Classification et régression en grande dimension

Stéphane Girard

Inria Grenoble Rhône-Alpes, projet Mistis
http://mistis.inrialpes.fr/~girard

Plan

1. Classification en grande dimension
2. Régression en grande dimension

Plan

1 Classification en grande dimension

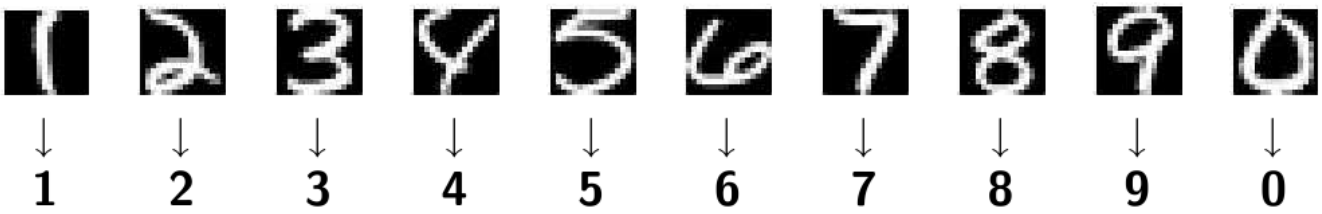
2 Régression en grande dimension

Introduction

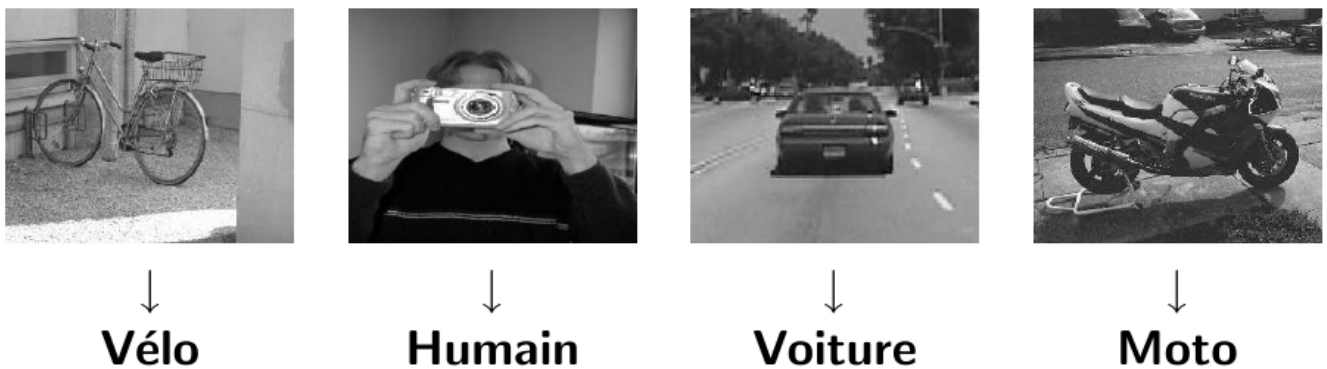
La classification est un problème récurrent :

- qui intervient généralement dans les applications nécessitant une prise de décision,
- les données modernes sont souvent de grande dimension.

Exemple 1 : reconnaissance optique de caractères



Exemple 2 : reconnaissance d'objets à partir d'images

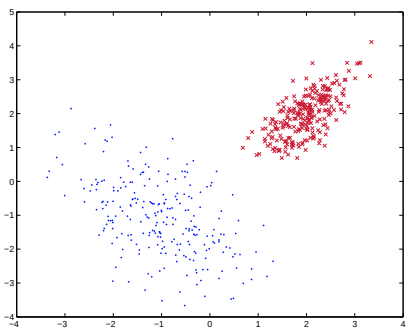
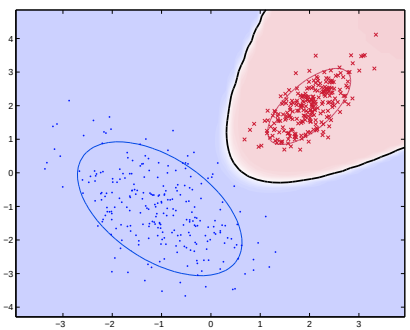
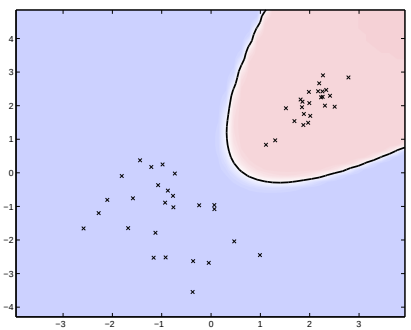


Le problème de la classification

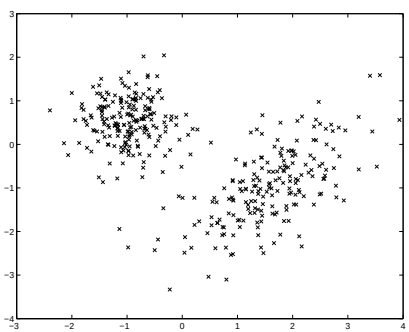
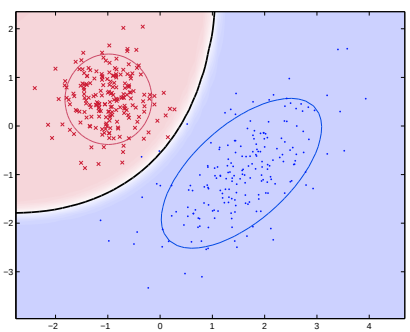
Le problème de la classification est :

- organiser des données $x_1, ..., x_n \in \mathbb{R}^p$ en k classes,
- les labels des données sont notés $z_1, ..., z_n \in \{1, ..., k\}$.

Approche supervisée : jeu de données complètes $(x_1, z_1), ..., (x_n, z_n)$ disponible pour l'apprentissage

Approche non-supervisée : uniquement les observations $x_1, ..., x_n$

Le modèle de mélange

On suppose classiquement que

- les observations $x_1, ..., x_n$ sont des réalisations indépendantes d'un vecteur aléatoire $X \in \mathbb{R}^p$,
- les labels $z_1, ..., z_n$ sont issus d'une variable aléatoire Z,

où :

- Z suit une loi multinomiale de paramètres $\pi_1, ..., \pi_k$ appelés proportions du mélange, *i.e.* $\mathbb{P}(Z = i) = \pi_i$, $i = 1, ..., k$.
- sachant $Z = i$, X suit une loi multidimensionnelle de densité $f_i(x)$.

En résumé, la densité de X s'écrit :

$$f(x) = \sum_{i=1}^{k} \pi_i f_i(x).$$

Règle de Bayes et modèle de mélange

La classification vise donc construire une règle de décision δ :

$$\begin{aligned} \delta : \mathbb{R}^p &\to \{1, ..., k\}, \\ x &\to z. \end{aligned}$$

La règle optimale δ^* (pour un coût 0-1), dite règle de Bayes ou du MAP (Maximum A Posteriori), est :

$$\begin{aligned} \delta^*(x) &= \underset{i=1,...,k}{\operatorname{argmax}} \mathbb{P}(Z = i | X = x) \\ &= \underset{i=1,...,k}{\operatorname{argmax}} \mathbb{P}(X = x | Z = i)\mathbb{P}(Z = i) \\ &= \underset{i=1,...,k}{\operatorname{argmin}} K_i(x), \end{aligned}$$

où la fonction de coût K_i est telle que $K_i(x) = -2\log(\pi_i f_i(x))$.
Remarque *: la construction de la règle de décision consiste à estimer f_i ou de façon équivalente K_i.*

Modèles gaussiens

Modèle gaussien Full-GMM (QDA en supervisé) :

$$K_i(x) = (x - \mu_i)^t \Sigma_i^{-1} (x - \mu_i) + \log(\det \Sigma_i) - 2\log(\pi_i) + C^{te}.$$

Modèle gaussien Com-GMM qui suppose que $\forall i,\ \Sigma_i = \Sigma$ (LDA en supervisé) :

$$K_i(x) = \mu_i^t \Sigma^{-1} \mu_i - 2\mu_i^t \Sigma^{-1} x - 2\log(\pi_i) + C^{te}.$$

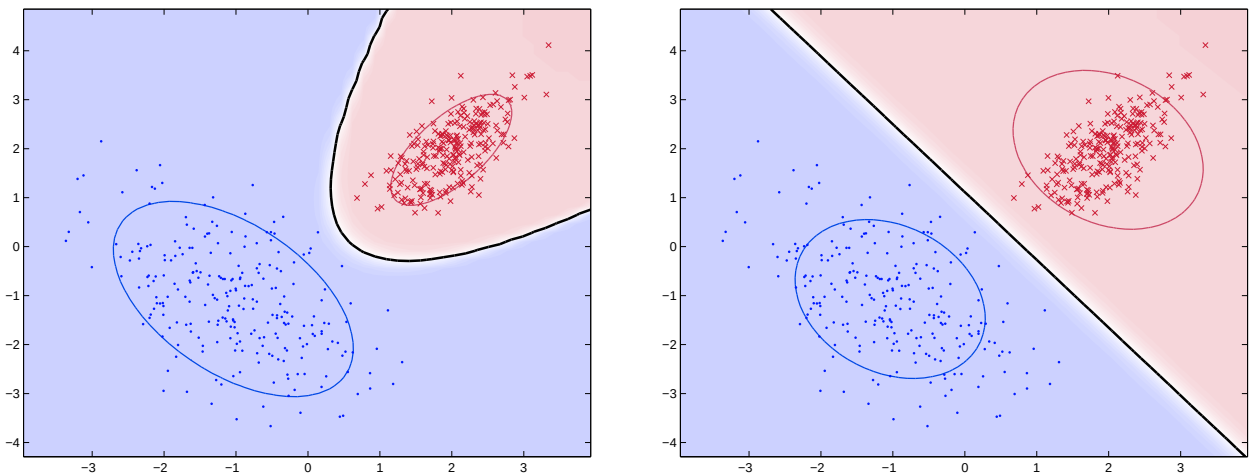


Fig. 1. Règles de décision de Full-GMM (gauche) et Com-GMM (droite).

Problème *: il est nécessaire d'inverser* Σ_i *ou* Σ*.*

Fléau de la dimension en classification

Fléau de la dimension dans le cas du mélange gaussien :

- le nombre de paramètres croît avec le carré de la dimension,

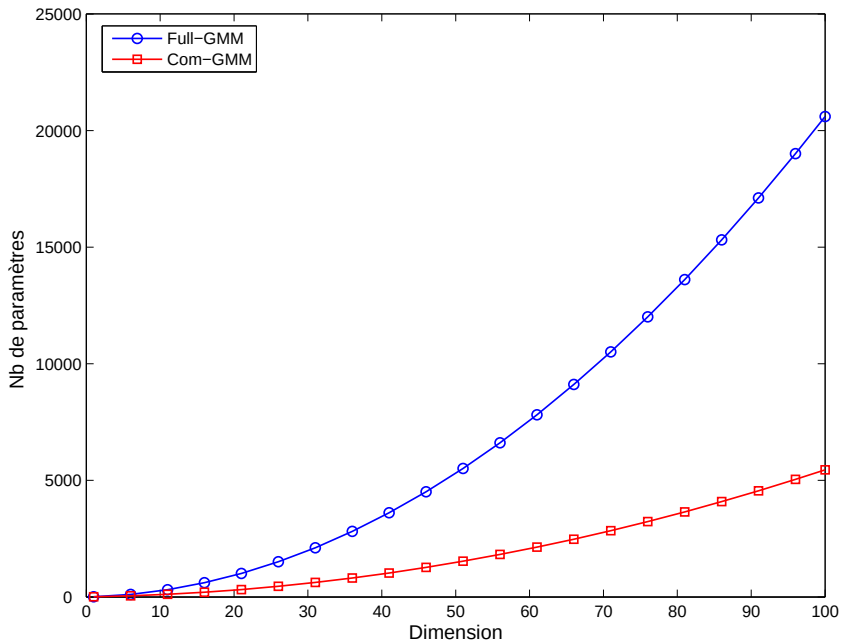


Fig. 2. Nombre de paramètres à estimer des modèles Full-GMM et Com-GMM en fonction de la dimension et ce pour 4 classes.

- si n est faible, les estimations des matrices de covariance sont mal conditionnées ou singulières,
- il est alors difficile ou impossible de les inverser et la règle de décision en est d'autant perturbée.

Les « bienfaits » de la dimension

Le phénomène de l'espace vide [Scot83] met en évidence que :

- les espaces de grande dimension sont quasiment vides,
- les données vivent dans des sous-espaces de dimensions intrinsèques inférieures à la dimension de l'espace p.

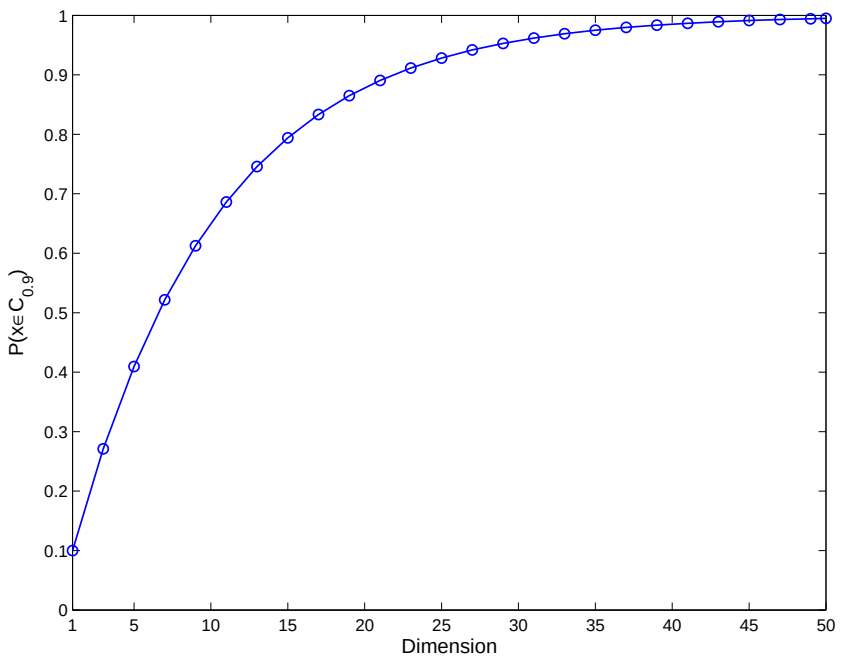


Fig. 5. Probabilité que $X \sim U_{B_p(0,1)}$ soit dans la coquille comprise entre les boules de rayon 0.9 et 1, en fonction de la dimension : $\mathbb{P}(X \in C_{[0.9,1]}) = 1 - 0.9^p$.

Les « bienfaits » de la dimension

Un autre phénomène intervient en grande dimension :

- les espaces de grande dimension étant quasiment vides,
- il est plus facile de séparer les groupes en grande dimension avec un classifieur adapté.

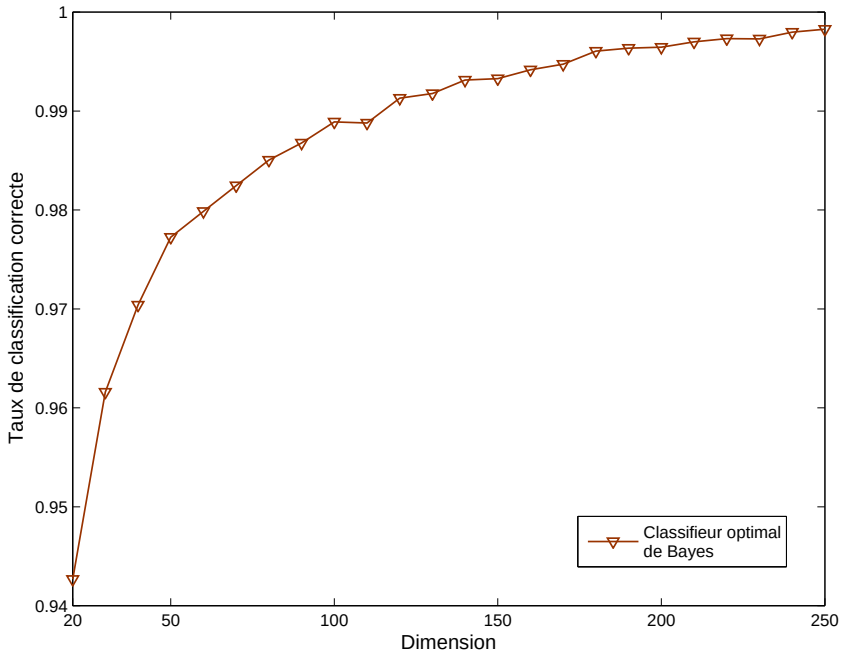


Fig. 6. Taux de classification correcte du classifieur optimal de Bayes en fonction de la dimension (données simulées).

L'idée de notre modélisation

Il est possible d'adapter ces postulats au cadre de la classification :

- les données de chaque classe vivent dans des sous-espaces différents de dimensions intrinsèques différentes,
- le fait de conserver toutes les dimensions permet de discriminer plus facilement les données.

Nous proposons donc une paramétrisation du modèle gaussien :

- qui exploite ces caractéristiques des données de grande dimension,
- au lieu de pallier les problèmes dus à la grande dimension des données.

Modélisation

Nous nous plaçons dans le cadre du modèle de mélange gaussien :

$$f(x) = \sum_{i=1}^{k} \pi_i f(x, \theta_i), \text{ avec } f(x, \theta_i) \sim \mathcal{N}(\mu_i, \Sigma_i).$$

En se basant sur la décomposition spectrale de Σ_i, on peut écrire :

$$\Sigma_i = Q_i \Delta_i Q_i^t,$$

où :

- Q_i est la matrice orthogonale des vecteurs propres de Σ_i,
- Δ_i est la matrice diagonale des valeurs propres de Σ_i.

Modélisation

Nous proposons de paramétrer la matrice Δ_i de la façon suivante :

$$\Delta_i = \begin{pmatrix} \begin{bmatrix} a_{i1} & & 0 \\ & \ddots & \\ 0 & & a_{id_i} \end{bmatrix} & \mathbf{0} \\ \mathbf{0} & \begin{bmatrix} b_i & & 0 \\ & \ddots & \\ 0 & & b_i \end{bmatrix} \end{pmatrix} \begin{matrix} \left.\vphantom{\begin{bmatrix} a \\ \\ a \end{bmatrix}}\right\} d_i \\ \left.\vphantom{\begin{bmatrix} a \\ \\ a \end{bmatrix}}\right\} (p - d_i) \end{matrix}$$

où $a_{ij} \geq b_i$, pour $j = 1, ..., d_i$.

Remarque *: cette paramétrisation est toujours possible car si on prend $d_i = p - 1$, pour $i = 1, ..., k$, alors on a le modèle Full-GMM.*

Modélisation

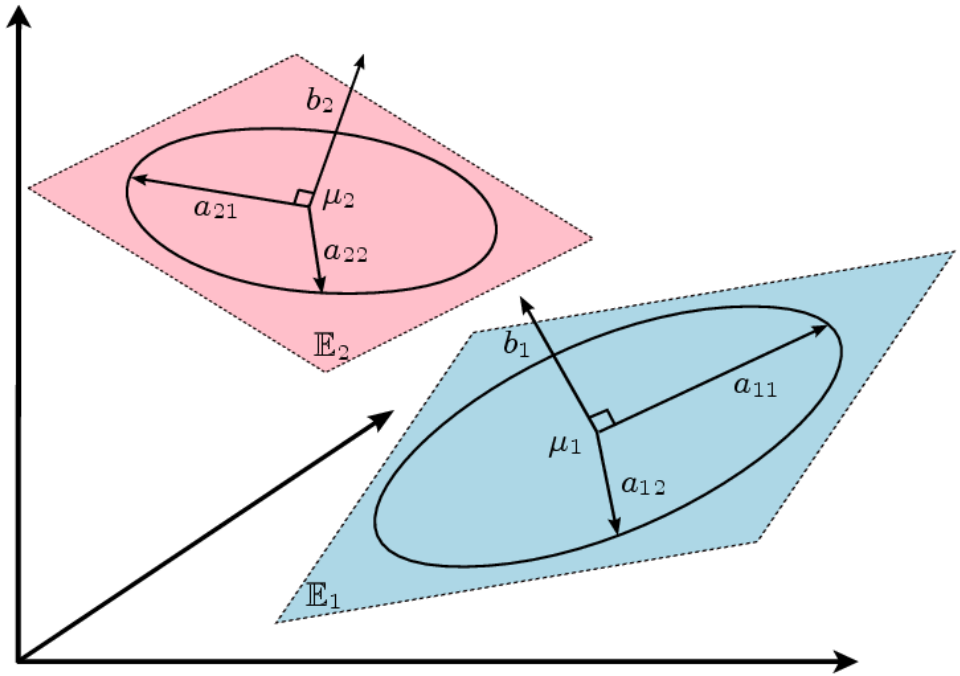


Fig. 7. Notre paramétrisation du modèle de mélange gaussien.

Nous définissons en outre :

- $\mathbb{E}_i$ l'espace engendré par les vect. prop. associés aux a_{ij},
- $\mathbb{E}_i^{\perp}$ son supplémentaire dans $\mathbb{R}^p$,
- P_i et $P_i^{\perp}$ les opérateurs de projection sur $\mathbb{E}_i$ et $\mathbb{E}_i^{\perp}$.

Le modèle $[a_{ij}b_iQ_id_i]$ et ses sous-modèles

Ainsi, nous obtenons une paramétrisation du modèle gaussien :

- qui est fonction de a_{ij}, b_i, Q_i et d_i,
- dont la complexité est contrôlée par les dimensions d_i des sous-espaces,
- que nous noterons $[a_{ij}b_iQ_id_i]$ dans la suite.

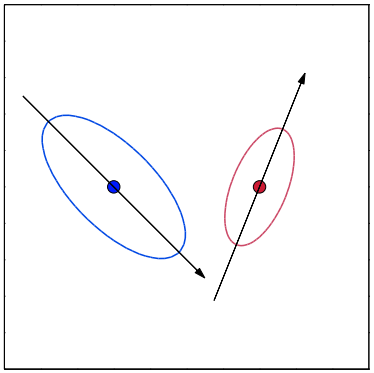
En forçant certains paramètres à être communs dans une même classe ou entre les classes :

- nous obtenons des modèles de plus en plus régularisés,
- qui vont du modèle général au modèle le plus parcimonieux.

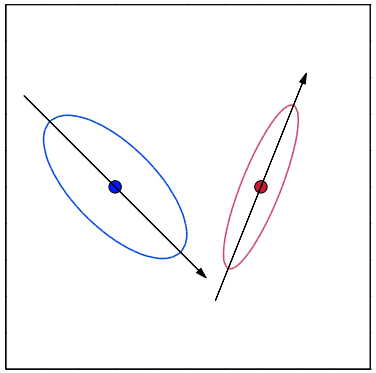
Notre famille contient 28 modèles répartis de la façon suivante :

- 14 modèles à orientations libres,
- 12 modèles à orientation commune,
- 2 modèles à matrice de covariance commune.

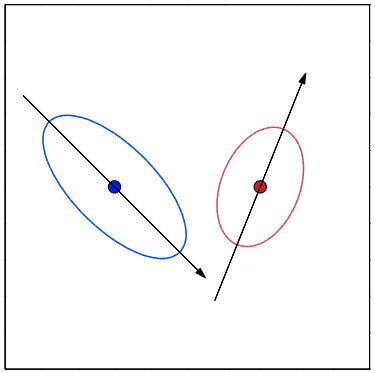
Le modèle $[a_{ij}b_iQ_id_i]$ et ses sous-modèles



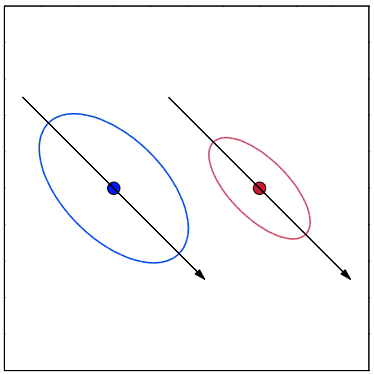
modèle $[a_ib_iQ_id]$



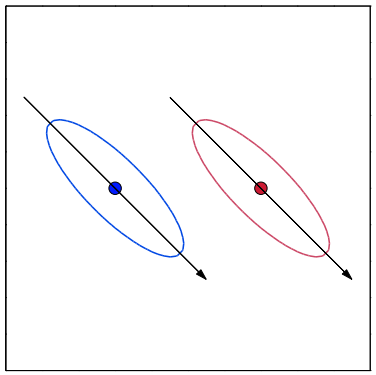
modèle $[ab_iQ_id]$



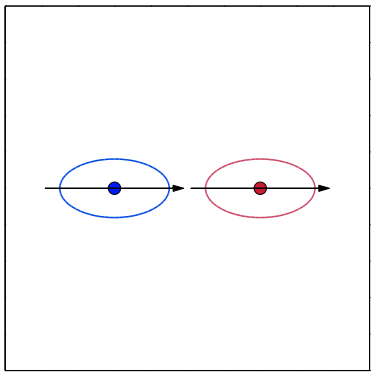
modèle $[a_ibQ_id]$



modèle $[a_ib_iQd]$



modèle $[abQd]$



modèle $[abI_2d]$

Fig. 8. Influence des paramètres a_i, b_i et Q_i sur les densités de 2 classes en dimension 2 et avec $d_1 = d_2 = 1$.

Le modèle $[a_{ij}b_iQ_id_i]$ et ses sous-modèles

Modèle	Nb de prms, $k = 4$ $d = 10$, $p = 100$	Type de classifieur
$[a_{ij}b_iQ_id_i]$	4231	Quadratique
$[a_{ij}b_iQd_i]$	1396	Quadratique
$[a_jbQd]$	1360	Linéaire
Full-GMM	20603	Quadratique
Com-GMM	5453	Linéaire

Table 1. Propriétés des modèles de la famille de $[a_{ij}b_iQ_id_i]$

Remarque *: le modèle* $[a_{ij}b_iQ_id_i]$ *qui engendre un classifieur quadratique requiert l'estimation de moins de paramètres que le modèle Com-GMM qui engendre un classifieur linéaire.*

Construction du classifieur HDDA

En supervisé, l'estimation des paramètres par MV est directe :

$$\hat{\pi}_i = \frac{n_i}{n},\ \hat{\mu}_i = \frac{1}{n_i}\sum_{j=1}^{n} s_{ij}x_j,$$

$$\hat{\Sigma}_i = W_i = \frac{1}{n_i}\sum_{j=1}^{n} s_{ij}(x_j - \hat{\mu}_i)(x_j - \hat{\mu}_i)^t,$$

où $n_i = \sum_{j=1}^{n} s_{ij}$ avec $s_{ij} = 1_{\{z_j=i\}}$.

Calcul des probabilités conditionnelles :

$$\mathbb{P}(Z = i|X = x_j, \theta) = 1/\sum_{\ell=1}^{k} \exp\left(\frac{1}{2}(K_i(x_j) - K_\ell(x_j))\right),$$

où la fonction de coût K_i est telle que $K_i(x) = -2\log(\pi_i f(x, \theta_i))$.

Expression de la fonction de coût K_i

Dans le cas du modèle $[a_i b_i Q_i d_i]$:

$$K_i(x) = \frac{1}{a_i}\|\mu_i - P_i(x)\|^2 + \frac{1}{b_i}\|x - P_i(x)\|^2 + d_i \log(a_i) + (p - d_i)\log(b_i) - 2\log(\pi_i).$$

Points forts :

- *pas besoin d'inverser la matrice de covariance,*
- *ni d'estimer les dernières colonnes de la matrice Q_i.*

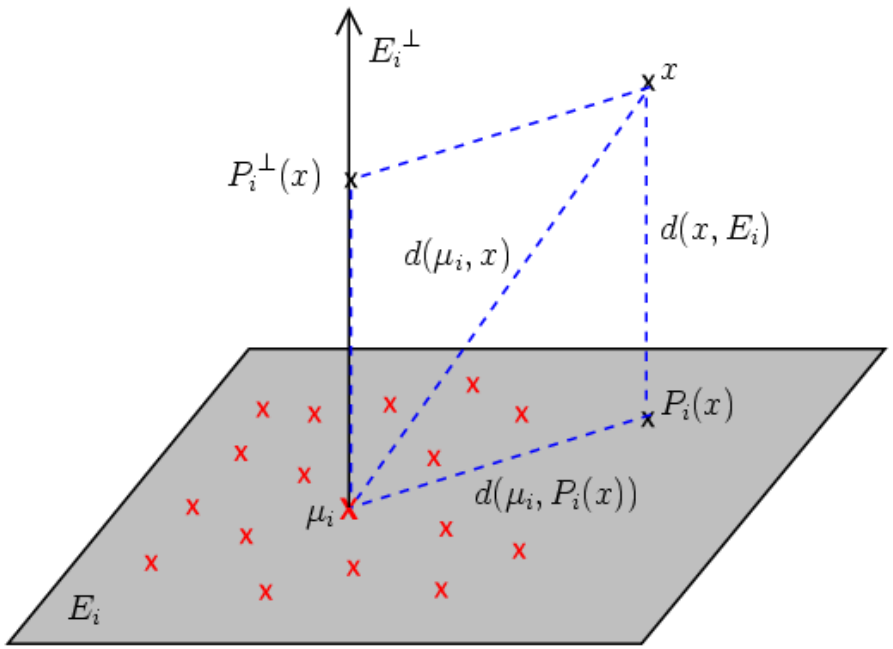


Fig. 9. Les sous-espaces $\mathbb{E}_i$ et $\mathbb{E}_i^\perp$ de la $i$ème composante.

Construction du classifieur HDDC

En non supervisé, les paramètres sont estimés par l'algorithme EM :

- Étape E : cette étape calcule à l'itération q les probabilités conditionnelles $t_{ij}^{(q)} = \mathbb{P}(Z = i|X = x_j, \theta^{(q)})$:

$$t_{ij}^{(q)} = 1/\sum_{\ell=1}^{k} \exp\left(\frac{1}{2}(K_i^{(q-1)}(x_j) - K_\ell^{(q-1)}(x_j))\right).$$

- Étape M : cette étape calcule les estimateurs des θ_i en maximisant la vraisemblance conditionnellement aux $t_{ij}^{(q)}$:

$$\hat{\pi}_i^{(q)} = \frac{n_i^{(q)}}{n},\ \hat{\mu}_i^{(q)} = \frac{1}{n_i^{(q)}}\sum_{j=1}^{n} t_{ij}^{(q)} x_j,$$

$$\hat{\Sigma}_i^{(q)} = W_i^{(q)} = \frac{1}{n_i^{(q)}}\sum_{j=1}^{n} t_{ij}^{(q)}(x_j - \hat{\mu}_i^{(q)})(x_j - \hat{\mu}_i^{(q)})^t,$$

où $n_i^{(q)} = \sum_{j=1}^{n} t_{ij}^{(q)}$.

Estimations des a_{ij}, b_i et Q_i

Les estimateurs du MV des paramètres du modèle $[a_{ij}b_iQ_id_i]$ sont explicites :

- Sous-espace $\mathbb{E}_i$: les d_i premières colonnes de Q_i sont estimées par les vecteurs propres associés aux d_i plus grandes valeurs propres λ_{ij} de W_i.
- Estimateur de a_{ij} : les paramètres a_{ij} sont estimés par les d_i plus grandes valeurs propres λ_{ij} de W_i.
- Estimateur de b_i : le paramètre b_i est estimé par :

$$\hat{b}_i = \frac{1}{(p-d_i)} \left(\text{trace}(W_i) - \sum_{j=1}^{d_i} \lambda_{ij} \right).$$

Remarque *: 16 des modèles de notre famille ont des estimateurs du MV explicites. Les autres requièrent une méthode itérative.*

Estimation des paramètres discrets

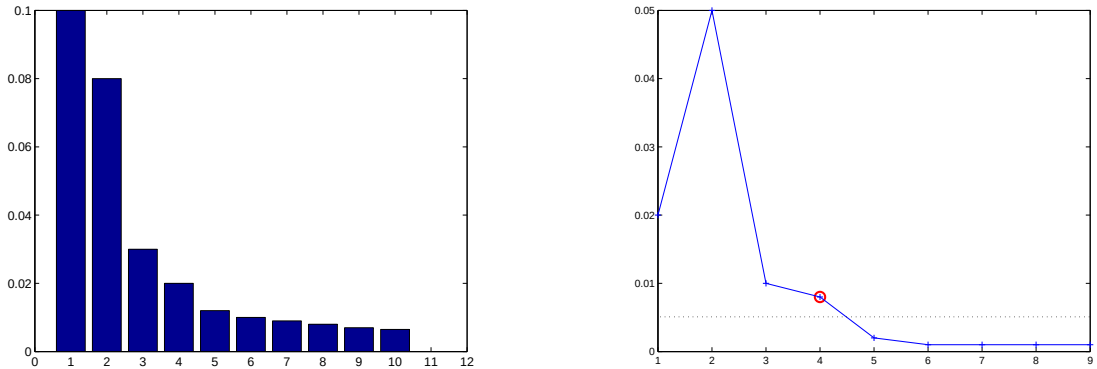


Fig. 10. Le scree-test de Cattell : éboulis des valeurs propres (gauche) et différences entre valeurs propres consécutives (droite).

Estimation des dimensions intrinsèques d_i :

- nous utilisons la méthode du *scree-test* de Cattell [Catt66],
- cela permet d'estimer de façon commune les k paramètres d_i,
- en supervisé, le seuil s est choisi par validation croisée,
- en non supervisé, s est choisi grâce au critère BIC [Schw78].

Estimation du nombre de groupes k :

- en supervisé, k est connu,
- en non supervisé, k est choisi grâce au critère BIC.

Considérations numériques

- Stabilité numérique : la règle de décision des classifieurs HDDA et HDDC ne dépend pas des vecteurs propres associés aux plus petites valeurs propres de W_i dont la détermination est instable.

- Réduction de la durée de calcul : pas besoin de déterminer les derniers vecteurs propres de $W_i \rightarrow$ réduction des temps de calcul avec une procédure adaptée ($\times 60$ pour $p = 1000$).

- Cas particulier où $n < p$: il est alors préférable, d'un point de vue numérique, de calculer les vecteurs propres de $U_i U_i^t$ au lieu de $W_i = U_i^t U_i$ où U_i contient les données centrées de C_i ($\times 500$ pour $n = 13$ et $p = 1000$).

HDDA : influence de la taille du jeu d'apprentissage

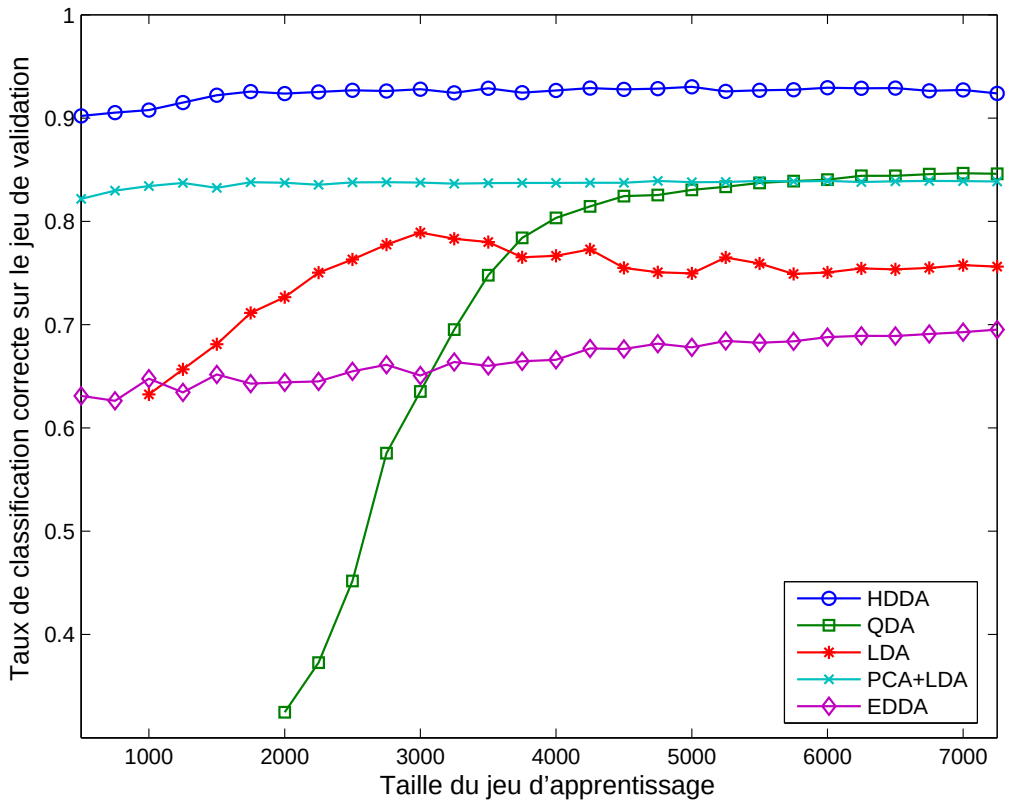


Fig. 12. Taux de classification correcte en fonction de la taille du jeu d'apprentissage (données réelles USPS $\in \mathbb{R}^{256}$***).***

Il apparaît que :

- l'HDDA est peu sensible à la taille du jeu d'apprentissage,
- l'HDDA est plus performante que les autres méthodes sur ce jeu de données réelles.

HDDA : comparaison avec les méthodes classiques

Méthode	Taux de classif. correcte	Temps d'app. (sec.)
HDDA $[a_{ij}bQ_id]$	**0.948**	~ 1
RDA ($\gamma = 0.3$, $\lambda = 0$)	0.935	~ 1
QDA (full-GMM)	0.846	~ 1
LDA (com-GMM)	0.757	~ 1
EDDA $[\lambda_k B_k]$	0.696	~ 1
SVM (linéaire)	0.926	~ 12

Table 2. Résultats de classification obtenus sur les données USPS ($p = 256$, $n_{app} = 7250$).

Il apparaît que :

- l'HDDA est plus performante que les autres méthodes sur ce jeu de données réelles,
- l'HDDA est aussi rapide que les autres méthodes basées sur le modèle de mélange (hors choix de modèles).

HDDC : sélection de modèles

Modèle de simulation	Modèle de classification					
	$[a_{ij}b_iQ_id_i]$	$[a_{ij}bQ_id_i]$	$[a_ib_iQ_id_i]$	$[a_ibQ_id_i]$	$[ab_iQ_id_i]$	$[abQ_id_i]$
$[a_{ij}b_iQ_id_i]$	96.7	82.8	97.3*	91.9	**97.5***	90.3
$[a_{ij}bQ_id_i]$	73.0	72.7	77.9	**78.2***	75.8	75.1
$[a_ib_iQ_id_i]$	97.9	87.1	98.3*	92.9	**98.6***	91.7
$[a_ibQ_id_i]$	82.6	80.0	**88.2***	86.3*	87.5	86.5
$[ab_iQ_id_i]$	96.5	82.5	**98.0***	84.4	95.2	82.2
$[abQ_id_i]$	71.2	75.2	**79.7**	79.3*	71.1	70.7

Table 3. Taux de classification correcte (en %) obtenus par les modèles de l'HDDC sur différents jeux de données simulées. Le modèle choisi par le critère BIC est noté d'une étoile.

Il apparaît que :

- le modèle $[a_ib_iQ_id_i]$ semble être particulièrement efficace,
- l'hypothèse que Δ_i n'a que deux valeurs propres différentes semble être un moyen efficace de régulariser son estimation.

HDDC : comparaison avec la sélection de variables

Modèle	Variables originales	Avec réduction de dimension (ACP)
Sphe-GMM	0.340	0.340
Diag-GMM	0.355	0.535
Com-GMM	0.625	0.635
Full-GMM	0.640	0.845
VS-GMM [Raft05]	0.925	/
HDDC $[a_i b_i Q_i d_i]$	**0.950**	/

Table 5. Taux de classification correcte sur les données « Crabes ».

Il apparaît que :

- notre approche est plus efficace que la réduction de dimension et la sélection de variables sur ce jeu de données réelles,
- l'HDDC est efficace même en dimension faible sur des données complexes.

HDDC : les étapes de l'algorithme

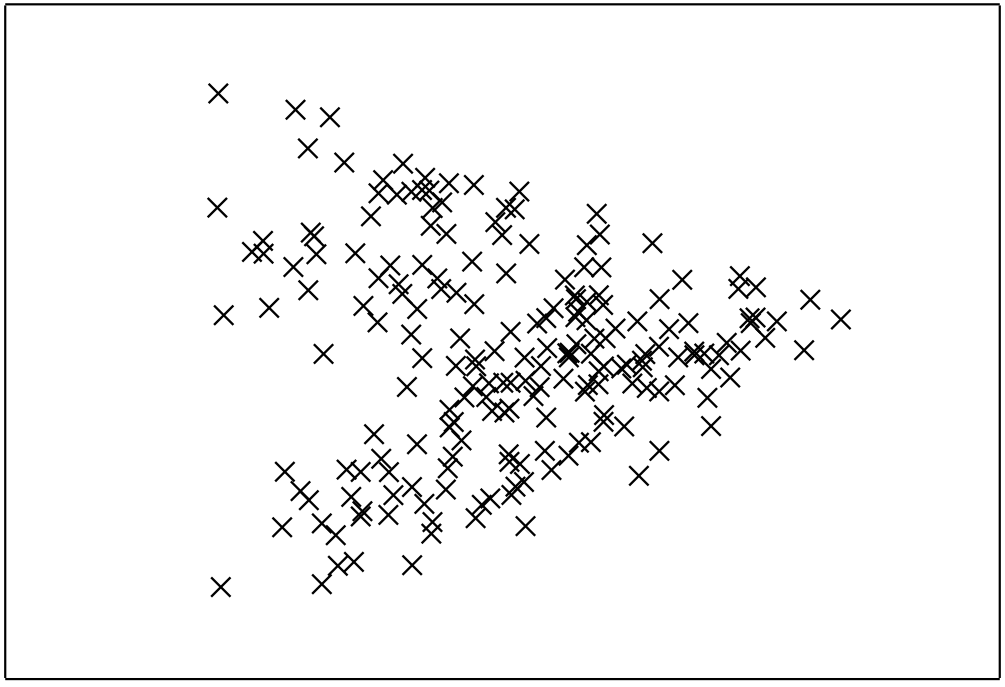


Fig. 13. Projection des données « Crabes » sur les axes principaux.

Données « Crabes » :

- 200 individus en dimension $p = 5$ (5 caractéristiques morphologiques des crabes),
- répartis en 4 classes (MB, FB, MO et FO).

HDDC : les étapes de l'algorithme

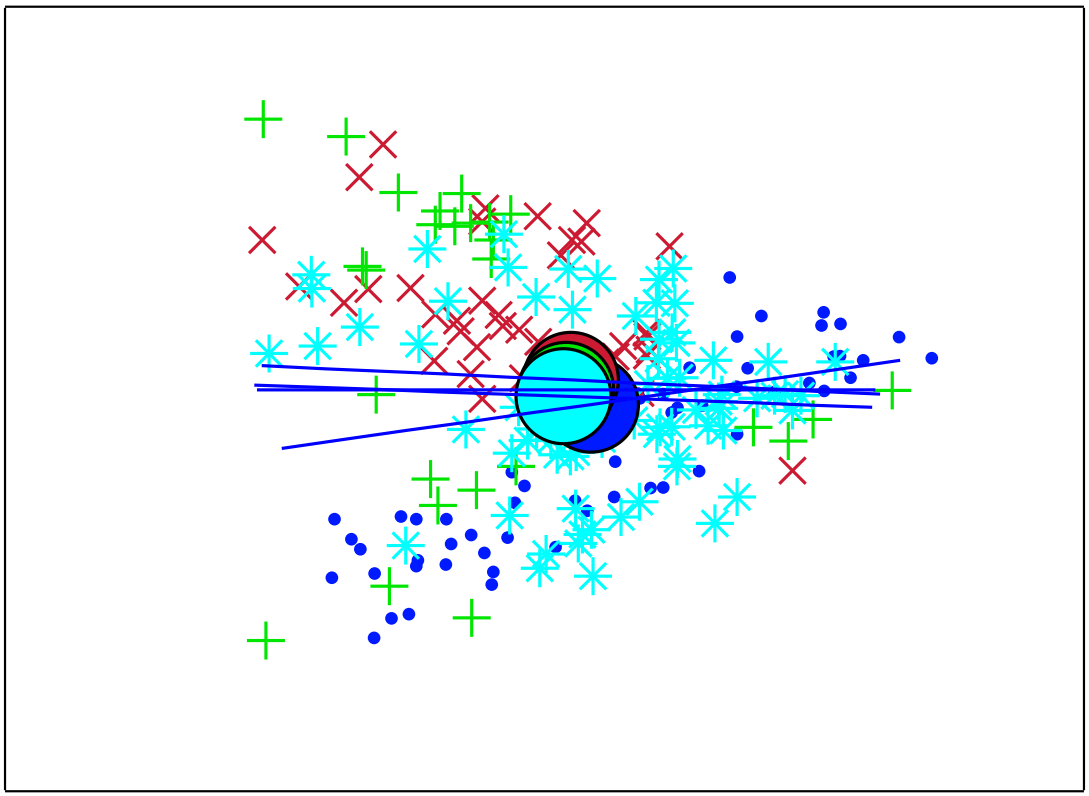


Fig. 14. Etape n° 1 de l'HDDC sur les données « Crabes ».

HDDC : les étapes de l'algorithme

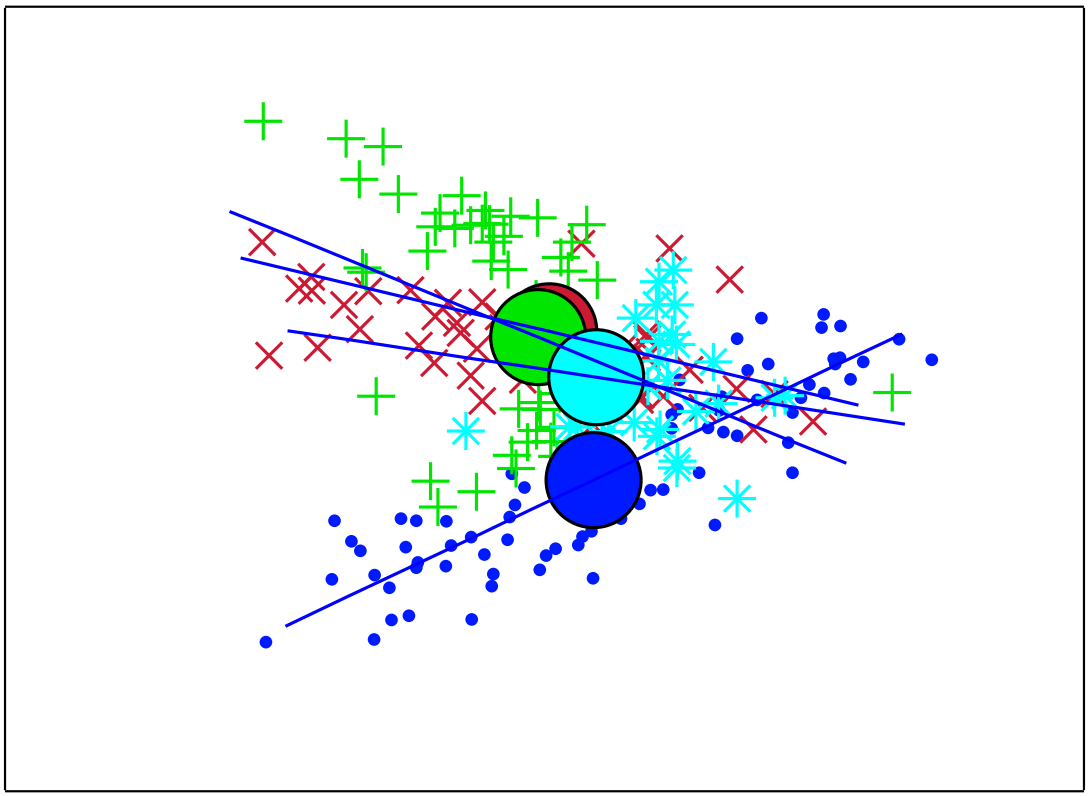


Fig. 14. Etape n° 4 de l'HDDC sur les données « Crabes ».

HDDC : les étapes de l'algorithme

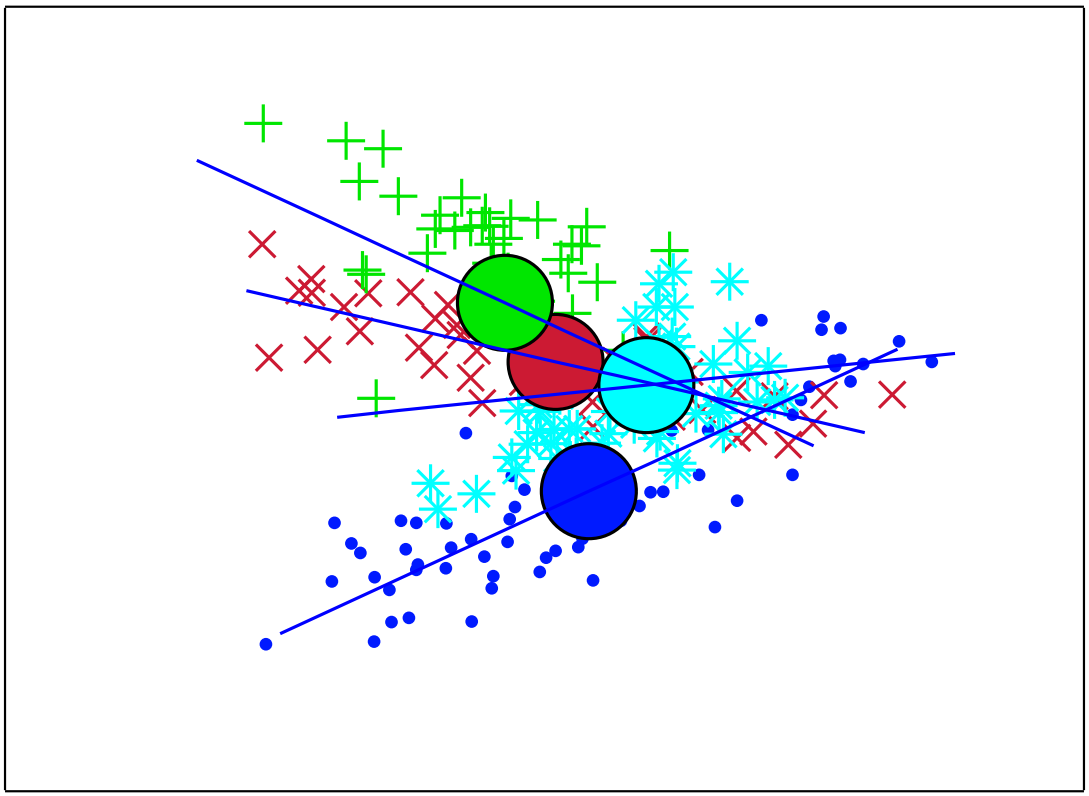


Fig. 14. Etape n° 7 de l'HDDC sur les données « Crabes ».

HDDC : les étapes de l'algorithme

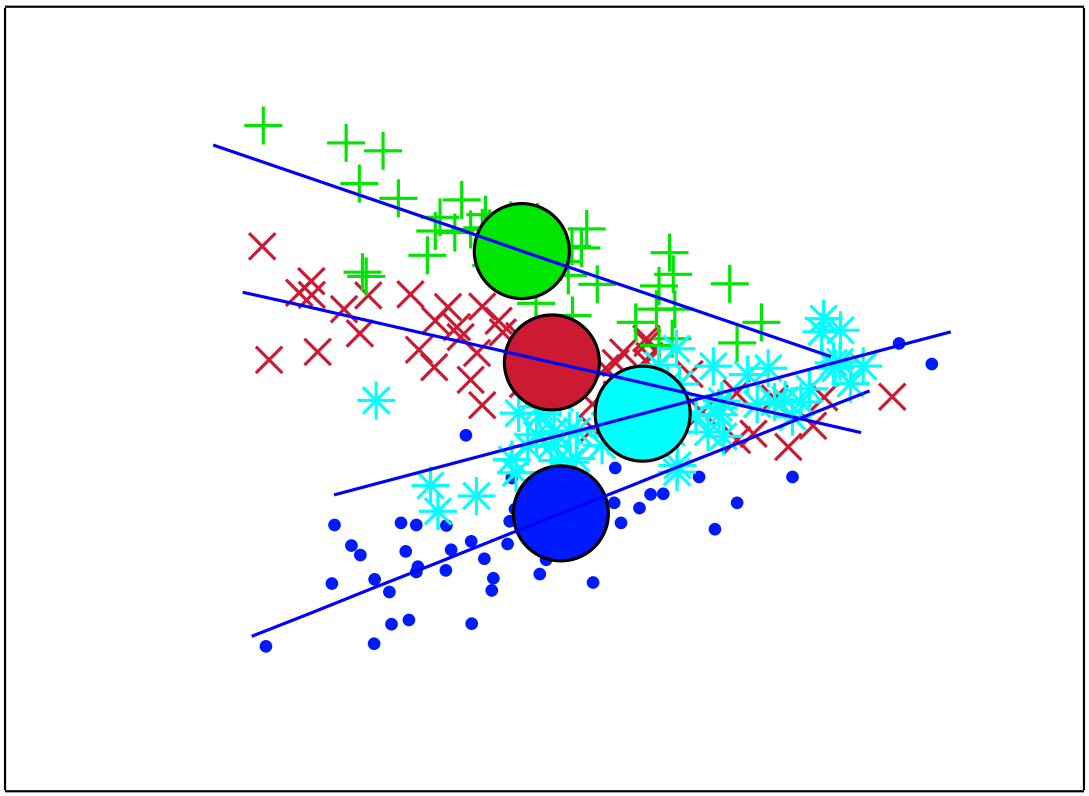


Fig. 14. Etape n° 10 de l'HDDC sur les données « Crabes ».

HDDC : les étapes de l'algorithme

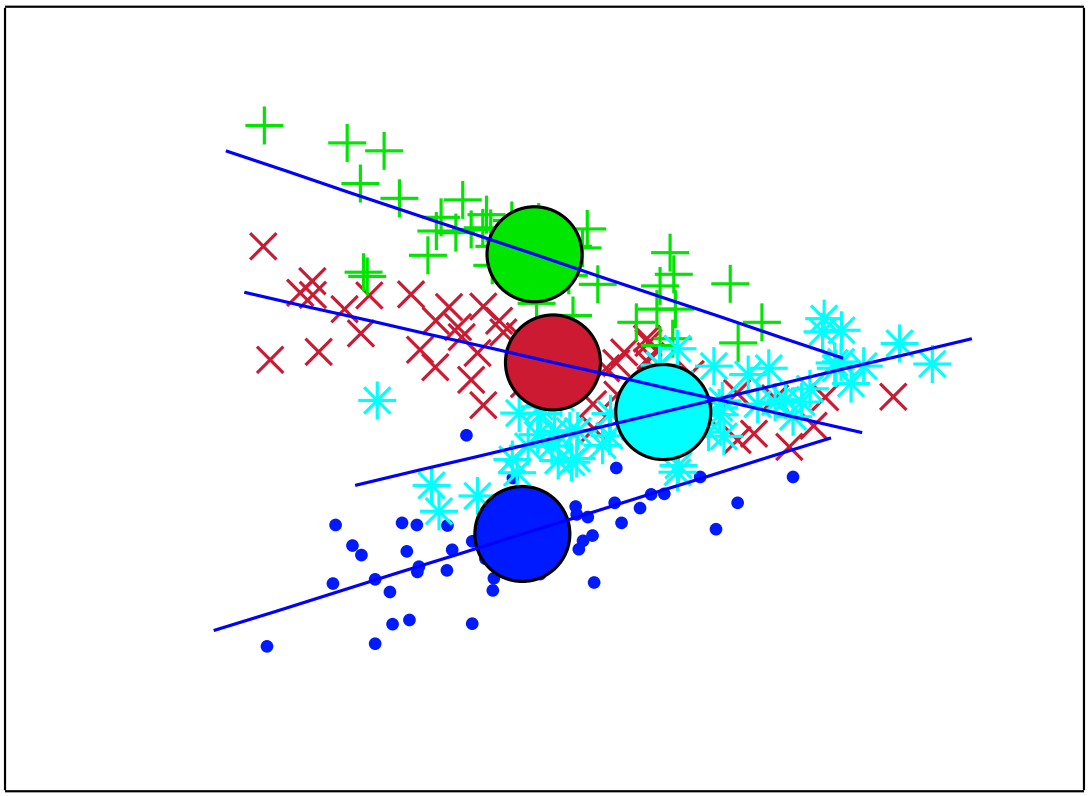


Fig. 14. Etape n° 12 de l'HDDC sur les données « Crabes ».

Application à la localisation d'objets en image

- ACI (2004–2006) avec le projet LEAR (INRIA Rhône-Alpes).
- Chaque image est représentée par environ 250 descripteurs - vecteurs de 128 caractéristiques locales (gradient, histogramme local des niveaux de gris, ...) - calculés en des points d'intérêt détectés automatiquement [Mik03].
- Au total, des milliers d'observations en dimension $p = 128$ à classer en $k = 2$ catégories : objet/fond.
- Classification semi-supervisée : on sait si l'objet est présent ou non dans l'image, mais on ne connait pas la classe des points d'intérêt.

Application à la localisation d'objets en image

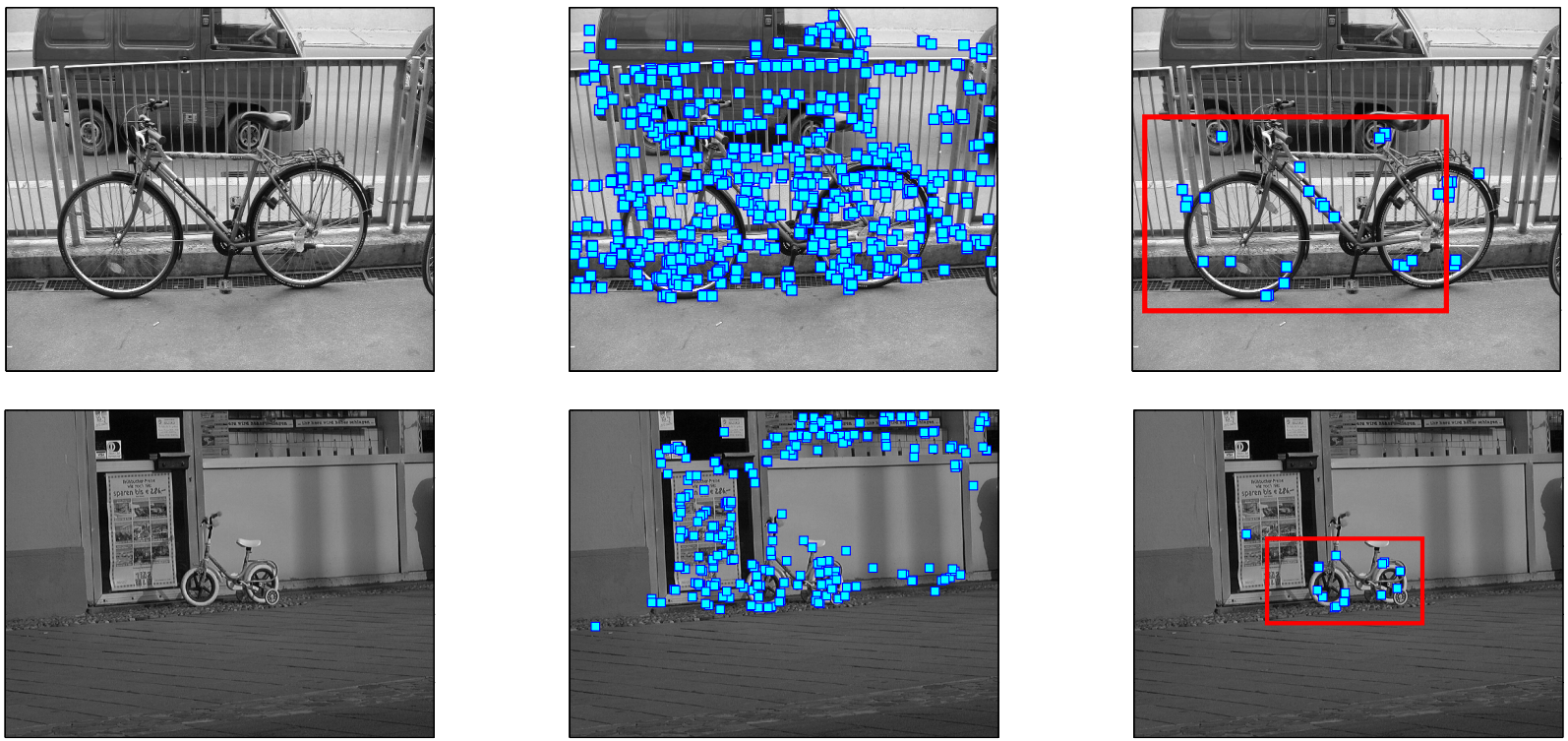


Fig. 16. Localisation de l'objet "vélo" sur des images de test.

References

– S. Girard & J. Saracco. Supervised and unsupervised classification using mixture models. In D. Fraix-Burnet and S. Girard, editors, Statistics for astrophysics, clustering and classification, volume 77, pages 69–90, EDP Sciences, 2016.
– M. Fauvel, C. Bouveyron & S. Girard, Parsimonious Gaussian process models for the classification of hyperspectral remote sensing images, *IEEE Geoscience and Remote Sensing Letters*, 12, 2423–2427, 2015.
– L. Bergé, C. Bouveyron & S. Girard. HDclassif : An R package for model-based clustering and discriminant analysis of high-dimensional data, *Journal of Statistical Software*, 46, 1–29, 2012.
– C. Bouveyron & S. Girard. Robust supervised classification with mixture models : Learning from data with uncertain labels, *Pattern Recognition*, 42, 2649–2658, 2009.
– C. Bouveyron, S. Girard & C. Schmid. High Dimensional Data Clustering, *Computational Statistics and Data Analysis*, 52, 502–519, 2007.
– C. Bouveyron, S. Girard & C. Schmid. High-dimensional discriminant Analysis, *Communication in Statistics - Theory and Methods*, 36, 2607–2623, 2007.

Plan

1 Classification en grande dimension

2 Régression en grande dimension

Multivariate regression

Let $Y \in \mathbb{R}$ and $X \in \mathbb{R}^p$. The goal is to estimate $G : \mathbb{R}^p \to \mathbb{R}$ such that

$$Y = G(X) + \xi \quad \text{where } \xi \text{ is independent of } X.$$

- Unrealistic when p is large (*curse of dimensionality*).
- **Dimension reduction** : Replace X by its projection on a subspace of lower dimension without loss of information on the distribution of Y given X.
- **Central subspace** : smallest subspace S such that, conditionally on the projection of X on S, Y and X are independent.

Dimension reduction

- Assume (for the sake of simplicity) that $\dim(S) = 1$ *i.e.* $S =$span(b), with $b \in \mathbb{R}^p \Longrightarrow$ **Single index model :**

$$Y = g(b^t X) + \xi$$

  where ξ is independent of X.
- The estimation of the p-variate function G is replaced by the estimation of the univariate function g and of the direction b.
- **Goal of SIR** [Li, 1991] : Estimate a basis of the central subspace. (*i.e.* b *in this particular case.*)

SIR

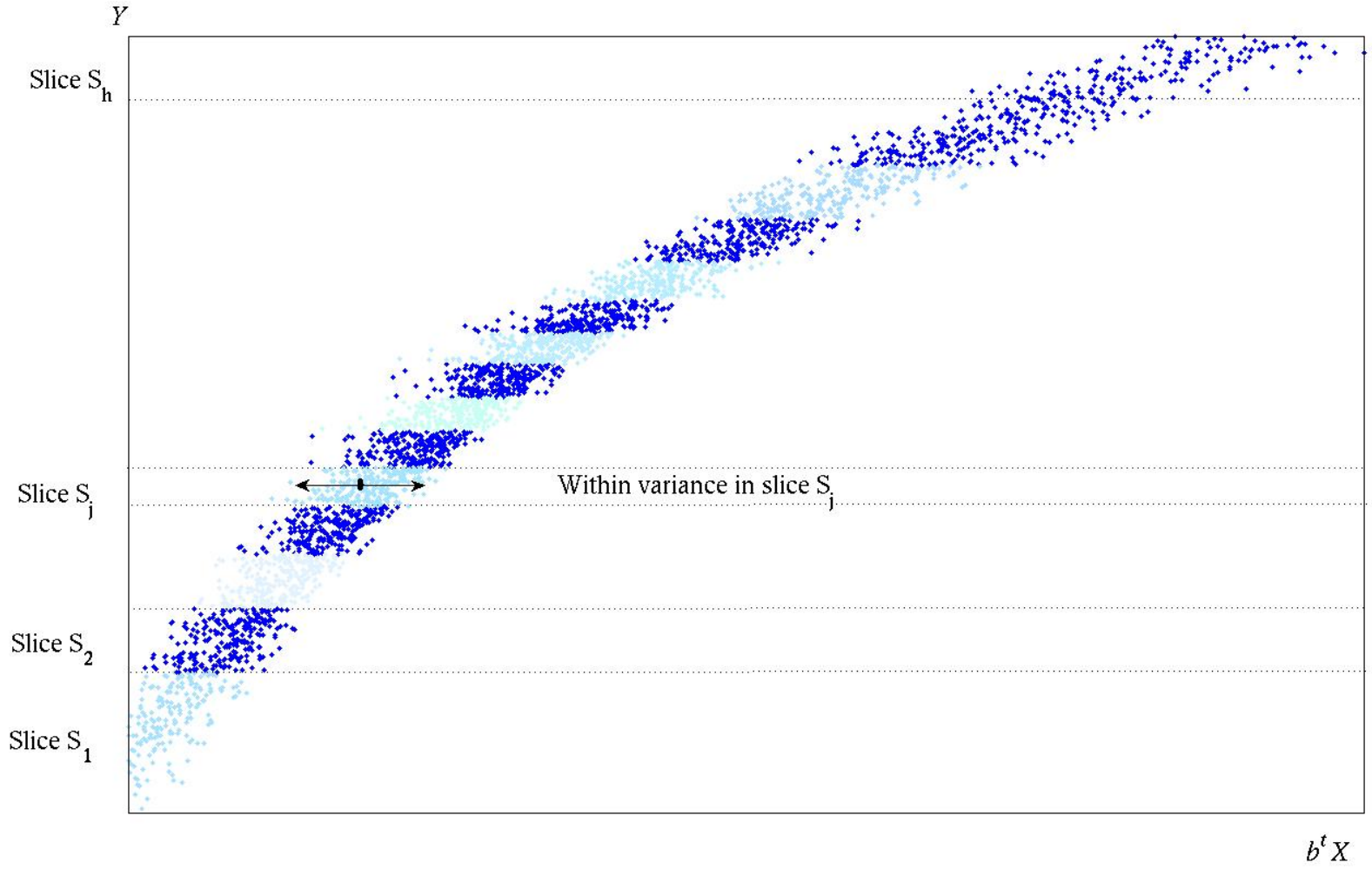
Idea :

- Find the direction b such that $b^t X$ best explains Y.
- Conversely, when Y is fixed, $b^t X$ should not vary.
- Find the direction b minimizing the variations of $b^t X$ given Y.

In practice :

- The support of Y is divided into h slices S_j.
- Minimization of the within-slice variance of $b^t X$ under the constraint $\text{var}(b^t X) = 1$.
- Equivalent to maximizing the between-slice variance under the same constraint.

Illustration



Estimation procedure

Given a sample $\{(X_1, Y_1), \ldots, (X_n, Y_n)\}$, the direction b is estimated by

$$\hat{b} = \underset{b}{\operatorname{argmax}}\, b^t \hat{\Gamma} b \text{ such that } b^t \hat{\Sigma} b = 1. \quad (1)$$

where $\hat{\Sigma}$ is the empirical covariance matrix and $\hat{\Gamma}$ is the between-slice covariance matrix defined by

$$\hat{\Gamma} = \sum_{j=1}^{h} \frac{n_j}{n} (\bar{X}_j - \bar{X})(\bar{X}_j - \bar{X})^t, \quad \bar{X}_j = \frac{1}{n_j} \sum_{Y_i \in S_j} X_i,$$

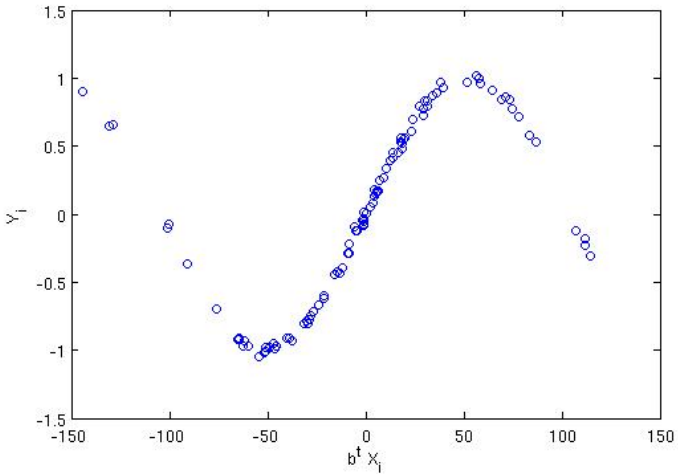
where n_j is the number of observations in the slice S_j.
The optimization problem (1) has a closed-form solution : $\hat{b}$ is the solution of the generalized eigenvector problem $\hat{\Gamma} b = \lambda \hat{\Sigma} b$.

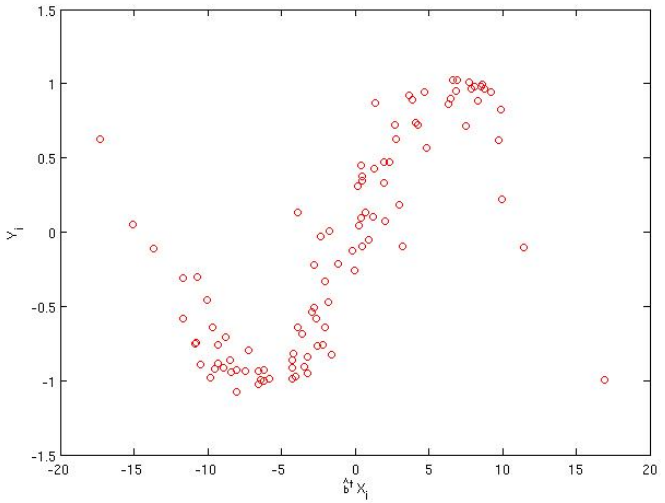
Illustration

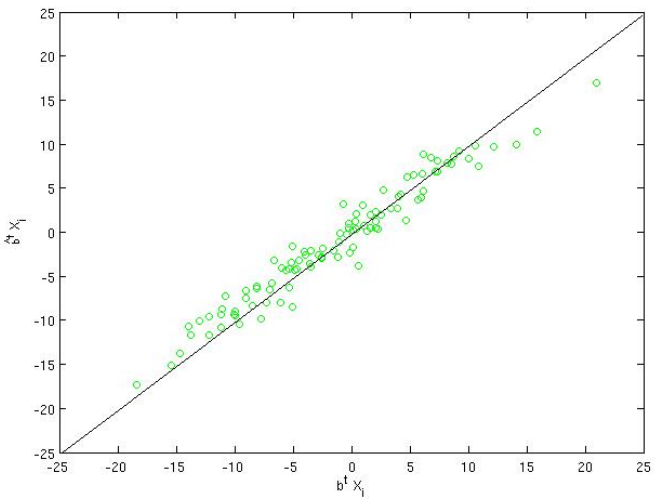
Simulated data.

- Sample $\{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ of size $n = 100$ with $X_i \in \mathbb{R}^{10}$ and $Y_i \in \mathbb{R}$, $i = 1, \ldots, n$.
- $X_i \sim \mathcal{N}_{10}(0, \Sigma)$ where $\Sigma = Q\Delta Q^t$ with
  - $\Delta =$diag$(10^2, \ldots, 2^2, 1^2)$,
  - Q is an orientation matrix drawn from the uniform distribution on the set of orthogonal matrices.
- $Y_i = g(b^t X_i) + \xi$ where
  - g is the link function $g(t) = \sin(\pi t/2)$,
  - b is the true direction $b = 5^{-1/2} Q(1, 1, 1, 1, 1, 0, 0, 0, 0, 0)^t$,
  - $\xi \sim \mathcal{N}_1(0, 9.10^{-4})$

Results in dimension $p = 10$ with $n = 100$ data







Blue : Y_i versus the projections $b^t X_i$ on the true direction b,

Red : Y_i versus the projections $\hat{b}^t X_i$ on the estimated direction $\hat{b}$,

Green : $\hat{b}^t X_i$ versus $b^t X_i$.

SIR for data streams

- We consider data arriving sequentially by blocks in a stream.
- Each data block $j = 1, \ldots, J$ is an i.i.d. sample (X_i, Y_i), $i = 1, \ldots, n$ from the single index model.
- **Goal** : Update the direction estimated on blocks $j = 1, \ldots, J-1$ at each arrival of a new Jth block of observations.

Method

- Compute the **individual directions** $\hat{b}_j$ on each block $j = 1, \ldots, J$ using SIR.
- Compute a **common direction** as

$$\hat{b} = \underset{||b||=1}{\operatorname{argmax}} \sum_{j=1}^{J} \cos^2(\hat{b}_j, b) \cos^2(\hat{b}_j, \hat{b}_J).$$

Idea : If $\hat{b}_j$ is close to $\hat{b}_J$ then $\hat{b}$ should be close to $\hat{b}_j$.
Explicit solution : $\hat{b}$ is the eigenvector associated to the largest eigenvalue of

$$M_J = \sum_{j=1}^{J} \hat{b}_j \hat{b}_j^t \cos^2(\hat{b}_j, \hat{b}_J).$$

Advantages of SIRdatastream

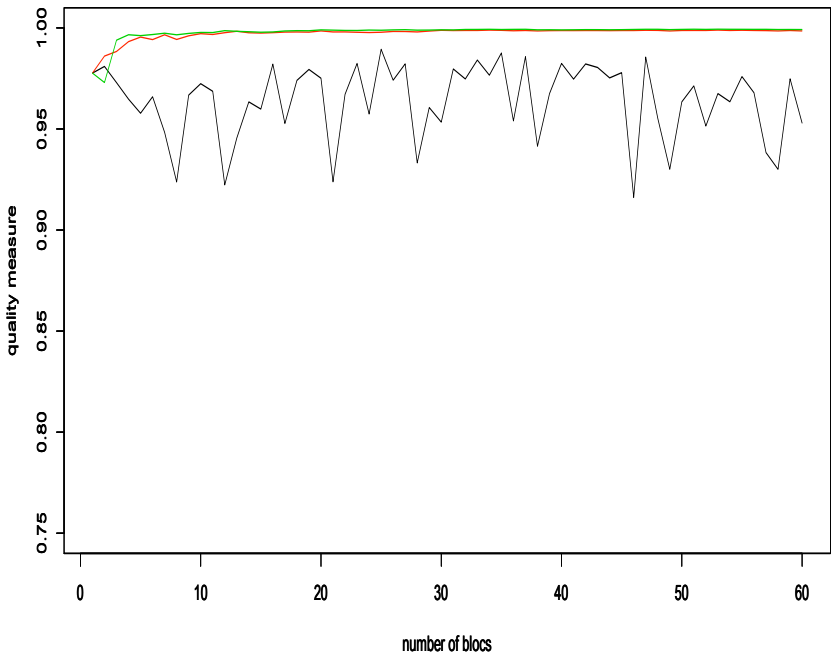
- Computational complexity $O(Jnp^2)$ v.s. $O(J^2np^2)$ for the brute-force method which would consist in applying SIR on the union of the j first blocks for $j = 1, \ldots, J$.
- Data storage $O(np)$ v.s. $O(Jnp)$ for the brute-force method.

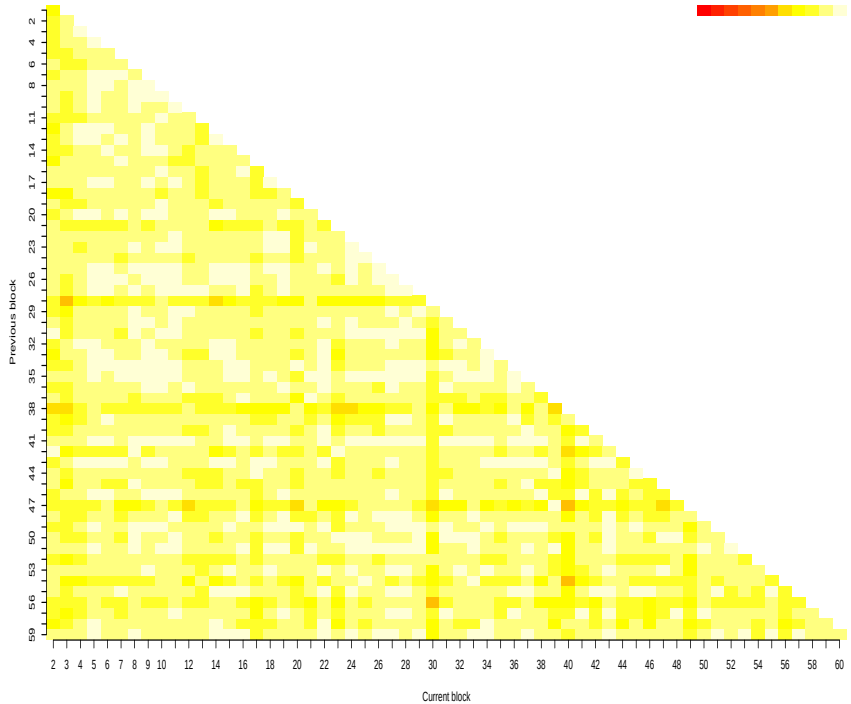
(under the assumption $n >> \max(J, p)$).

- Interpretation of the weights $\cos^2(\hat{b}_j, \hat{b}_J)$.

Illustration on simulations

Scenario 1 : A common direction in all the 60 blocks.

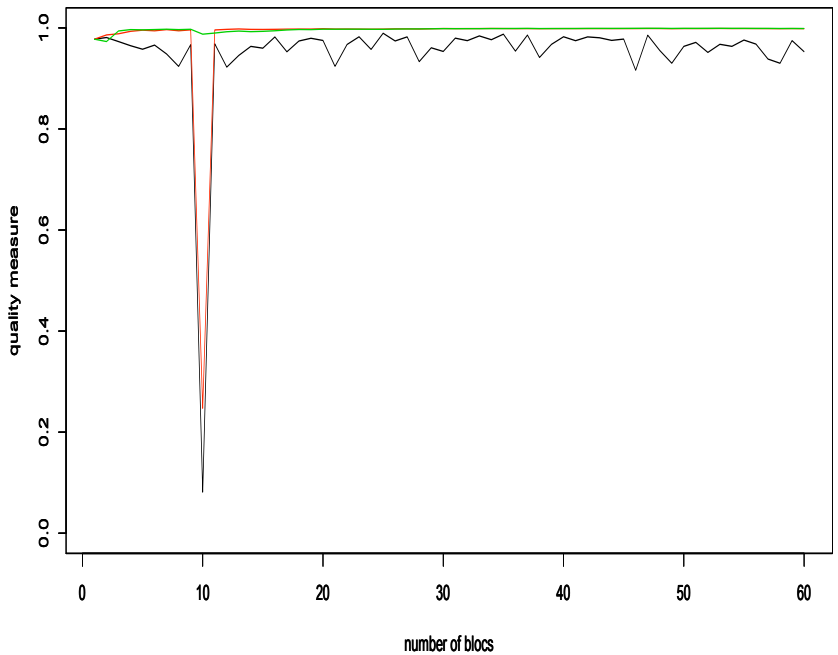


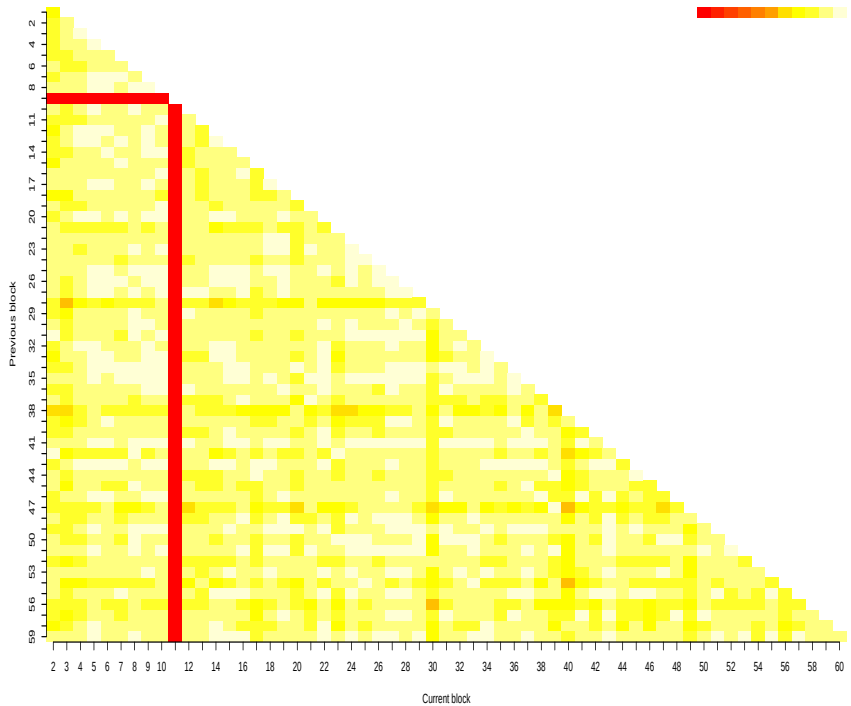


Left : $\cos^2(\hat{b}, b)$ for SIRdatastream, SIR brute-force and SIR estimators at each time t. *Right :* $\cos^2(\hat{b}_j, \hat{b}_J)$. The lighter (yellow) is the color, the larger is the weight. Red color stands for very small squared cosines.

Illustration on simulations

Scenario 2 : The 10th block is an outlier.

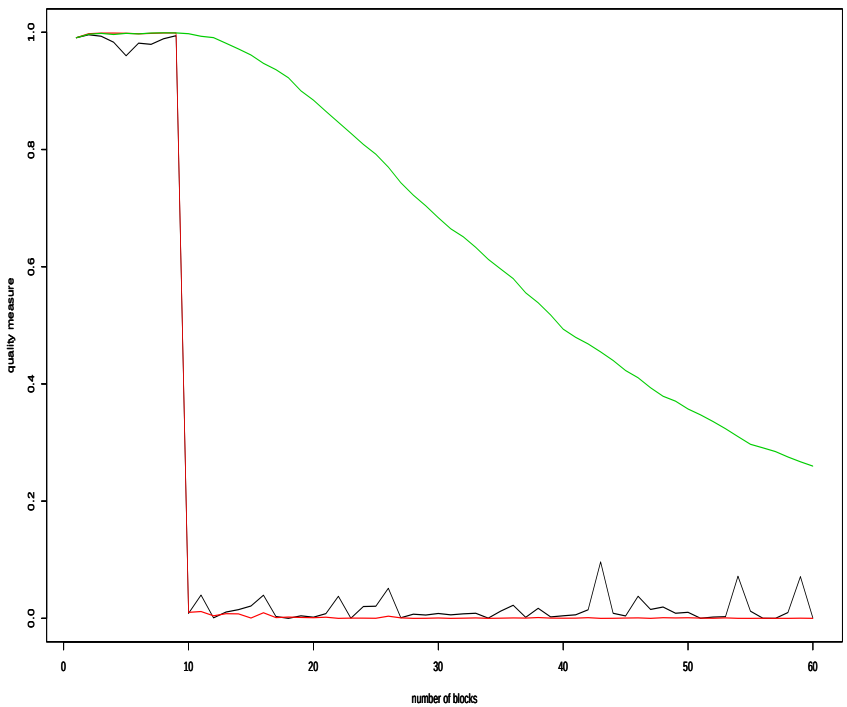


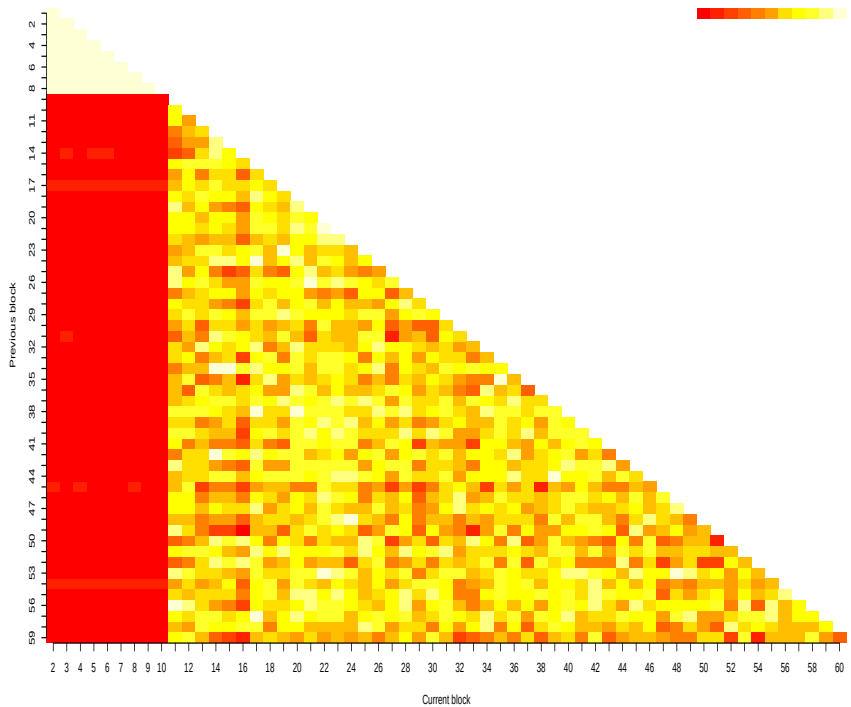


Left : $\cos^2(\hat{b}, b)$ for SIRdatastream, SIR brute-force and SIR estimators at each time t. *Right :* $\cos^2(\hat{b}_j, \hat{b}_J)$. The lighter (yellow) is the color, the larger is the weight. Red color stands for very small squared cosines.

Illustration on simulations

Scenario 3 : A drift occurs from the 10th block (b to b')

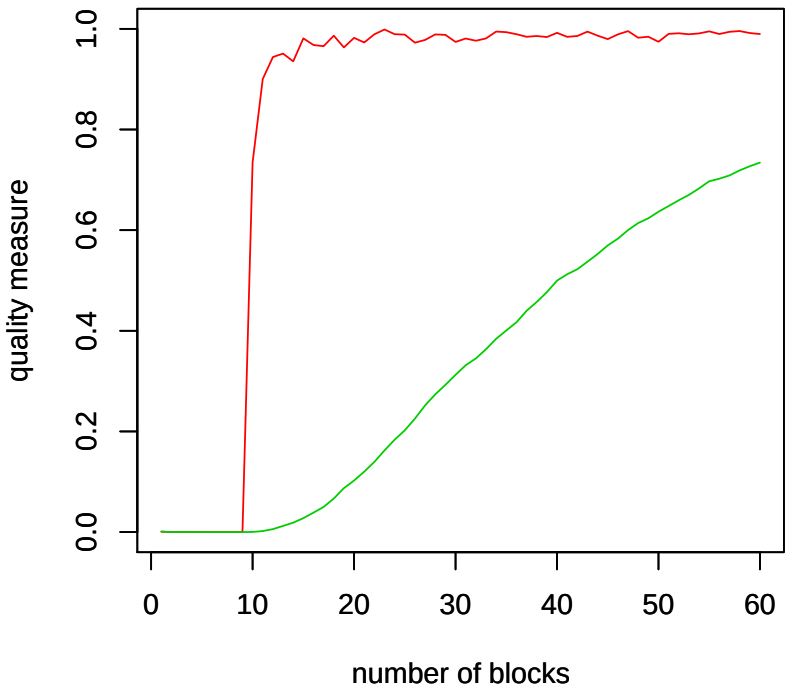


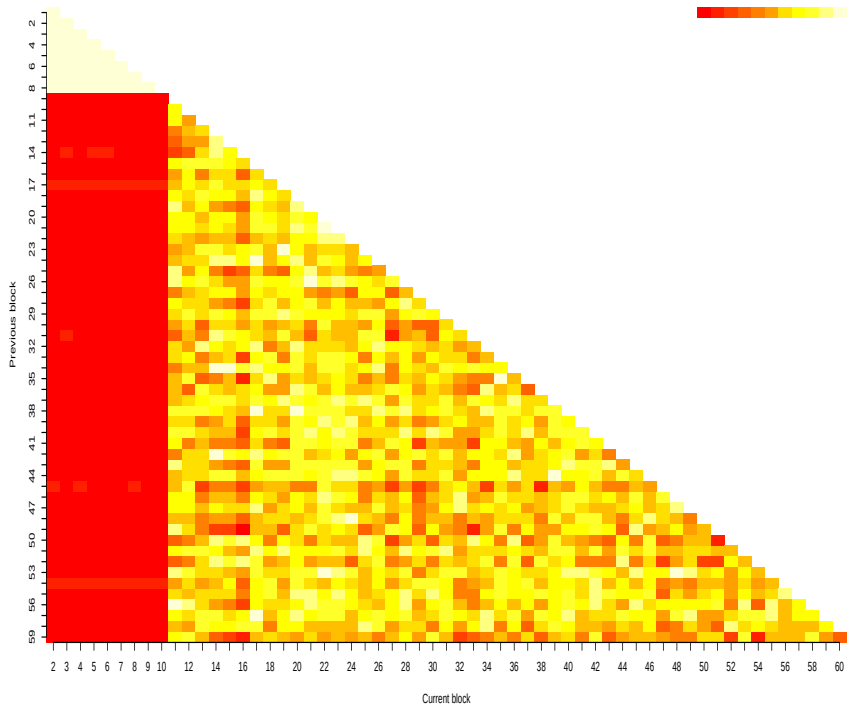


Left : $\cos^2(\hat{b}, b)$ for SIRdatastream, SIR brute-force and SIR estimators at each time t. *Right :* $\cos^2(\hat{b}_j, \hat{b}_J)$. The lighter (yellow) is the color, the larger is the weight. Red color stands for very small squared cosines.

Illustration on simulations

Scenario 3 (cont'd) :A drift occurs from the 10th block (b to b')





Left : $\cos^2(\hat{b}, b')$ for SIRdatastream and SIR brute-force. *Right :* $\cos^2(\hat{b}, b')$

Estimation of Mars surface physical properties from hyperspectral images

Context :

- Observation of the south pole of Mars at the end of summer, collected during orbit 61 by the French imaging spectrometer OMEGA on board Mars Express Mission.
- 3D image : On each pixel, a spectra containing $p = 184$ wavelengths is recorded.
- This portion of Mars mainly contains water ice, CO_2 ice and dust.

Goal : For each spectra $X \in \mathbb{R}^p$, estimate the corresponding physical parameter $Y \in \mathbb{R}$ (grain size of CO_2 ice).

An inverse problem

Forward problem.

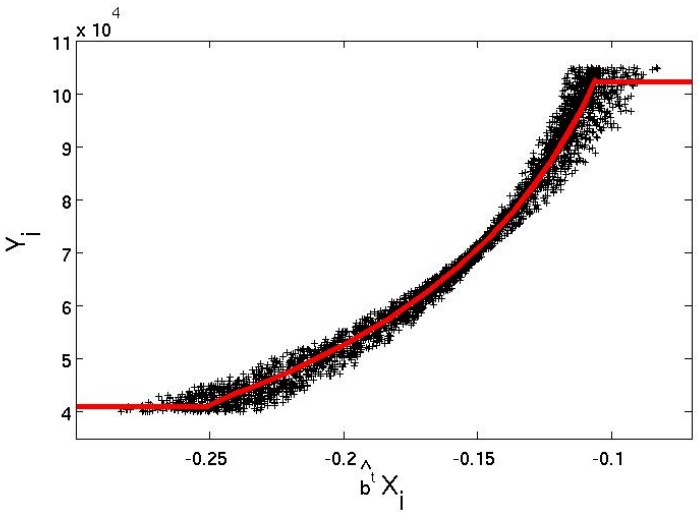
- Physical modeling of individual spectra with a surface reflectance model.
- Starting from a physical parameter Y, simulate $X = F(Y)$.
- Generation of $n = 12,000$ synthetic spectra with the corresponding parameters.

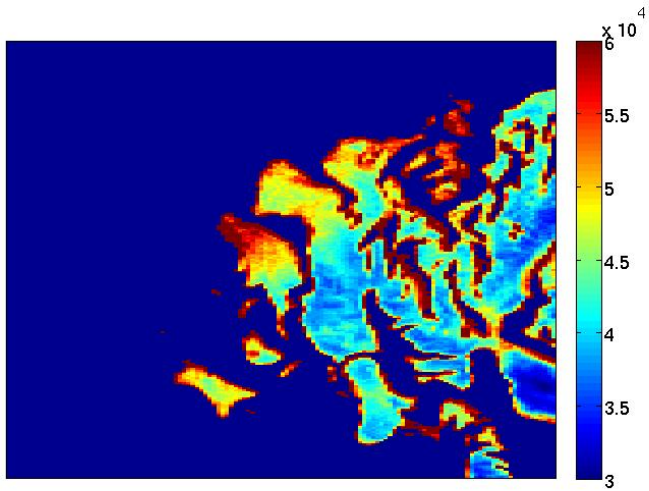
$\Longrightarrow$ Learning database.

Inverse problem.

- Estimate the functional relationship $Y = G(X)$.
- Dimension reduction assumption $G(X) = g(b^t X)$.
- b is estimated by SIR, g is estimated by a nonparametric one-dimensional regression.

Estimated CO_2 maps





Grain size of CO_2 ice estimated with SIR on a hyperspectral image of Mars.

References

– A. Chiancone, F. Forbes & S. Girard. Student Sliced Inverse Regression, *Computational Statistics and Data Analysis*, to appear.
– S. Girard & J. Saracco. An introduction to dimension reduction in nonparametric kernel regression, In D. Fraix-Burnet and D. Valls-Gabaud, editors, Regression methods for astrophysics, volume 66, pages 167–196, EDP Sciences, 2014.
– R. Coudret, S. Girard, & J. Saracco. A new sliced inverse regression method for multivariate response, *Computational Statistics and Data Analysis*, 77, 285–299, 2014.
– M. Chavent, S. Girard, V. Kuentz, B. Liquet, T.M.N. Nguyen & J. Saracco. A sliced inverse regression approach for data stream, *Computational Statistics*, 29, 1129–1152, 2014.
– C. Bernard-Michel, S. Douté, M. Fauvel, L. Gardes & S. Girard. Retrieval of Mars surface physical properties from OMEGA hyperspectral images using Regularized Sliced Inverse Regression, *Journal of Geophysical Research - Planets*, 114, E06005, 2009.
– C. Bernard-Michel, L. Gardes & S. Girard. Gaussian Regularized Sliced Inverse Regression, *Statistics and Computing*, 19, 85–98, 2009.