres d'un mélange de M gaussiennes (cf séance 2) :

$$p(\boldsymbol{x}) = \sum_{m=1}^{M} \pi_m \mathcal{N}_{\mu_m, \Sigma_m}(\boldsymbol{x})$$

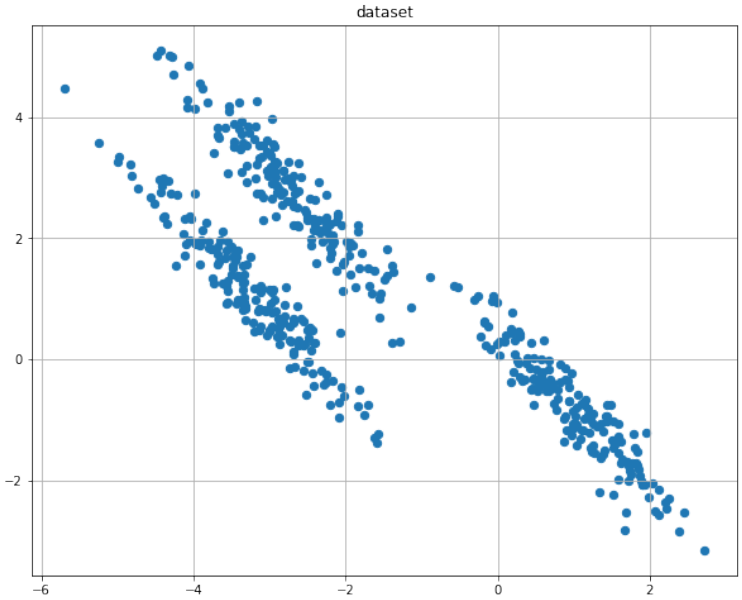
$\pi_m \leftrightarrow p(\mathcal{C}_m)$
$\mathcal{N}_{\mu_m, \Sigma_m}(\boldsymbol{x}) \leftrightarrow p(\boldsymbol{x}|\mathcal{C}_m)$
On dispose de

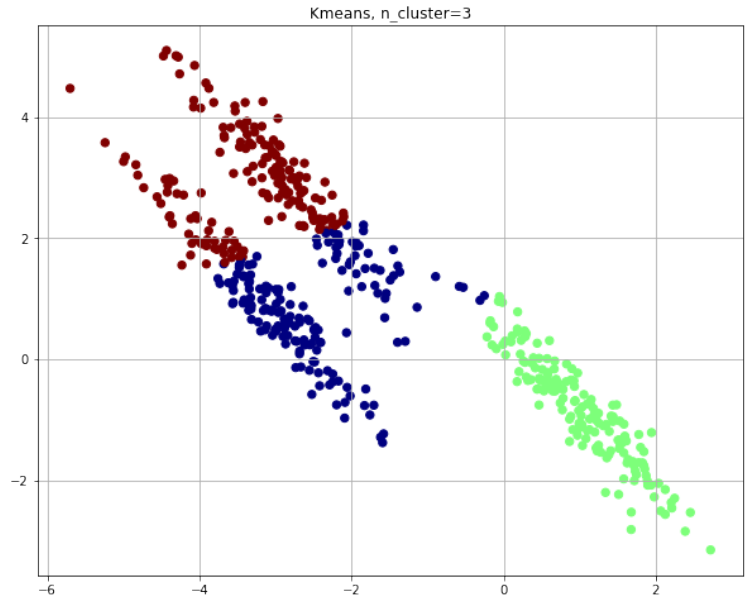
$$\gamma_{nm} = \frac{\pi_m \mathcal{N}_{\mu_m, \Sigma_m}(\boldsymbol{x}_n)}{\sum_{m=1}^{M} \pi_m \mathcal{N}_{\mu_m, \Sigma_m}(\boldsymbol{x}_n)} = p(\mathcal{C}_m|\boldsymbol{x}_n)$$

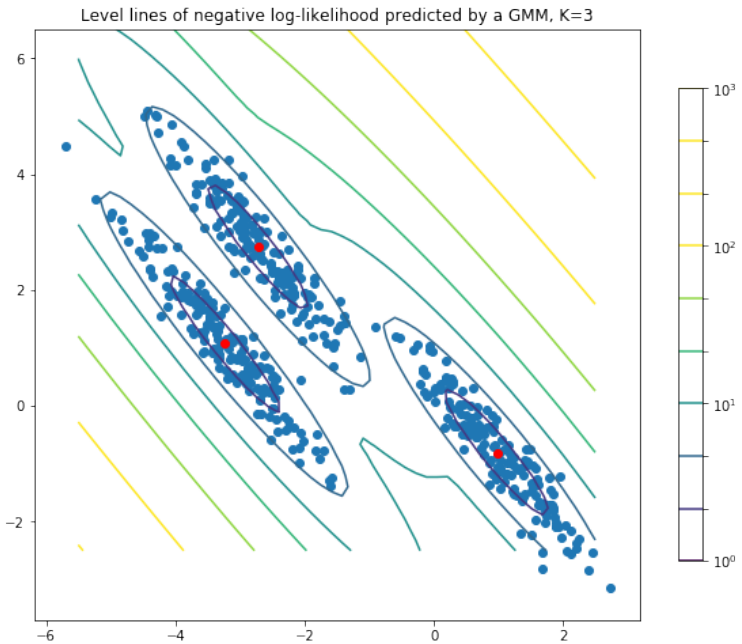
Partitionnement d'après GMM :
on affecte $\boldsymbol{x}_n$ à $\mathcal{C}_k$ tel que
$k = \text{argmax}_m \gamma_{nm} = \text{argmax}_m \pi_m \mathcal{N}_{\mu_m, \Sigma_m}(\boldsymbol{x}_n)$.

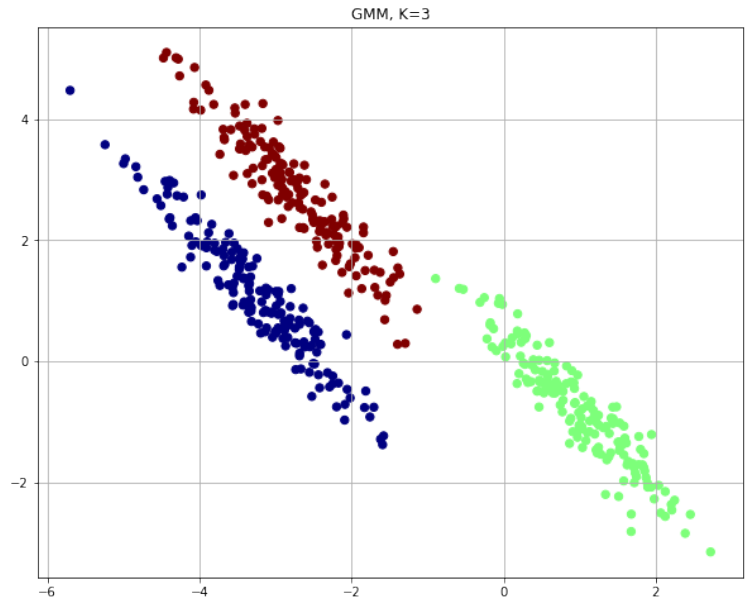
Remarque : lien avec les K-moyennes. . .cf polycopié

Exemple (cf TP 2 exercice 1)









Plan

1. Classification et décision statistique
   - Éléments de théorie statistique de la décision
   - Le ≪ meilleur ≫ classifieur : classifieur de Bayes
2. Mise en œuvre du classifieur de Bayes
   - Classifieur naïf de Bayes
   - Classification aux plus proches voisins
   - Régression logistique
   - GMM et partitionnement
3. Point méthodologique : apprentissage, validation, test
4. Conclusion

Validation des modèles d'apprentissage supervisé

Objectif : estimer les performances d'un modèle.

Exemple (cf `TP1_Ex1`) : régression linéaire.

Méthode :

1. estimation des paramètres sur la base d'apprentissage (x_i, y_i)
2. calcul d'un score moyen (par ex. RMSE) sur la base de test

Idée sous-jacente : le score moyen de test est une estimation de l'espérance du score (score moyen face à de nouvelles observations)

$\rightarrow$ et si on veut sélectionner **un** modèle parmi les régressions polynomiales de degré $d = 1, 2, 5, 10$?
Rappel : d est un hyperparamètre

On peut envisager de sélectionner d minimisant le score moyen de test.

Sélection de modèle

$\rightarrow$ et si on veut comparer à un autre modèle (avec ses propres hyperparamètres) ?
(exemple : Lasso, hyperparamètre additionnel = coef. régularisation)

Problème : comparer les scores de test des modèles d'hyperparamètres optimaux peut fournir une estimation biaisée optimiste de l'erreur de prédiction car on ne considère que des scores minimum.
(attention aux fluctuations d'échantillonnage, cf exemple séance 1)

Solution :

1. estimation des paramètres sur la base d'apprentissage
2. sélection des hyperparamètres par le score estimé sur la base de validation
3. on compare les modèles sur une base de test indépendante

Remarques

1. les trois bases de données doivent être indépendantes, et doivent partager la même distribution de probabilité des données
2. répartition possible :
   apprentissage 60%, validation 20 %, test 20%
3. on peut joindre base d'apprentissage et base de validation pour faire de la validation croisée (cf séance 1) pour réduire les fluctuations d'échantillonnage
4. si la base de validation est suffisamment grande, on peut comparer directement les scores de validation croisée des différents modèles
   (le score moyen de validation croisée est alors proche du score sur la base de test)
5. souvent, confusions entre test et validation dans la littérature...

Plan

1. Classification et décision statistique
   - Éléments de théorie statistique de la décision
   - Le ≪ meilleur ≫ classifieur : classifieur de Bayes
2. Mise en œuvre du classifieur de Bayes
   - Classifieur naïf de Bayes
   - Classification aux plus proches voisins
   - Régression logistique
   - GMM et partitionnement
3. Point méthodologique : apprentissage, validation, test
4. Conclusion

Fonction discriminante pour la classification supervisée

Notion de **fonction discriminante** :
f_k tel que $f(\boldsymbol{x}) = \text{argmax}_k f_k(\boldsymbol{x})$

- MAP (classifieur de Bayes, **théoriquement** optimal) :

$$f(\boldsymbol{x}) = \text{argmax}_k p(\mathcal{C}_k|\boldsymbol{x}) = \text{argmax}_k p(\mathcal{C}_k)p(\boldsymbol{x}|\mathcal{C}_k)$$

  Problème : on ne connaît pas les distributions de probabilité
  $\rightarrow$ on ajoute des hypothèses : classifieur naïf de Bayes, régression logistique, P-p.p.v....

- Cours suivants : autres hypothèses, autres fonctions discriminantes

Conclusion – Résumé

Théorie statistique de la décision, et mise en œuvre :

- le classifieur bayésien (MAP) minimise l'erreur moyenne de classification (classifieur **idéal théorique**)
- si *prior* uniformes : classification au maximum de vraisemblance (ML)
- simplification si composantes conditionnellement indépendantes : classifieur naïf de Bayes
- simplification si $p(\mathcal{C}_1|\boldsymbol{x}) = \sigma(\beta_0 + \boldsymbol{\beta}_1 \cdot \boldsymbol{x})$: régression logistique
- simplification si $p(\boldsymbol{x}|\mathcal{C}_1)$ ne varie pas trop localement et suffisamment d'observations : classification aux P plus proches voisins