Algorithme de Chemin de Régularisation pour l'apprentissage Statistique

Karina Zapién Arreola

Directeur de thèse : Stéphane CANU
Encadrant : Gilles GASSO

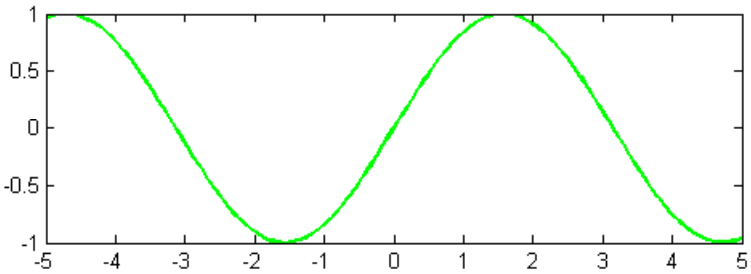
Laboratoire d'Informatique, Traitement de l'Information et des Systèmes EA 4108
INSA de Rouen, France

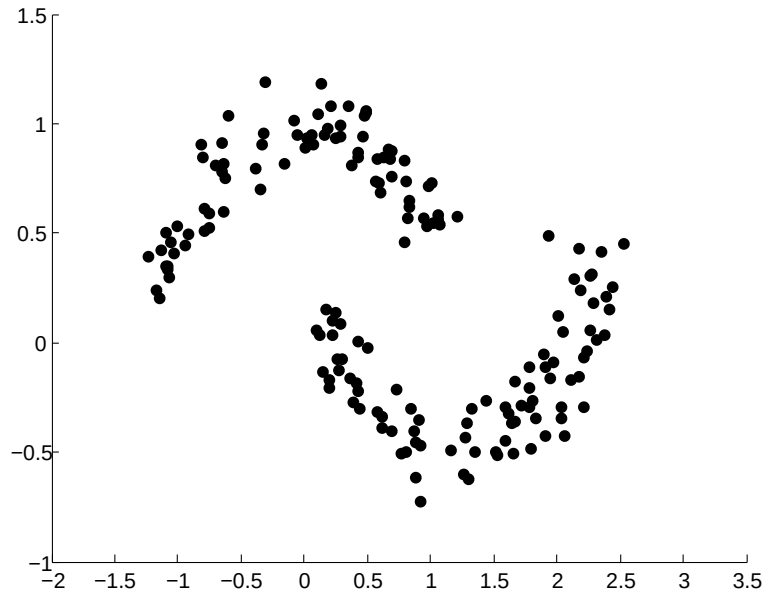
9 Juillet 2009

Plan de la présentation

Un ***problème d'apprentissage*** comprend des :

- Échantillons (images, textes, mesures, ...)



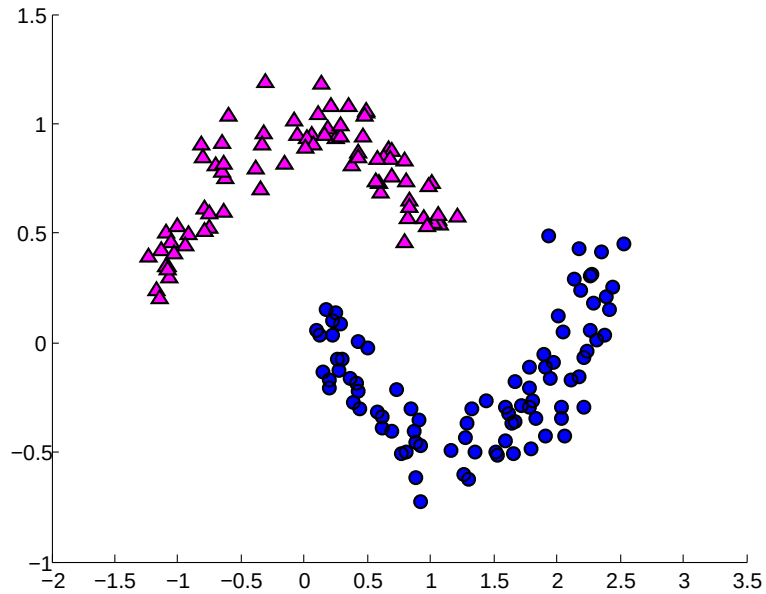


Un ***problème d'apprentissage*** comprend des :

- Échantillons (images, textes, mesures, ...)

$$S_X = \{\mathbf{x}_i\}_{i=[\![n]\!]},\ \mathbf{x}_i \in \mathcal{X}$$

Notation : $[\![n]\!] \equiv 1, \ldots, n$, $\mathcal{X}$ un espace vectoriel et $\mathcal{Y}$ un espace normé.



Un ***problème d'apprentissage*** comprend des :

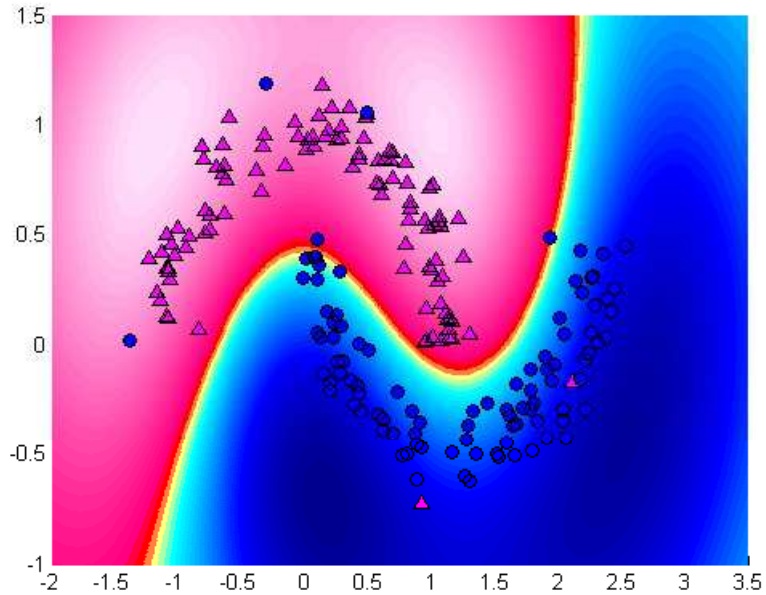
- **Échantillons** (images, textes, mesures, ...)

$$S_X = \{\mathbf{x}_i\}_{i=[\![n]\!]},\ \mathbf{x}_i \in \mathcal{X}$$

- **Étiquettes** associées (classes, rangs, ...)

$$S_Y = \{y_i\}_{i=[\![n]\!]},\ y_i \in \mathcal{Y}$$

Notation : $[\![n]\!] \equiv 1, \ldots, n$, $\mathcal{X}$ un espace vectoriel et $\mathcal{Y}$ un espace normé.



Un ***problème d'apprentissage*** comprend des :

- **Échantillons** (images, textes, mesures, ...)

$$S_X = \{\mathbf{x}_i\}_{i=[\![n]\!]},\ \mathbf{x}_i \in \mathcal{X}$$

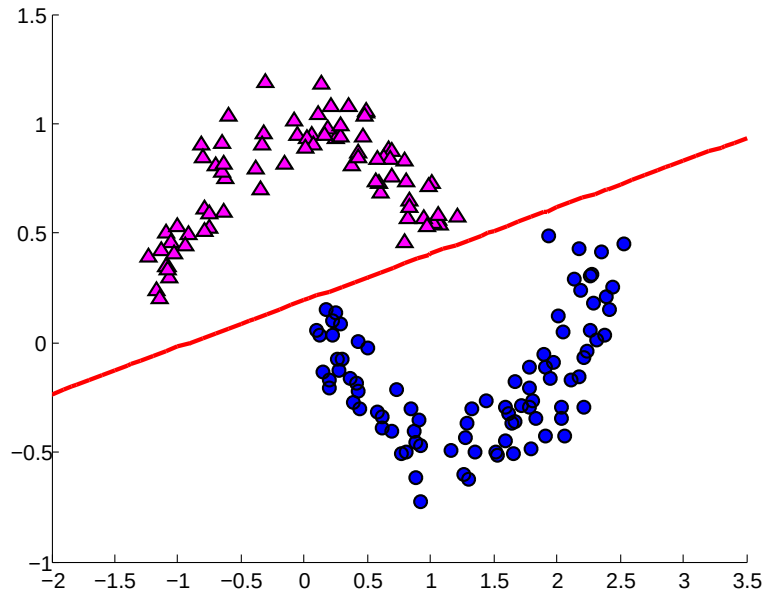
- **Étiquettes** associées (classes, rangs, ...)

$$S_Y = \{y_i\}_{i=[\![n]\!]},\ y_i \in \mathcal{Y}$$

- **Probabilité jointe** (généralement inconnue)

$$\mathbb{P}(\mathbf{x}, y), \mathbf{x} \in \mathcal{X}, y \in \mathcal{Y}$$

Notation : $[\![n]\!] \equiv 1, \ldots, n$, $\mathcal{X}$ un espace vectoriel et $\mathcal{Y}$ un espace normé.



Un ***problème d'apprentissage*** comprend des :

- **Échantillons** (images, textes, mesures, ...)

$$S_X = \{\mathbf{x}_i\}_{i=[\![n]\!]},\ \mathbf{x}_i \in \mathcal{X}$$

- **Étiquettes** associées (classes, rangs, ...)

$$S_Y = \{y_i\}_{i=[\![n]\!]},\ y_i \in \mathcal{Y}$$

- **Probabilité jointe** (généralement inconnue)

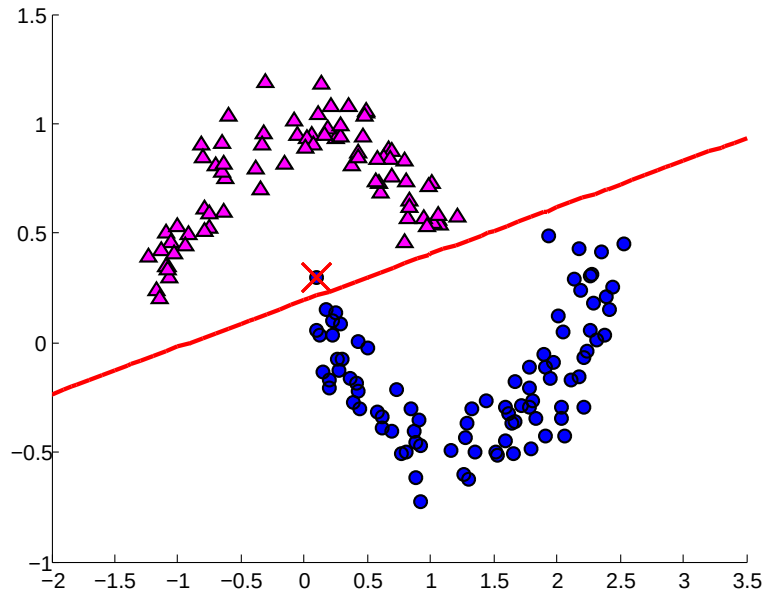
$$\mathbb{P}(\mathbf{x}, y), \mathbf{x} \in \mathcal{X}, y \in \mathcal{Y}$$

- On cherche une **fonction de décision** :

$$f(\mathbf{x}) : \mathcal{X} \to \mathcal{Y}, \qquad f \in \mathcal{H}$$

qui prédit correctement l'étiquette de x

Notation : $[\![n]\!] \equiv 1, \ldots, n$, $\mathcal{X}$ un espace vectoriel et $\mathcal{Y}$ un espace normé.



Un ***problème d'apprentissage*** comprend des :

- **Échantillons** (images, textes, mesures, ...)
$$S_X = \{\mathbf{x}_i\}_{i=[\![n]\!]},\ \mathbf{x}_i \in \mathcal{X}$$
- **Étiquettes** associées (classes, rangs, ...)
$$S_Y = \{y_i\}_{i=[\![n]\!]},\ y_i \in \mathcal{Y}$$
- **Probabilité jointe** (généralement inconnue)
$$\mathbb{P}(\mathbf{x}, y), \mathbf{x} \in \mathcal{X}, y \in \mathcal{Y}$$
- On cherche une **fonction de décision** :
$$f(\mathbf{x}) : \mathcal{X} \to \mathcal{Y}, \qquad f \in \mathcal{H}$$
qui prédit correctement l'étiquette de x
- même sur des données n'ayant pas servi à son apprentissage (**généralisation**).

Notation : $[\![n]\!] \equiv 1, \ldots, n$, $\mathcal{X}$ un espace vectoriel et $\mathcal{Y}$ un espace normé.

Espace de Fonctions à Noyau

- $\mathbf{k} : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ est une fonction à noyau sur un ensemble $\mathcal{X}$.
- $\mathbf{k}$ est symétrique si : $\mathbf{k}(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{k}(\mathbf{x}_j, \mathbf{x}_i)$.
- $\mathbf{k}$ est strictement positive si : $\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j \mathbf{k}(\mathbf{x}_i, \mathbf{x}_j) \geq 0$.

$\forall n > 0,\ \forall \{\mathbf{x}_i\}_{i=[\![n]\!]} \mathbf{x}_i \in \mathcal{X}, \{\alpha_i\}_{i=[\![n]\!]}, \alpha_i \in \mathbb{R}$ avec au moins un $\alpha_i \neq 0$.

- $\mathbf{k} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ est une **fonction à noyau** sur un ensemble $\mathcal{X}$.
- $\mathbf{k}$ est symétrique si : $\mathbf{k}(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{k}(\mathbf{x}_j, \mathbf{x}_i)$.
- $\mathbf{k}$ est strictement positive si : $\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j \mathbf{k}(\mathbf{x}_i, \mathbf{x}_j) \geq 0$.

$\forall n > 0,\ \forall \{\mathbf{x}_i\}_{i=[\![n]\!]} \mathbf{x}_i \in \mathcal{X}, \{\alpha_i\}_{i=[\![n]\!]}, \alpha_i \in \mathbb{R}$ avec au moins un $\alpha_i \neq 0$.

Propriété de reproduction

Un espace (de fonctions) de Hilbert $\mathcal{H}$ muni du produit scalaire $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ possède la **propriété de reproduction** s'il existe $\mathbf{k}$ symétrique positif tel que :

1. $\forall \mathbf{x} \in \mathcal{X}$, $\mathbf{k}(\mathbf{x}, \cdot)$ est une fonction qui appartient à $\mathcal{H}$.
2. $\forall \mathbf{x} \in \mathcal{X},\ \forall f \in \mathcal{H}$ on a : $f(\mathbf{x}) = \langle f(\cdot), \mathbf{k}(\mathbf{x}, \cdot) \rangle_{\mathcal{H}}$

$\Longrightarrow$ **espace de Hilbert à noyau reproduisant (RKHS)**.

L'espace de recherche $\mathcal{H}$ est la fermeture de :

$$\mathcal{H}_0 = \{f(\mathbf{x}) : \mathbf{f(x)} = \sum_{i=1}^{m_f} \alpha_i \mathbf{k}(\mathbf{x}, \mathbf{x}_i), \alpha_i \in \mathbb{R}, m_f \in \mathbb{N}, \mathbf{x}_i \in \mathcal{X}\}$$

muni d'une forme bilinéaire : $\langle f, g \rangle_{\mathcal{H}} = \sum_{i=1}^{m_f} \sum_{j=1}^{m_g} \alpha_i \beta_j k(\mathbf{z}_i, \mathbf{y}_j)$

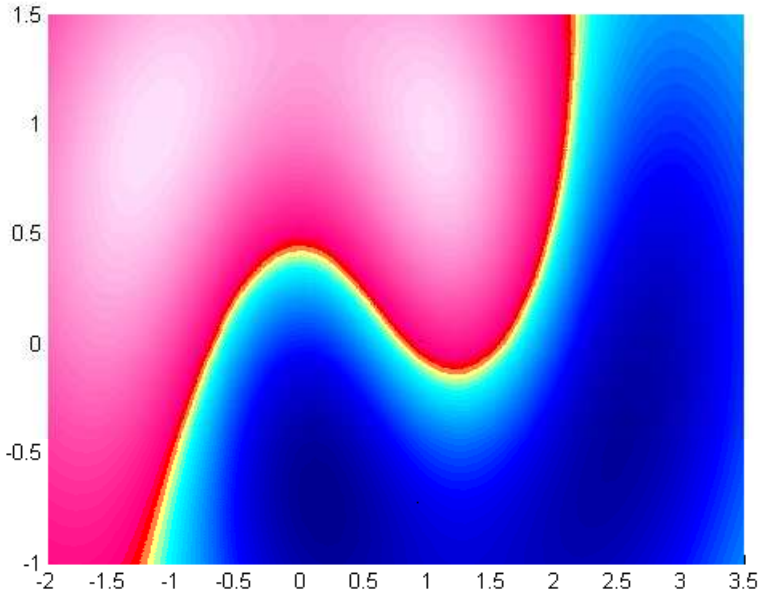
Dérivée fonctionelle

Dans l'espace $\mathcal{H}$, il est possible de calculer des gradients :

- $\nabla_f \|f\|_{\mathcal{H}}^2 = \nabla_f \langle f, f \rangle_{\mathcal{H}} = 2f(\cdot)$
- $\nabla_f f(\mathbf{x}) = \nabla_f \langle f(\cdot), \mathbf{k}(\mathbf{x}, \cdot) \rangle_{\mathcal{H}} = \mathbf{k}(\mathbf{x}, \cdot)$.

$f(\mathbf{x}) = \sum_{i=1}^{m_f} \alpha_i \mathbf{k}(\mathbf{x}, \mathbf{z}_i)$,
$g(\mathbf{x}) = \sum_{i=1}^{m_g} \beta_i \mathbf{k}(\mathbf{x}, \mathbf{y}_i)$,
$m_f, m_g \in \mathbb{N}$, $\mathbf{x}, \mathbf{y}_j, \mathbf{z}_i \in \mathcal{X}$, $\alpha_i, \beta_j \in \mathbb{R}, i = [\![m_f]\!], j = [\![m_g]\!]$

Recherche de la fonction de décision



- **Fonction de coût** :

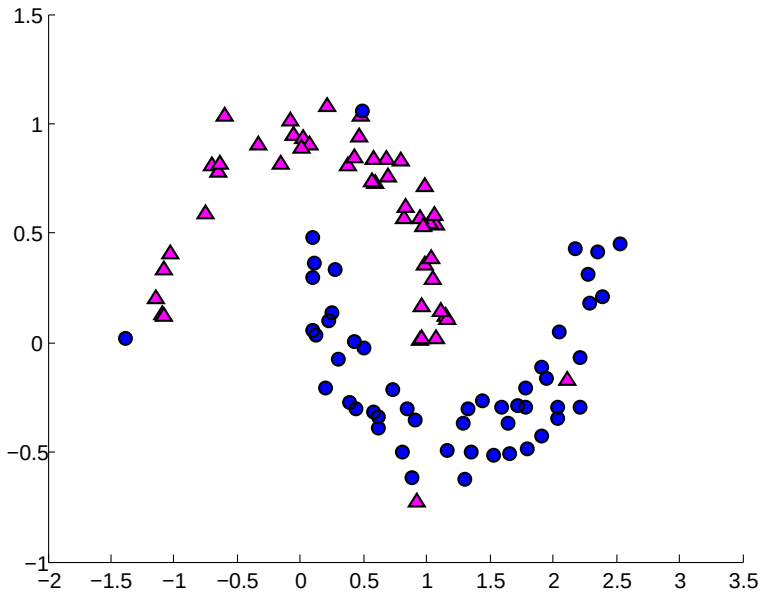
$$\ell : \mathcal{H} \times \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^+ \cup \{0\}$$

à minimiser par rapport à la loi $\mathbb{P}$.

- **Risque statistique** :

$$\mathbb{E}_{\mathcal{X}\times\mathcal{Y}}[\ell(f, \mathbf{x}, y)] = \int_{\mathcal{X}\times\mathcal{Y}} \ell(f, \mathbf{x}, \mathbf{y}) d\mathbb{P}(\mathbf{x}, y)$$

Recherche de la fonction de décision



- **Fonction de coût** :

$$\ell : \mathcal{H} \times \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^+ \cup \{0\}$$

à minimiser par rapport à la loi $\mathbb{P}$.

- **Risque statistique** :

$$\mathbb{E}_{\mathcal{X}\times\mathcal{Y}}[\ell(f, \mathbf{x}, y)] = \int_{\mathcal{X}\times\mathcal{Y}} \ell(f, \mathbf{x}, \mathbf{y}) d\mathbb{P}(\mathbf{x}, y)$$

- **Problème :** $\mathbb{P}$ est inconnue et seulement une partie des données est accessible.

- **Risque Empirique** :

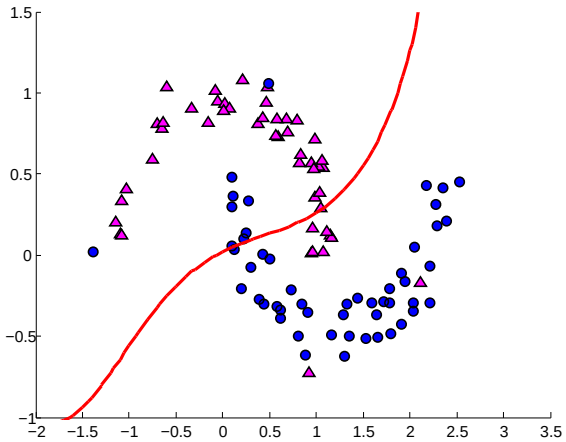
$$\mathcal{L}(f, S) = \frac{1}{n} \sum_{i=1}^{n} \ell(f, \mathbf{x}_i, y_i),$$

$$S = \{(\mathbf{x}_i, y_i)\}_{i=[\![n]\!]}$$

Sous-apprentissage

Contrôlé par

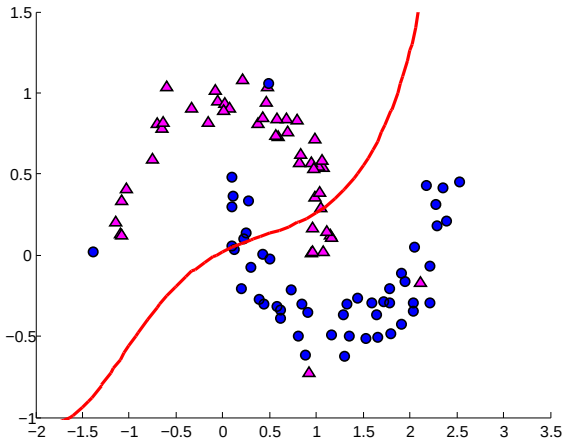
$\mathcal{L}(f, S)$



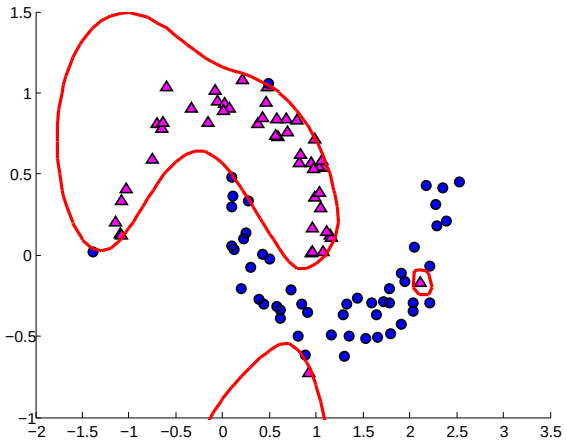
Sous-apprentissage

Contrôlé par

$\mathcal{L}(f, S)$



Sur-apprentissage

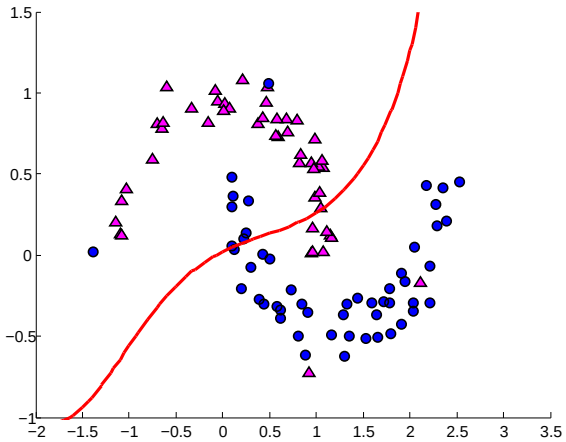


Problème d'optimisation multi-critères

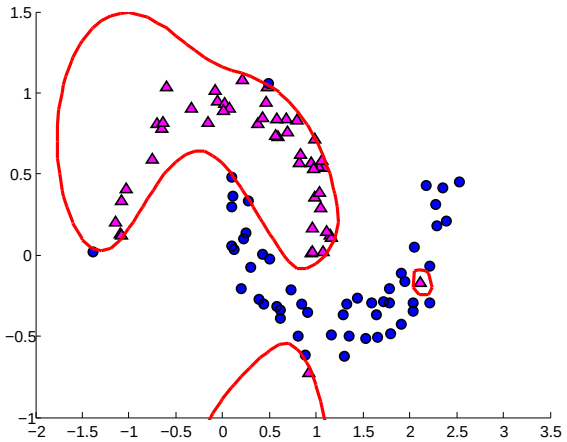
Sous-apprentissage

Contrôlé par

$\mathcal{L}(f, S)$



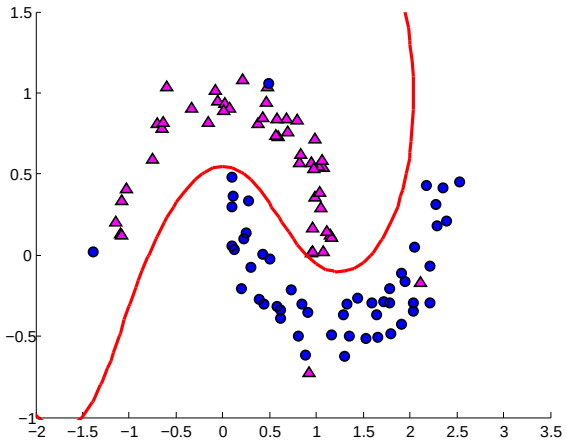
Sur-apprentissage



Complexité

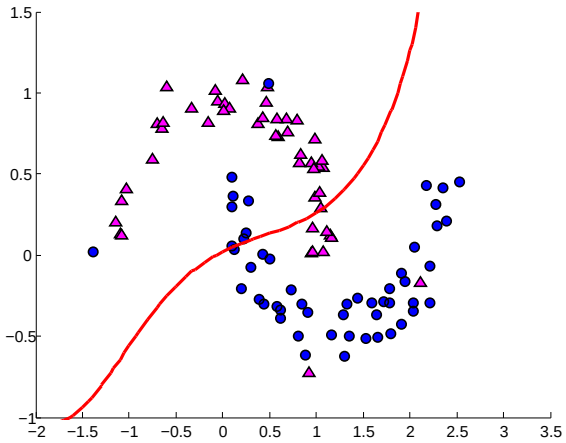
Contrôlée par

$\Omega : \mathcal{H} \rightarrow \mathbb{R}^+ \cup \{0\}$

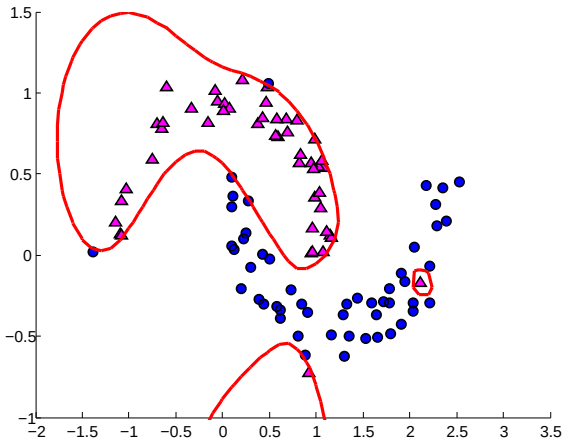


Problème d'optimisation multi-critères

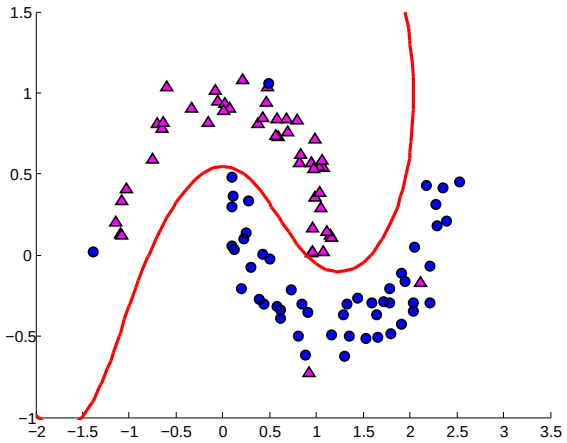
Sous-apprentissage
Contrôlé par
$\mathcal{L}(f, S)$



Sur-apprentissage



Complexité
Contrôlée par
$\Omega : \mathcal{H} \rightarrow \mathbb{R}^+ \cup \{0\}$



Problème à résoudre : trouver le meilleur compromis entre $\mathcal{L}$ et Ω

$$\hat{f} = \underset{f \in \mathcal{H}}{\text{argmin}} \left\{ \begin{array}{l} \mathcal{L}(f, S) \\ \Omega(f) \end{array} \right.$$

Différentes formulations du problème

$$\begin{cases} \min\limits_{f \in \mathcal{H}} & \mathcal{L}(f, S) \\ \text{s. c.} & \Omega(f) \leq C \end{cases}$$

Régularisation d'Ivanov

$$\begin{cases} \min\limits_{f \in \mathcal{H}} & \Omega(f) \\ \text{s. c.} & \mathcal{L}(f, S) \leq C' \end{cases}$$

Régularisation de Morozov

$$\min_{f \in \mathcal{H}} \mathcal{L}(f, S) + \lambda \Omega(f)$$

Régularisation de Thikonov

Remarque

Les 3 formulations sont équivalentes si $\mathcal{L}$ et Ω sont convexes

Objectif

Sélection du bon modèle $\Leftrightarrow$ trouver la valeur optimale de C, C' ou λ.

Le chemin de régularisation est l'ensemble des solutions f_λ obtenues en faisant varier le paramètre λ sur $\mathbb{R}^+$ i.e. Chemin $= \{f_\lambda,\ \lambda \in [0, +\infty]\}$.

- La **fonction de décision** recherchée est la solution d'un **problème d'optimisation multi-critères**.
- On a **plusieurs formulations** pour résoudre le problème multi-critères.
- On s'intéresse au réglage du paramètre de régularisation qui contrôle le compromis entre les critères.
- Un **ensemble de validation** est nécessaire pour ce but.

Feuille de route

Discrimination : Séparateurs à Vaste Marge (SVM) : $\mathcal{Y} = \{+1, -1\}$

Fonction de coût charnière

$$\ell(f, \mathbf{x}, y) = \max\left\{0, 1 - yf(\mathbf{x})\right\}$$

Fonction de régularisation

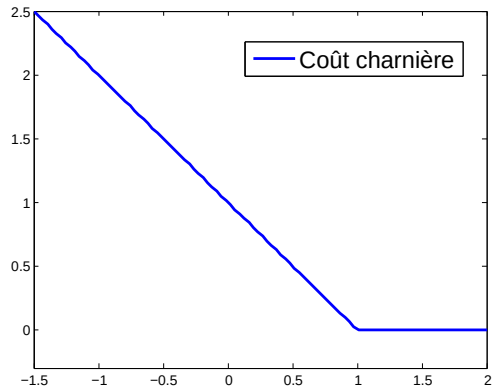
$$\Omega(f) = \|f\|_{\mathcal{H}}^2 = \langle f_0, f_0 \rangle_{\mathcal{H}}$$

$$\min_f \quad \sum_{i=1}^{n} \max\{0, 1 - y_i f(\mathbf{x}_i)\} + \frac{\lambda}{2}\|f\|_{\mathcal{H}}^2$$

Fonction de décision

$f(\mathbf{x}) = f_0(\mathbf{x}) + b$

$f_0 \in \mathcal{H}$ avec $\mathcal{H}$ un RKHS et $b \in \mathbb{R}$



Le problème peut être résolu avec le Lagrangien.

Mon approche de la solution utilise la sous-différentielle.

Sous-différentielle

$g(\mathbf{z}) : \mathcal{X} \rightarrow \mathbb{R}$ convexe. La **sous-différentielle** de g en $\mathbf{z}_0 \in \mathcal{X}$ est l'ensemble :

$$\partial_{\mathbf{z}} g(\mathbf{z}_0) = \{\boldsymbol{\nu} \in \mathcal{X} : g(\mathbf{z}) - g(\mathbf{z}_0) \geq \langle \boldsymbol{\nu}, \mathbf{z} - \mathbf{z}_0 \rangle, \forall \mathbf{z} \in \mathcal{X}\}.$$

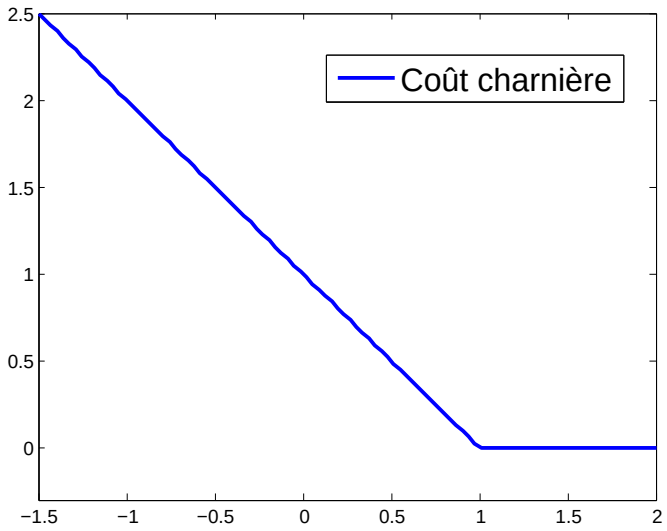
Un élément ν de l'ensemble est un **sous-gradient**.

Condition d'optimalité [Schi 07]

Pour une fonction convexe $g(\mathbf{z})$, on a :

$$g(\mathbf{z}^*) = \min_{\mathbf{z} \in \mathcal{X}} g(\mathbf{z}) \quad \Longleftrightarrow \quad 0 \leq g(\mathbf{z}) - g(\mathbf{z}^*) \; \forall \mathbf{z} \in \mathcal{X} \quad \Longleftrightarrow \quad 0 \in \partial g(\mathbf{z}^*).$$

Conditions d'optimalité : Discrimination



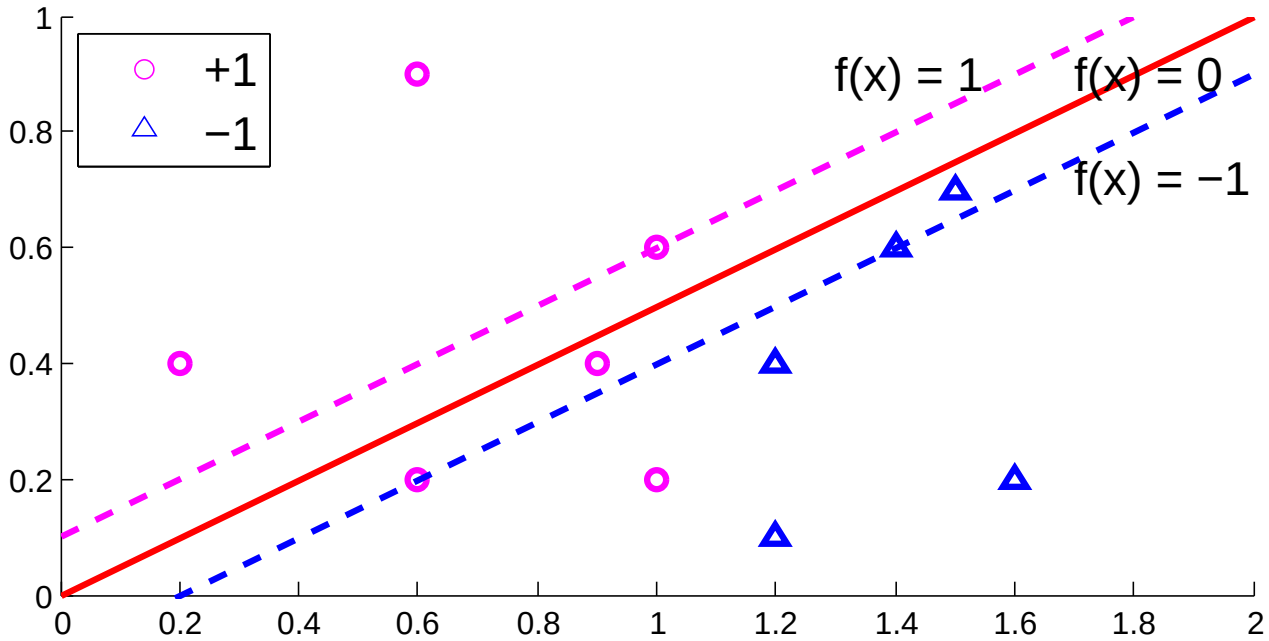
Pour $\ell(f, \mathbf{x}_i, y_i) = \max\{0, 1 - y_i f(\mathbf{x}_i)\}$, ν_i est un sous-gradient de ℓ :

$$\partial_{y_i f(\mathbf{x}_i)} \ell(f, \mathbf{x}_i, y_i) = \begin{cases} \nu_i, & -1 \leq \nu_i \leq 0 \quad \text{if} \quad y_i f(\mathbf{x}_i) = 1 \\ \nu_i & \nu_i = -1, \quad \text{if} \quad y_i f(\mathbf{x}_i) < 1, \\ \nu_i & \nu_i = 0, \quad \text{if} \quad y_i f(\mathbf{x}_i) > 1, \end{cases}$$

Pour faciliter la notation après, soit $\alpha_i = -\nu_i$ c.à.d.

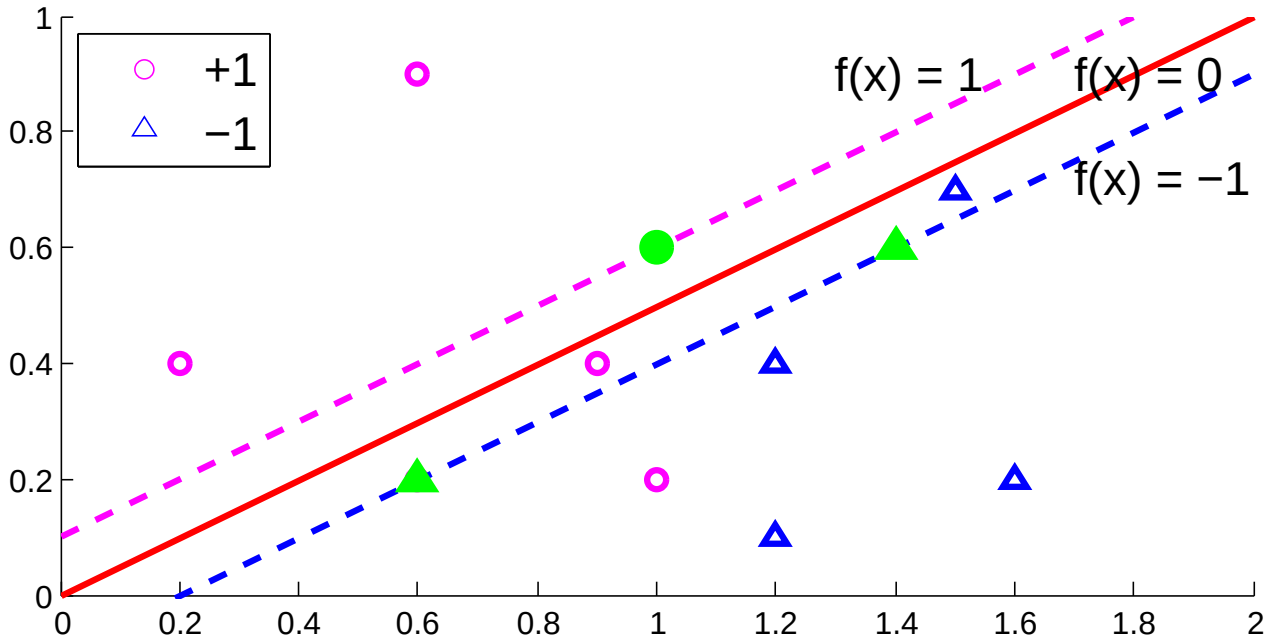
$$0 \leq \alpha_i \leq 1 \quad \forall\ i = [\![n]\!].$$

On analyse ℓ par rapport à sa différentiabilité :



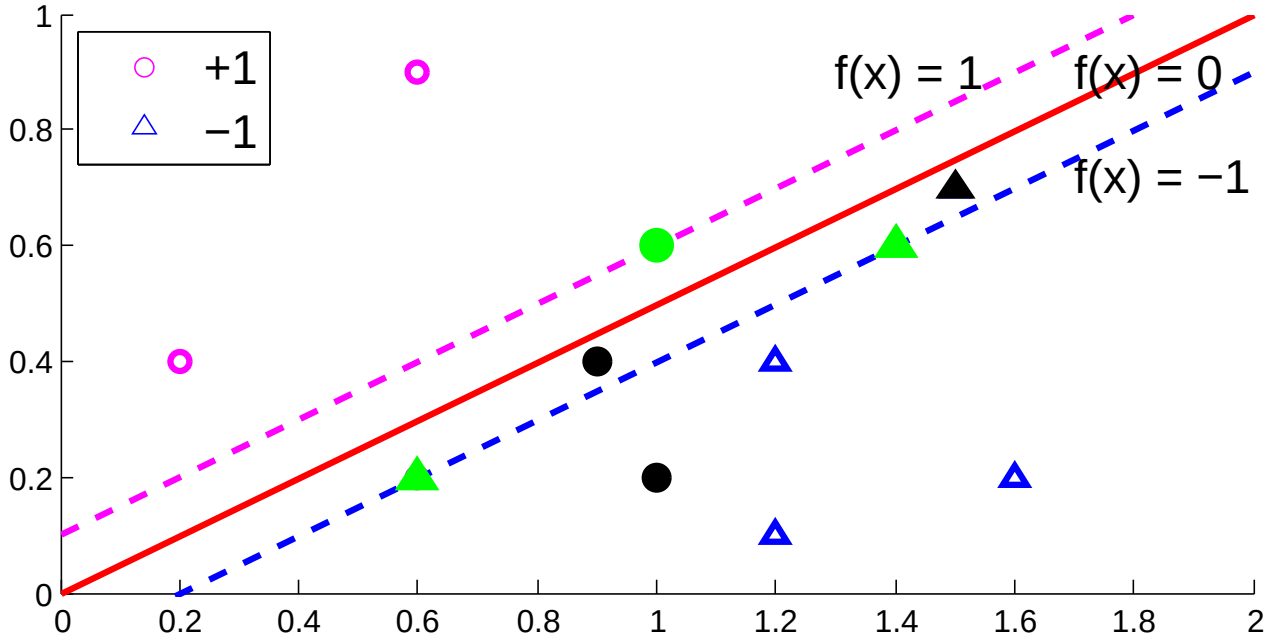
On analyse ℓ par rapport à sa différentiabilité :

- $\mathcal{I}_\alpha = \{i : y_i f(\mathbf{x}_i) = 1\} \Rightarrow \quad 0 \leq \alpha_i \leq 1,\ i \in \mathcal{I}_\alpha$, Points sur la Marge



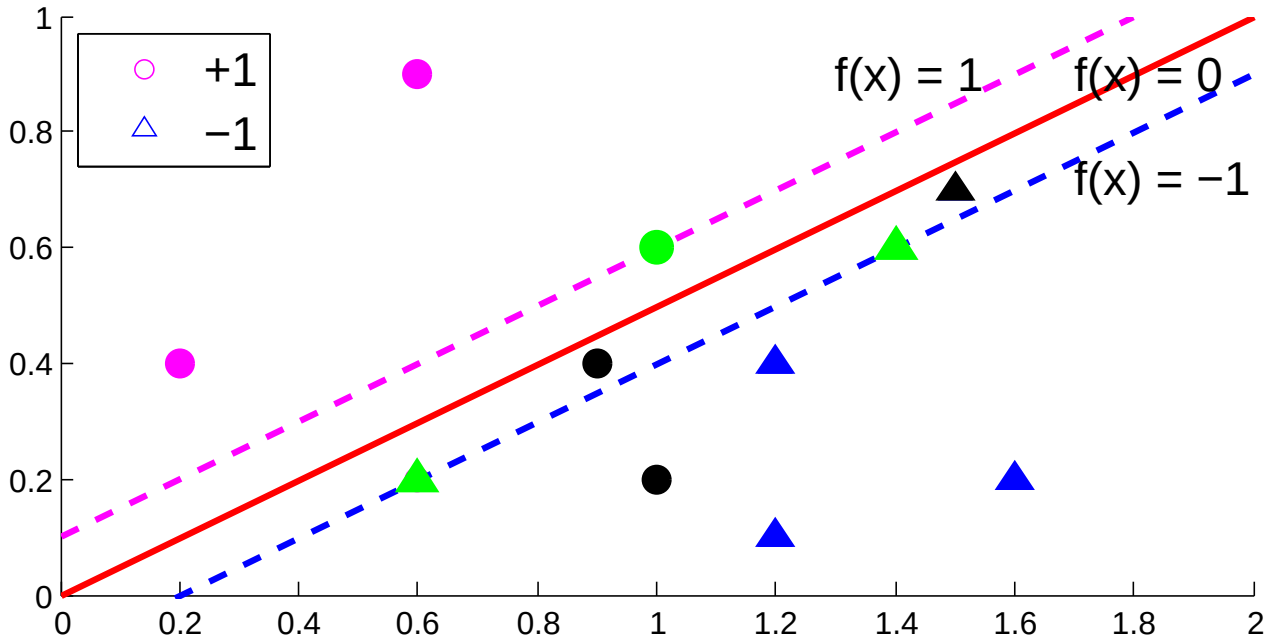
On analyse ℓ par rapport à sa différentiabilité :

- $\mathcal{I}_\alpha = \{i : y_i f(\mathbf{x}_i) = 1\} \Rightarrow 0 \leq \alpha_i \leq 1,\ i \in \mathcal{I}_\alpha$, Points sur la Marge
- $\mathcal{I}_\mathbf{1} = \{i : y_i f(\mathbf{x}_i) < 1\} \Rightarrow \alpha_i = 1,\ i \in \mathcal{I}_1$. **Points Mal Classés**



On analyse ℓ par rapport à sa différentiabilité :

- $\mathcal{I}_\alpha = \{i : y_i f(\mathbf{x}_i) = 1\} \Rightarrow 0 \leq \alpha_i \leq 1,\ i \in \mathcal{I}_\alpha$, Points sur la Marge
- $\mathcal{I}_1 = \{i : y_i f(\mathbf{x}_i) < 1\} \Rightarrow \alpha_i = 1,\ i \in \mathcal{I}_1$. **Points Mal Classés**
- $\mathcal{I}_0 = \{i : y_i f(\mathbf{x}_i) > 1\} \Rightarrow \alpha_i = 0,\ i \in \mathcal{I}_0$, Points Bien Classés



On analyse ℓ par rapport à sa différentiabilité :

- $\mathcal{I}_\alpha = \{i : y_i f(\mathbf{x}_i) = 1\} \Rightarrow 0 \leq \alpha_i \leq 1,\ i \in \mathcal{I}_\alpha$, Points sur la Marge
- $\mathcal{I}_1 = \{i : y_i f(\mathbf{x}_i) < 1\} \Rightarrow \alpha_i = 1,\ i \in \mathcal{I}_1$. **Points Mal Classés**
- $\mathcal{I}_0 = \{i : y_i f(\mathbf{x}_i) > 1\} \Rightarrow \alpha_i = 0,\ i \in \mathcal{I}_0$, Points Bien Classés

Soit $J(f) = \frac{\lambda}{2}\|f\|^2_{\mathcal{H}} + \sum_{i=1}^{n} \max\{0, 1 - y_i f(\mathbf{x}_i)\}$, alors

$$\begin{aligned} J(f) &= \frac{\lambda}{2}\|f\|^2_{\mathcal{H}} + \sum_{i \in \mathcal{I}_\alpha} \max\{0, 1 - y_i f(\mathbf{x}_i)\} \\ &\quad + \sum_{i \in \mathcal{I}_0} \max\{0, 1 - y_i f(\mathbf{x}_i)\} + \sum_{i \in \mathcal{I}_1} \max\{0, 1 - y_i f(\mathbf{x}_i)\ . \end{aligned}$$

Conditions d'optimalité : Discrimination

On peut calculer la sous-différentielle de J en utilisant la règle de la somme [Schi 07] et de la chaîne [Clar 98] :

$$\begin{aligned}
\partial_{f_0} J &= \lambda f_0(\cdot) - \sum_{i \in \mathcal{I}_\alpha} \alpha_i y_i \mathbf{k}(\mathbf{x}_i, \cdot) - \sum_{i \in \mathcal{I}_0} \alpha_i y_i \mathbf{k}(\mathbf{x}_i, \cdot) - \sum_{i \in \mathcal{I}_1} \alpha_i y_i \mathbf{k}(\mathbf{x}_i, \cdot). \\
\partial_b J &= -\sum_{i \in \mathcal{I}_0} \alpha_i y_i - \sum_{i \in \mathcal{I}_\alpha} \alpha_i y_i - \sum_{i \in \mathcal{I}_1} \alpha_i y_i
\end{aligned}$$

Conditions d'optimalité : Discrimination

On peut calculer la sous-différentielle de J en utilisant la règle de la somme [Schi 07] et de la chaîne [Clar 98] :

$$\begin{aligned}
\partial_{f_0} J &= \lambda f_0(\cdot) - \sum_{i\in\mathcal{I}_\alpha} \alpha_i y_i \mathbf{k}(\mathbf{x}_i,\cdot) - \sum_{i\in\mathcal{I}_0} \alpha_i y_i \mathbf{k}(\mathbf{x}_i,\cdot) - \sum_{i\in\mathcal{I}_1} \alpha_i y_i \mathbf{k}(\mathbf{x}_i,\cdot). \\
\partial_b J &= -\sum_{i\in\mathcal{I}_0} \alpha_i y_i - \sum_{i\in\mathcal{I}_\alpha} \alpha_i y_i - \sum_{i\in\mathcal{I}_1} \alpha_i y_i
\end{aligned}$$

A l'optimalité, il existe $\alpha_i, i \in [\![n]\!]$ tel que

$$\begin{aligned}
0 &= \lambda f_0(\cdot) - \sum_{i\in\mathcal{I}_\alpha} \alpha_i y_i \mathbf{k}(\mathbf{x}_i,\cdot) - \sum_{i\in\mathcal{I}_0} \alpha_i y_i \mathbf{k}(\mathbf{x}_i,\cdot) - \sum_{i\in\mathcal{I}_1} \alpha_i y_i \mathbf{k}(\mathbf{x}_i,\cdot). \\
0 &= -\sum_{i\in\mathcal{I}_0} \alpha_i y_i - \sum_{i\in\mathcal{I}_\alpha} \alpha_i y_i - \sum_{i\in\mathcal{I}_1} \alpha_i y_i
\end{aligned}$$

Théorème de représentation [Kime 71]

La fonction de décision appartient à un **RKHS** de noyau $k(\cdot,\cdot)$.

$$f(\mathbf{x}) = \frac{1}{\lambda}\sum_{i=1}^{n} \alpha_i y_i k(\mathbf{x}, \mathbf{x}_i) + b$$

A l'optimalité, il existe $\alpha_i, i \in [\![n]\!]$ tel que

$$0 = \lambda f_0(\cdot) - \sum_{i\in\mathcal{I}_\alpha} \alpha_i y_i \mathbf{k}(\mathbf{x}_i,\cdot) - \sum_{i\in\mathcal{I}_0} \alpha_i y_i \mathbf{k}(\mathbf{x}_i,\cdot) - \sum_{i\in\mathcal{I}_1} \alpha_i y_i \mathbf{k}(\mathbf{x}_i,\cdot).$$

$$0 = -\sum_{i\in\mathcal{I}_0} \alpha_i y_i - \sum_{i\in\mathcal{I}_\alpha} \alpha_i y_i - \sum_{i\in\mathcal{I}_1} \alpha_i y_i$$

Théorème de représentation [Kime 71]

La fonction de décision appartient à un **RKHS** de noyau $k(\cdot,\cdot)$.

$$f(\mathbf{x}) = \frac{1}{\lambda}\sum_{i=1}^{n} \alpha_i y_i k(\mathbf{x}, \mathbf{x}_i) + b$$

Théorème (Equivalence des solutions)

Les sous-gradients $\{\alpha_i^*\}_{i=[\![n]\!]}$ utilisés comme coefficients d'une solution optimale ${f_0}^*$ sont une solution du problème SVM si chaque sous-gradient est vu comme un multiplicateur de Lagrange de $\mathbf{x}_i$ et vice versa.

Résolution du SVM

Problème dual

Un problème dual sous forme de problème paramétrique quadratique (QP) peut être dérivé. Pour le résoudre, il existe :

- SMO [Plat 99]
- Méthodes de contraintes actives [Loos 07]
- Méthodes de points intérieurs

Problème Primal/Bidual

Un problème bidual analogue au primal peut être aussi dérivé.

- Méthodes pour le primal/bidual [Chap 07, Shal 07, Do 08, Bott 08]

Sélection de modèle

Pour la sélection de modèle, un problème doit être résolu pour chaque valeur du paramètre de régularisation.

Les solutions du SVM ont une structure particulière qui peut être exploitée pour la recherche du modèle [Mark 52, Osbo 99, Hast 04, Ross 07].

A l'optimalité, les conditions suivantes doivent être satisfaites :

$$1 = y_j f(\mathbf{x}_j) \quad = \quad \frac{1}{\lambda}\left(\sum_{i \in \mathcal{I}_\alpha \cup \mathcal{I}_1} \alpha_i y_i y_j \mathbf{k}(\mathbf{x}_i, \mathbf{x}_j) + \alpha_0 y_j\right) \quad \forall\, \mathbf{x}_j \in \mathcal{I}_\alpha$$

$$0 = \sum_{i=1}^{n} \alpha_i y_i$$

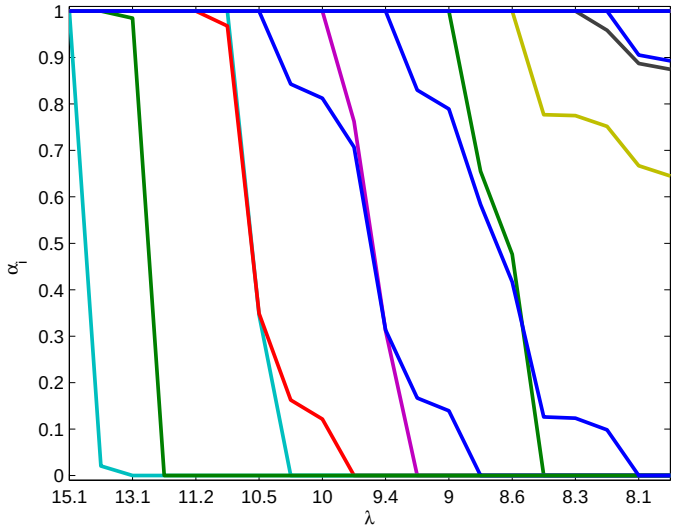
dont on déduit la solution :

$$\begin{bmatrix} \alpha_0 \\ \boldsymbol{\alpha}_{\mathcal{I}_\alpha} \end{bmatrix} = \begin{bmatrix} 0 & \mathbf{y}_{\mathcal{I}_\alpha}^\top \\ \mathbf{y}_{I_\alpha} & Y_{\mathcal{I}_\alpha} K_{\mathcal{I}_\alpha, \mathcal{I}_\alpha} Y_{\mathcal{I}_\alpha} \end{bmatrix}^{-1} \begin{bmatrix} -\mathbf{y}_{\mathcal{I}_1}^\top \mathbb{1}_{\mathcal{I}_1} \\ \lambda \mathbb{1}_{I_\alpha} - Y_{\mathcal{I}_1} K_{\mathcal{I}_1, \mathcal{I}_\alpha} \mathbf{y}_{\mathcal{I}_\alpha} \end{bmatrix}$$

avec $\alpha_0 = \lambda b$.

En dérivant par rapport à λ, on obtient une relation linéaire :

$$\begin{bmatrix} \frac{\partial \alpha_0}{\partial \lambda} \\ \frac{\partial \boldsymbol{\alpha}_{\mathcal{I}_\alpha}}{\partial \lambda} \end{bmatrix} = \begin{bmatrix} 0 & \mathbf{y}_{\mathcal{I}_\alpha}^\top \\ \mathbf{y}_{I_\alpha} & Y_{\mathcal{I}_\alpha} K_{\mathcal{I}_\alpha, \mathcal{I}_\alpha} Y_{\mathcal{I}_\alpha} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \mathbb{1}_{I_\alpha} \end{bmatrix}.$$



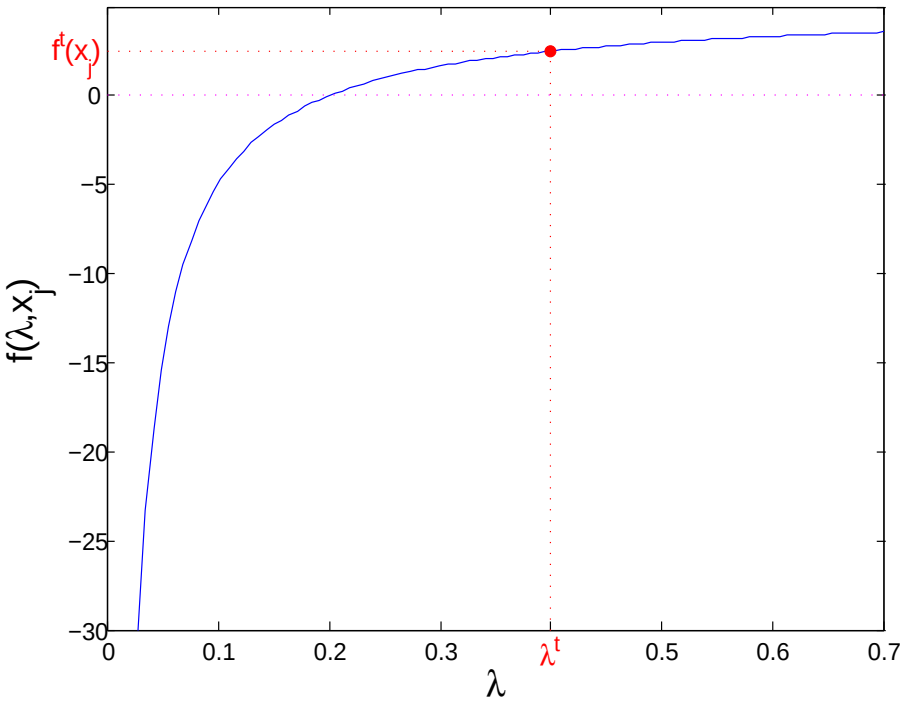
Valide tant que les ensembles $\mathcal{I}_\alpha, \mathcal{I}_0$ et $\mathcal{I}_1$ ne changent pas (pas d'événement).

Avec la **détection des événements** toutes les solutions $\{\alpha_i\}_{i=0,\ldots,n}$ pour toutes les valeurs de λ **peuvent être obtenues**.

Quel λ choisir ?

Courbes d'erreur de validation

Question : Quel modèle (λ) choisir : erreur de validation.



Comment suivre l'**évolution** de l'erreur de validation ?

Valeur de $\mathbf{x}_j$, $y_j > 0$ pour λ^t :

$$f^t(\mathbf{x}_j) > 0$$

Point bien classé.

Forme de la fonction de décision

$f(\mathbf{x}_j)$ dépend hyperboliquement de λ : $f(\mathbf{x}_j) = \frac{1}{\lambda}\tau_j^t + \upsilon_j^t$.

τ_j^t et υ_j^t dépendent de $\{\alpha_i\}_{i=0,\ldots,n}$ et restent constants si $\mathcal{I}_\alpha, \mathcal{I}_0$ et $\mathcal{I}_1$ sont fixes.

Courbes d'erreur de validation

Question : Quel modèle (λ) choisir : erreur de validation.

Comment suivre l'**évolution** de l'erreur de validation ?

Changement de $f(\mathbf{x}_j)$ par rapport à λ.

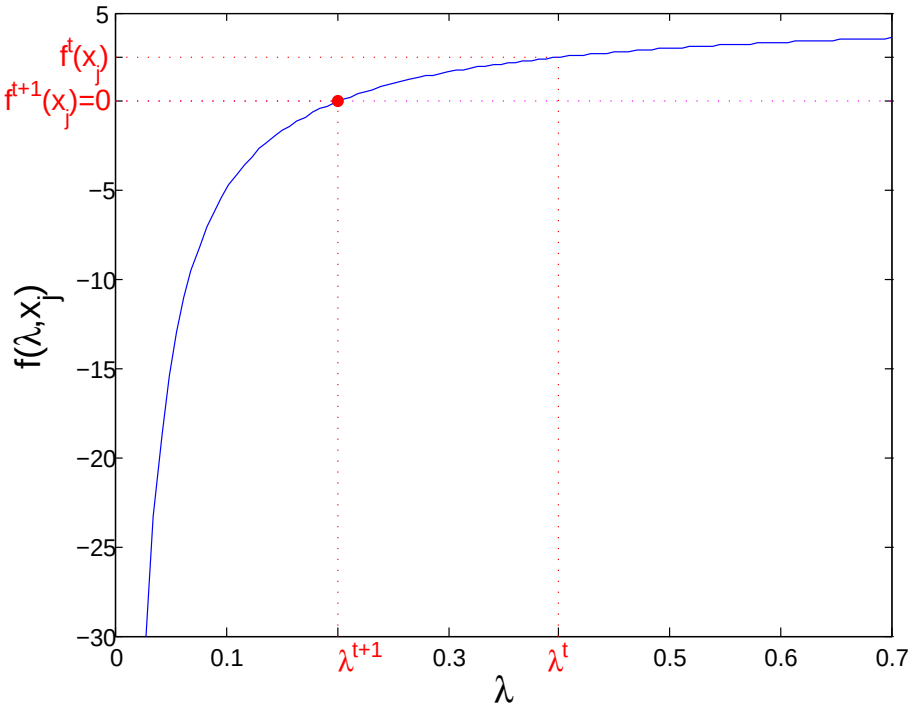
Forme de la fonction de décision

$f(\mathbf{x}_j)$ dépend hyperboliquement de λ : $f(\mathbf{x}_j) = \frac{1}{\lambda}\tau_j^t + \upsilon_j^t$.

τ_j^t et υ_j^t dépendent de $\{\alpha_i\}_{i=0,\ldots,n}$ et restent constants si $\mathcal{I}_\alpha, \mathcal{I}_0$ et $\mathcal{I}_1$ sont fixes.

Courbes d'erreur de validation

Question : Quel modèle (λ) choisir : erreur de validation.



Comment suivre l'**évolution** de l'erreur de validation ?

Événement en λ^{t+1} !

$$f^{t+1}(\mathbf{x}_j) = 0.$$

Changement de l'erreur de validation.

Point mal classé

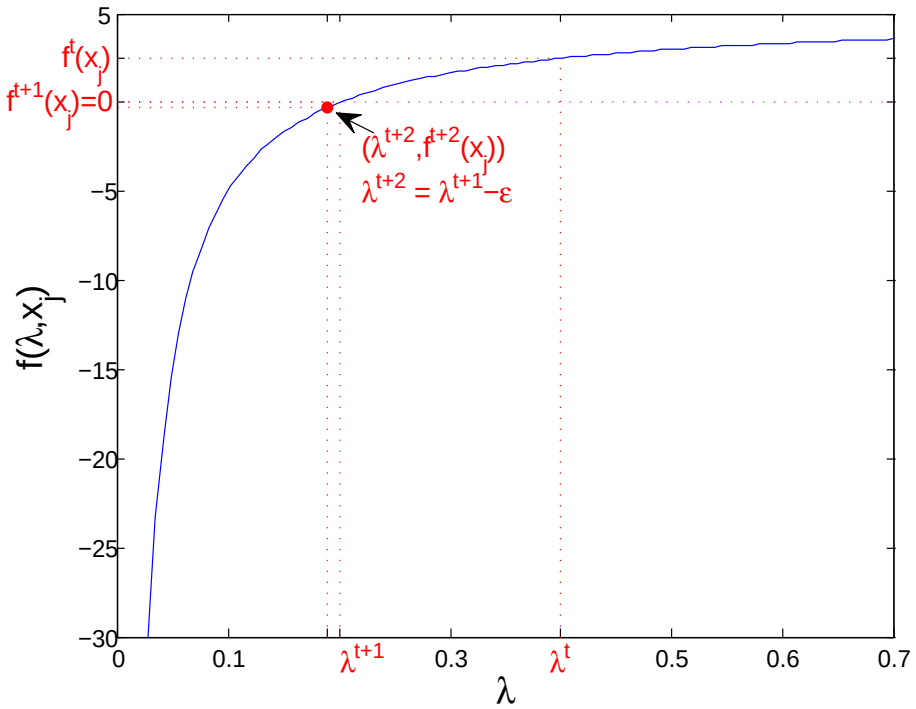
Forme de la fonction de décision

$f(\mathbf{x}_j)$ dépend hyperboliquement de λ : $f(\mathbf{x}_j) = \frac{1}{\lambda}\tau_j^t + \upsilon_j^t$.

τ_j^t et υ_j^t dépendent de $\{\alpha_i\}_{i=0,\ldots,n}$ et restent constants si $\mathcal{I}_\alpha, \mathcal{I}_0$ et $\mathcal{I}_1$ sont fixes.

Courbes d'erreur de validation

Question : Quel modèle (λ) choisir : erreur de validation.



Comment suivre l'**évolution** de l'erreur de validation ?

Pour $\lambda < \lambda^{t+1}$

$f(\mathbf{x}_j) < 0.$

Pas de changement de l'erreur de validation

Forme de la fonction de décision

$f(\mathbf{x}_j)$ dépend hyperboliquement de λ : $f(\mathbf{x}_j) = \frac{1}{\lambda}\tau_j^t + \upsilon_j^t$.

τ_j^t et υ_j^t dépendent de $\{\alpha_i\}_{i=0,\ldots,n}$ et restent constants si $\mathcal{I}_\alpha, \mathcal{I}_0$ et $\mathcal{I}_1$ sont fixes.

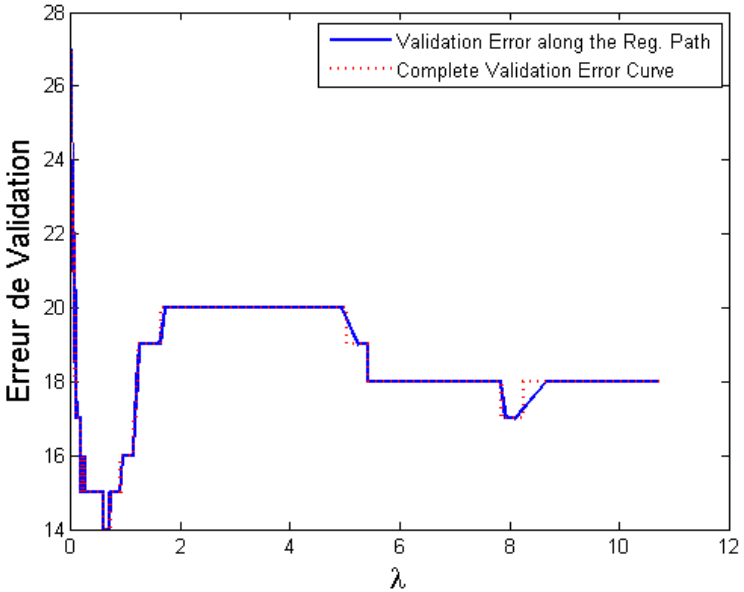
La courbe complète de validation peut être construite avec cette relation.

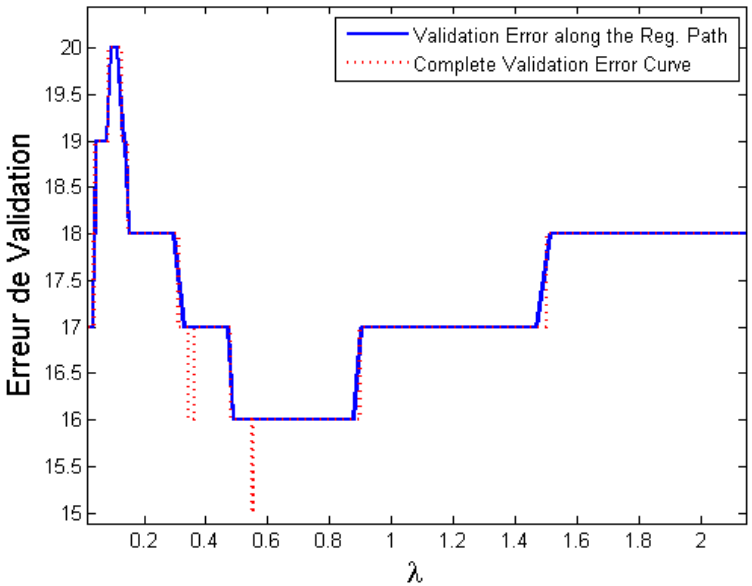
Partition pour λ

Le chemin de régularisation propose une partition pour les valeurs admissibles de λ.

(Apprentissage-Validation) : valeurs d'intérêt de λ

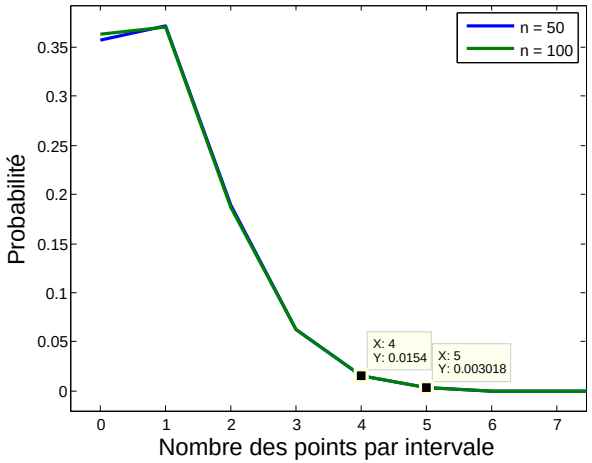
Des valeurs intéressantes peuveut se trouver en dehors du chemin de régularisation (échantillonage non-homogène, grande dimension).

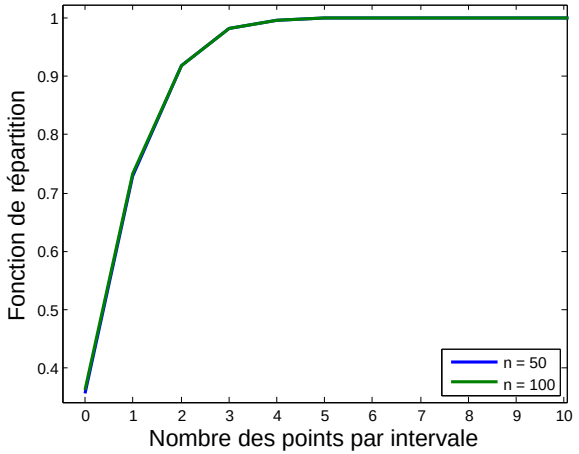




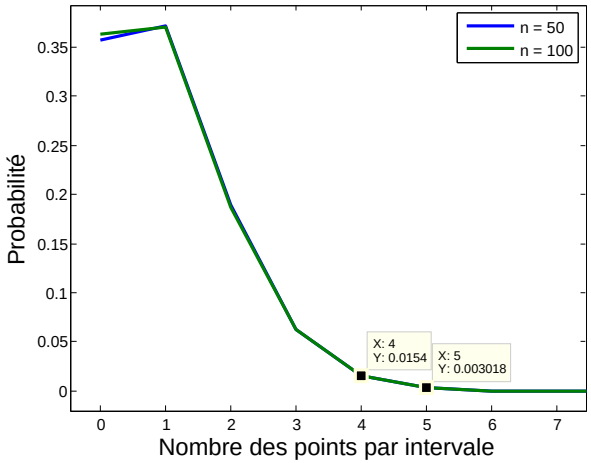
Courbes d'erreur de validation : Expériences

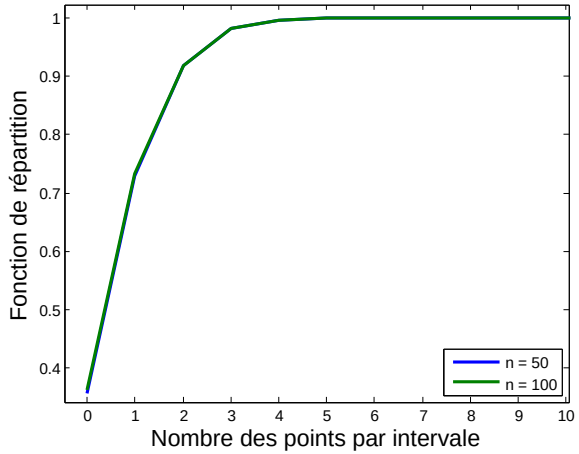
Le nombre **théorique** de points de validation entre chaque point d'interruption du chemin de régularisation :



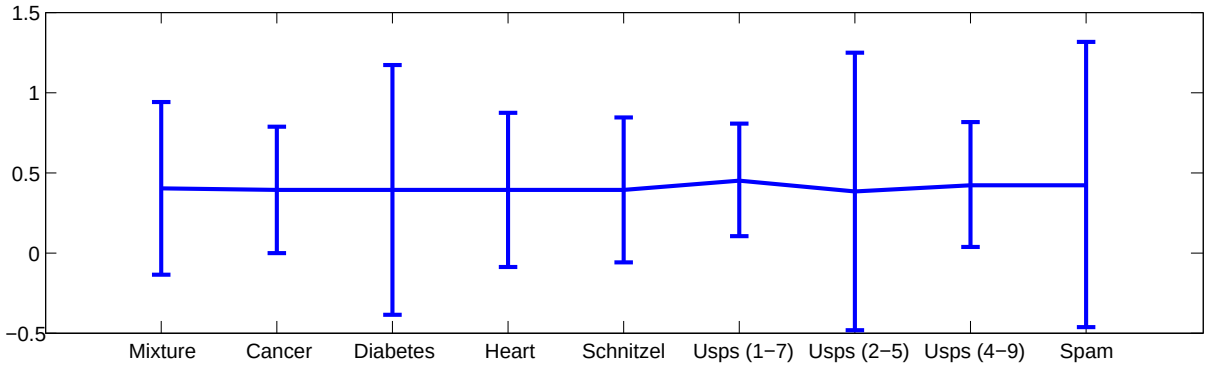


Le nombre **théorique** de points de validation entre chaque point d'interruption du chemin de régularisation :





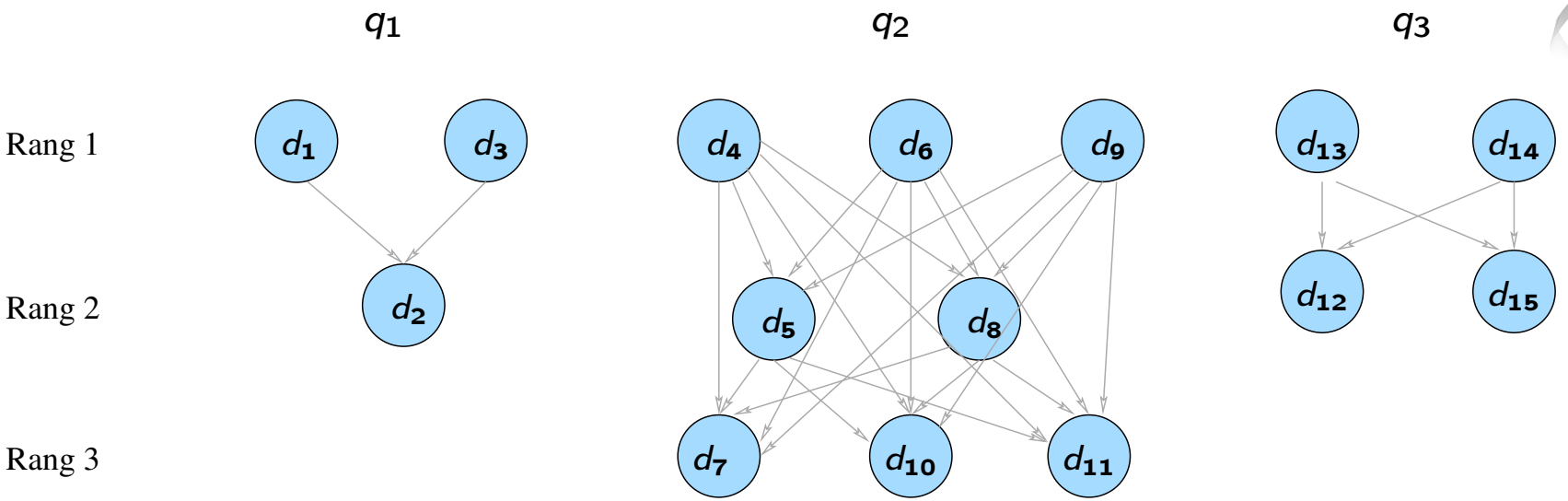
Le nombre **expérimental** de points de validation entre chaque point d'interruption du chemin de régularisation :



Courbes d'erreur de validation : Récapitulatif

- La **sous-différentielle** est une technique appropriée pour résoudre un **problème multi-critères non-différentiable**.
- La **solution** donnée par la **sous-différentielle** est **équivalente** à celle donnée par une approche avec le **Lagrangien**.
- Le **chemin de régularisation** donne un **bon échantillon des valeurs de** λ parmi lequel on choisit un modèle.
- Par rapport aux événements d'apprentissage, on peut espérer un comportement similaire en validation (très probablement moins de 3 points par intervalle).

Feuille de route



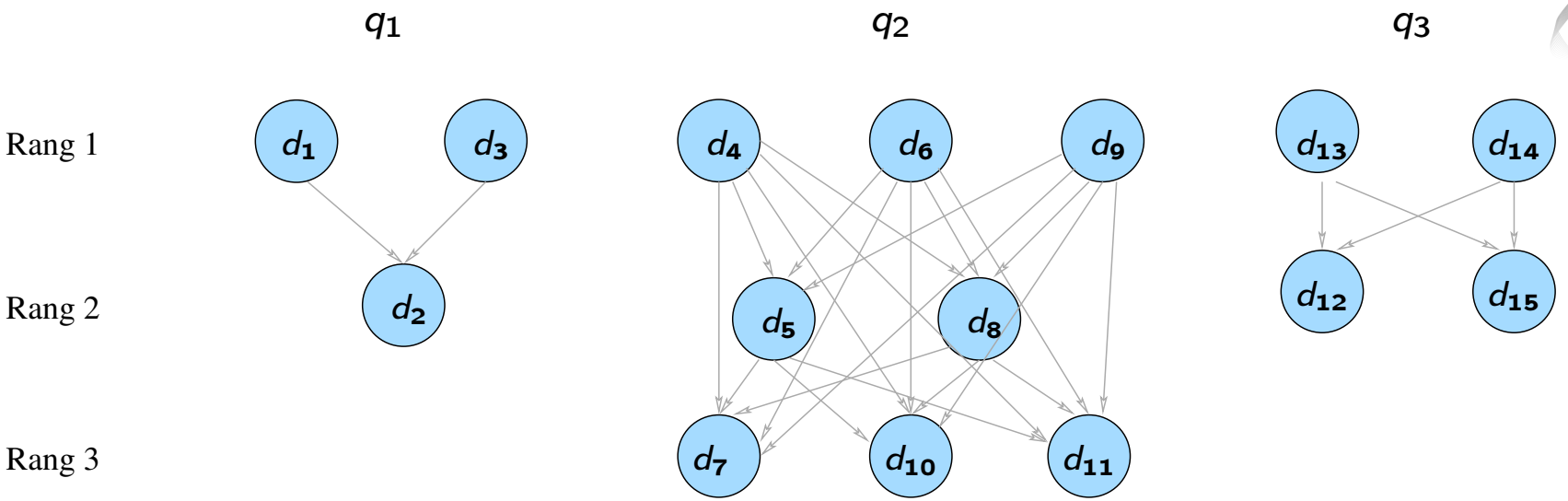
Requêtes de la forme (q, d, y), avec :

- q - requête
- d - document
- y - rang du document d pour la requête q.

Soit $\mathbf{x} = (q, d)$. On cherche f tel que pour une paire $(\mathbf{x}_i, \mathbf{x}_j)$

$$f(\mathbf{x}_i) > f(\mathbf{x}_j) \quad \text{si} \quad y_i > y_j$$

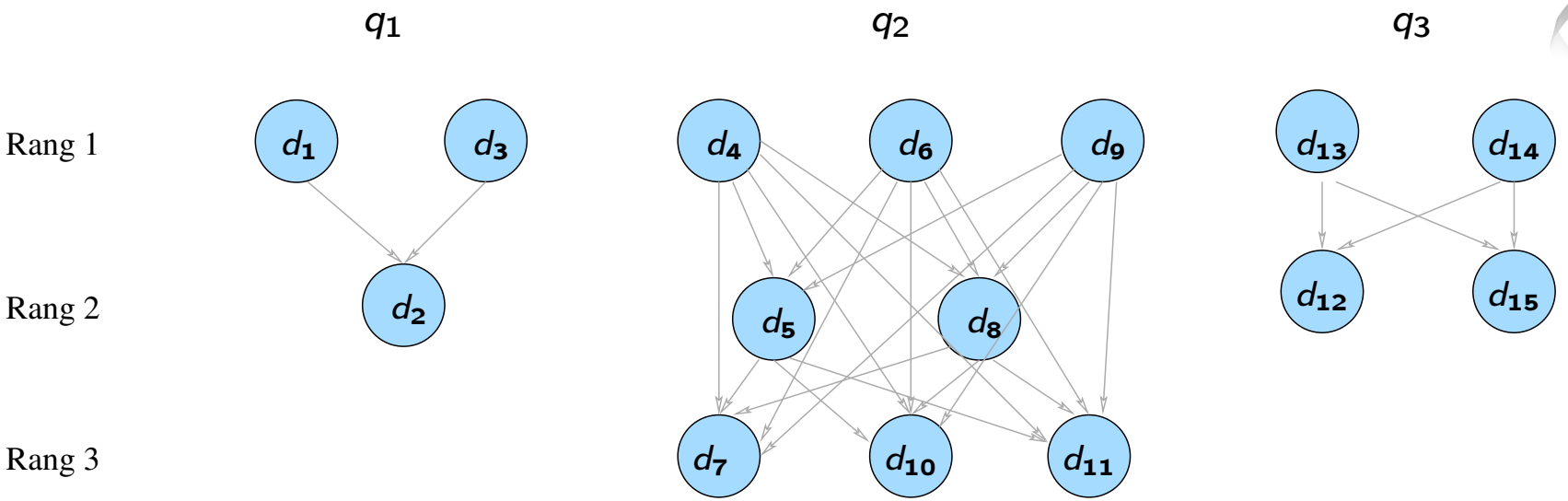
avec $\mathbf{x}_i = (q_u, d_u), \mathbf{x}_j = (q_v, d_v)$



Le problème primal d'ordonnancement est défini comme suit :

$$\begin{aligned} \min_{f,\boldsymbol{\xi}} \quad & \sum_{i=1}^{m} \xi_i + \frac{\lambda}{2} \|f\|_{\mathcal{H}}^2 \\ \text{s. c.} \quad & f(\mathbf{x}_{u_i}) - f(\mathbf{x}_{v_i}) \geq 1 - \xi_i \quad \forall i = [\![m]\!] \\ & \xi_i \geq 0 \quad \forall i = [\![m]\!] \end{aligned}$$

m le nombre de contraintes. Ici, $m \in O(n^2)$.



Un problème dual de forme quadratique peut être dérivé :

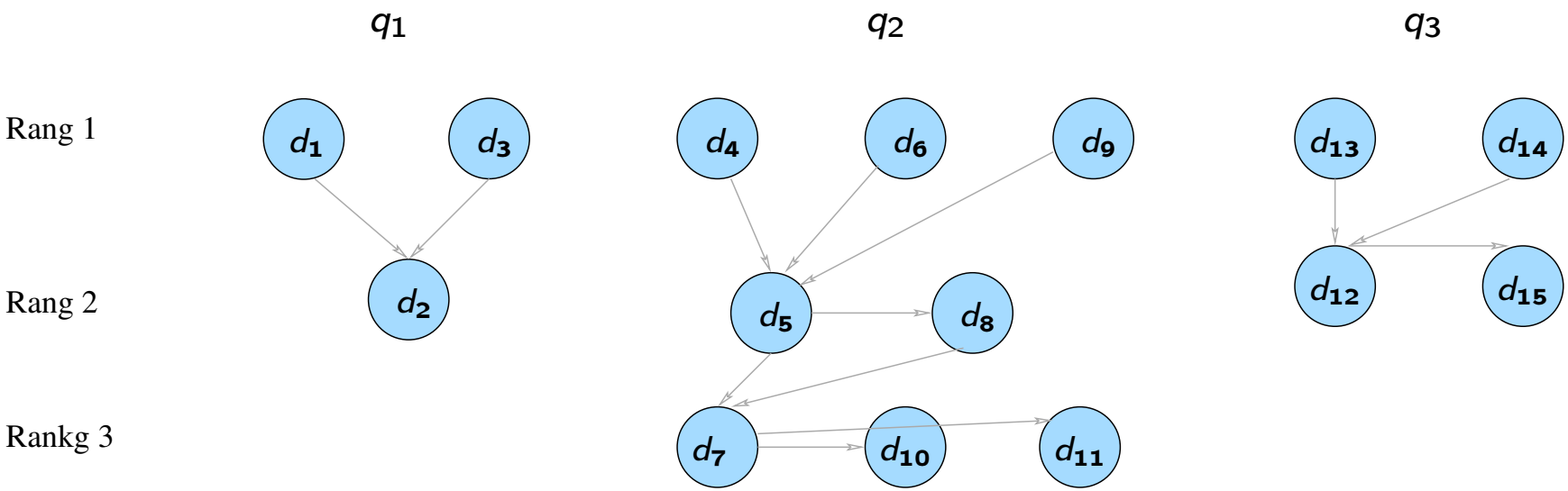
$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & \boldsymbol{\alpha}^\top \mathbb{1} - \frac{1}{2\lambda} \boldsymbol{\alpha}^\top P K P^\top \boldsymbol{\alpha} \\ \text{s. c.} \quad & \mathbf{0} \leq \boldsymbol{\alpha} \leq \mathbb{1}. \end{aligned}$$

On retrouve la forme de la fonction de décision : $f(\mathbf{x}) = \frac{1}{\lambda} \boldsymbol{\alpha}^\top P \mathbf{k}(\mathbf{x})$
Le problème est de taille m qui est généralement $O(n^2)$.

avec $\boldsymbol{\alpha} \in \mathbb{R}^m$, $P \in \mathbb{R}^{m \times n}$, matrice de contraintes, K la matrice de Gram, $K_{ij} = \mathbf{k}(\mathbf{x}_i, \mathbf{x}_j)$, $\forall i, j = [\![n]\!]$ et $\mathbf{k}(\mathbf{x}) = \big(\mathbf{k}(\mathbf{x}, \mathbf{x}_1), \mathbf{k}(\mathbf{x}, \mathbf{x}_2), \ldots, \mathbf{k}(\mathbf{x}, \mathbf{x}_n)\big)^\top$.

Ordonnancement : Singularité de RankSVM

- La matrice P induit une singularité dans le problème dual
- Des méthodes efficaces pour les SVM linéaires ont été développées [Joac 06, Chap 09].
- Nous proposons d'utiliser un graphe réduit [Zapi 09] pour briser la complexité.
- Celle-ci sera de l'ordre $O(n)$



Input : set $\{(\mathbf{x}_i, y_i, q_i)\}$, $i = [\![n]\!]$.
Output : matrice réduite adjacente $P' \in \mathbb{R}^{m' \times n}$.
Set n_r = nombre de rangs, n_q = nombre des requêtes, and $j = 1$.
for $Q = 1$ to n_q **do**
  for $Q = n_r - 1$ to 1 **do**
    Choix aléatoire de k, $\mathbf{x}_k$, avec rang $y_k = R$ et $q_k = Q$, $n_s = |V|$.
    Construire contraintes intra-rang
    Construire contraintes inter-rang
  end for
end for

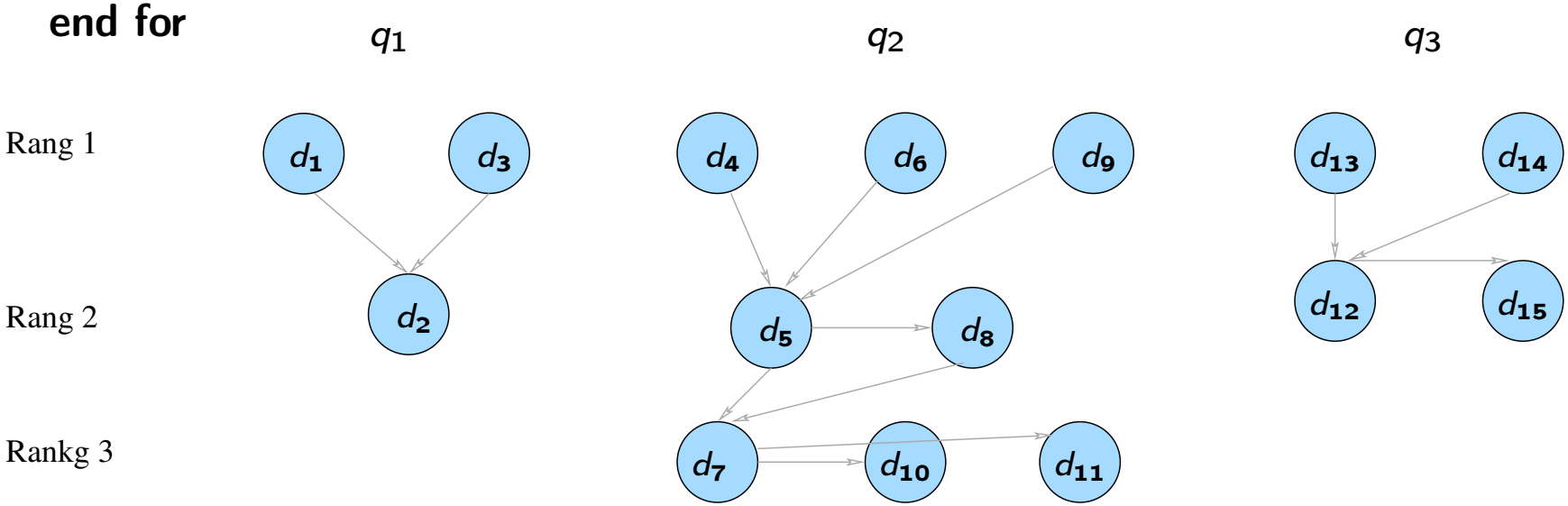
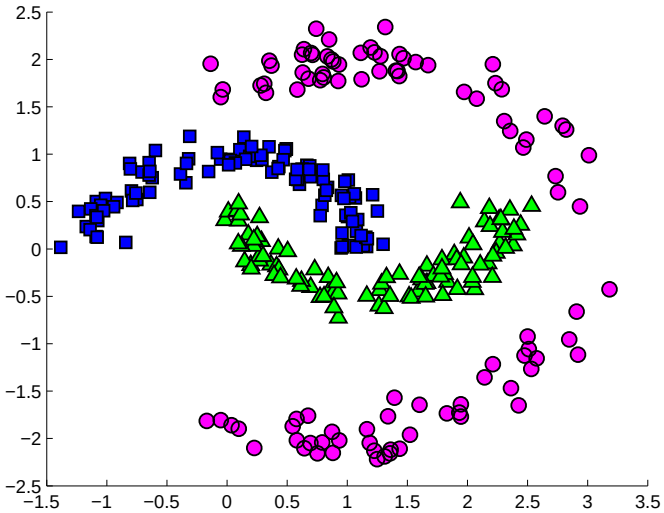


Illustration des résultats avec les exemples « 3moons »et « 3mixtures »

- Les exemples sont divisés en **trois rangs**.
- Cercles magentas : exemples plus pertinents
- Carrés bleus : exemples moyennement pertinents
- Triangles verts : exemples moins pertinents

La fonction de décision donne les valeurs les plus élevées aux exemples avec des rangs plus pertinents.



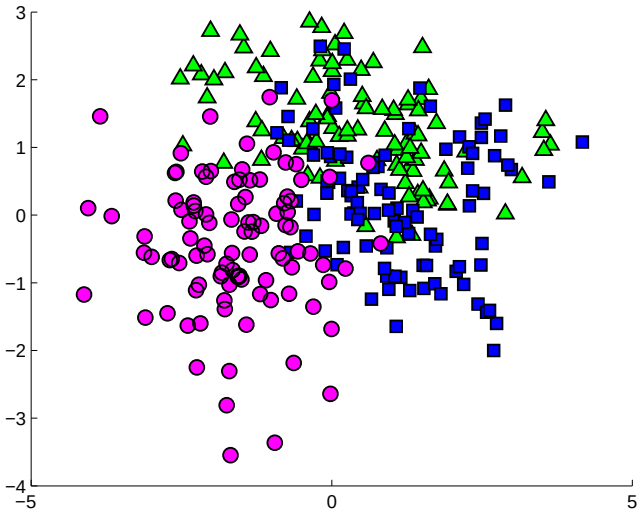
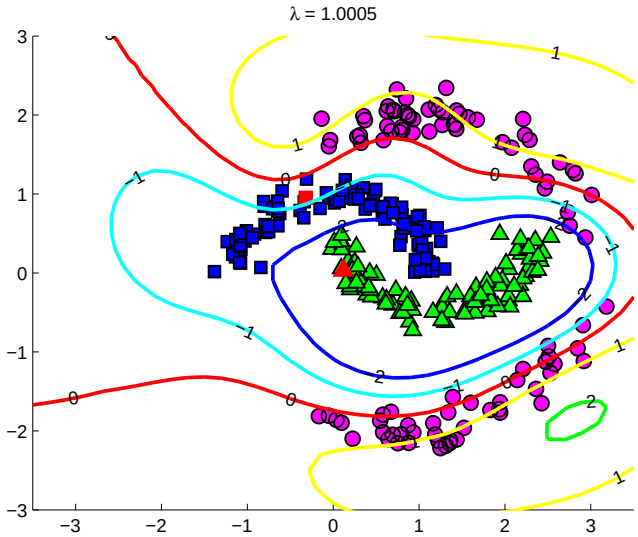
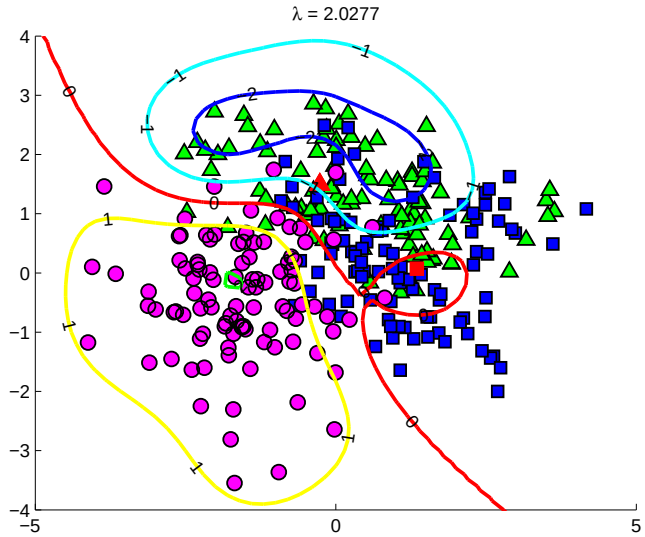


Illustration des résultats avec les exemples « 3moons »et « 3mixtures »

- Les exemples sont divisés en **trois rangs**.
- Cercles magentas : exemples plus pertinents
- Carrés bleus : exemples moyennement pertinents
- Triangles verts : exemples moins pertinents

La fonction de décision donne les valeurs les plus élevées aux exemples avec des rangs plus pertinents.



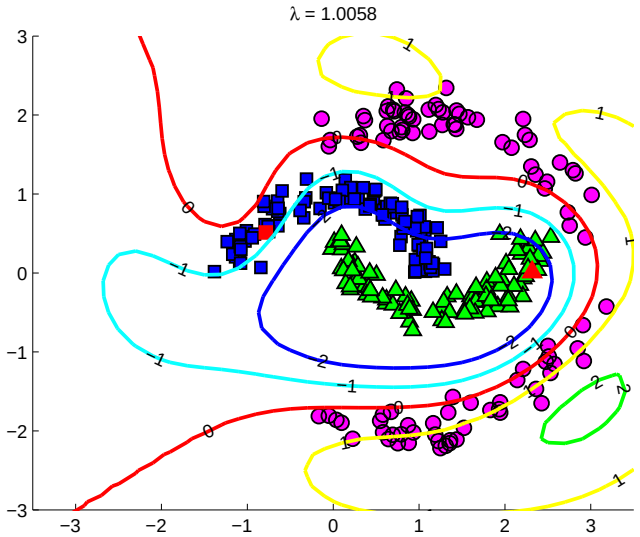


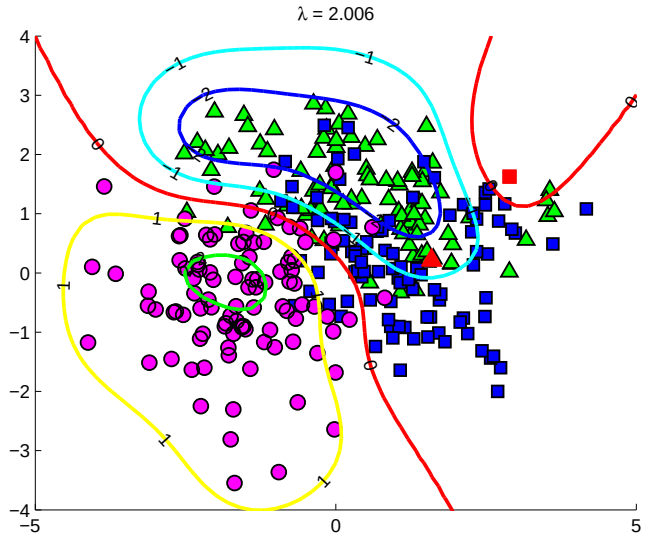
Ordonnancement : Utilisation du Graphe Réduit

Illustration des résultats avec les exemples « 3moons »et « 3mixtures »

- Les exemples sont divisés en **trois rangs**.
- Cercles magentas : exemples plus pertinents
- Carrés bleus : exemples moyennement pertinents
- Triangles verts : exemples moins pertinents

La fonction de décision donne les valeurs les plus élevées aux exemples avec des rangs plus pertinents.



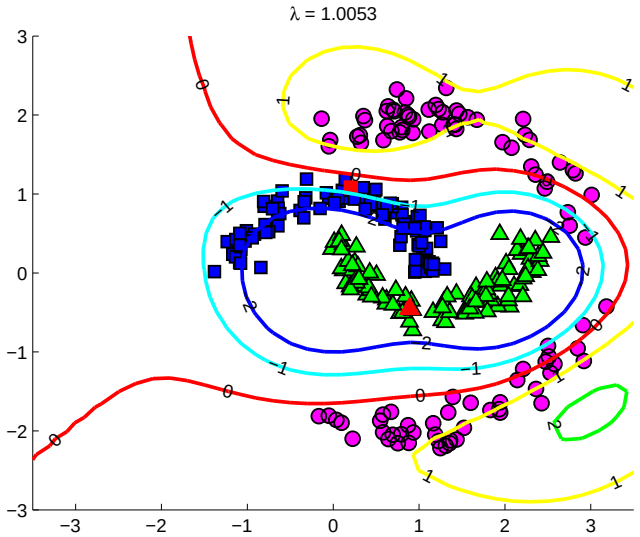


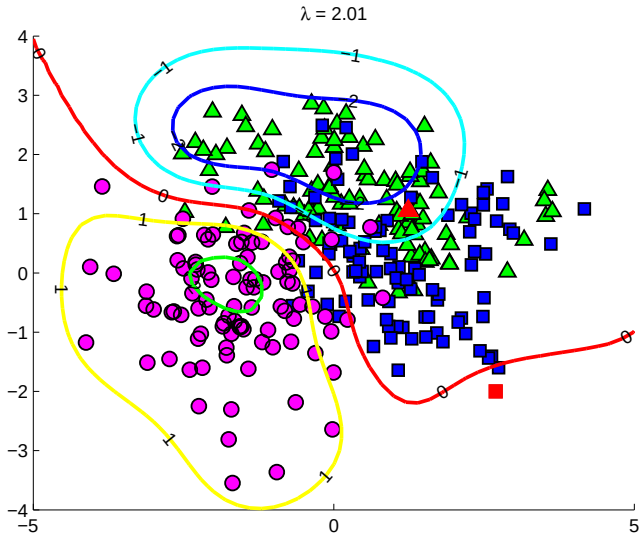
Ordonnancement : Utilisation du Graphe Réduit

Illustration des résultats avec les exemples « 3moons »et « 3mixtures »

- Les exemples sont divisés en **trois rangs**.
- Cercles magentas : exemples plus pertinents
- Carrés bleus : exemples moyennement pertinents
- Triangles verts : exemples moins pertinents

La fonction de décision donne les valeurs les plus élevées aux exemples avec des rangs plus pertinents.





Ohsumed : Base de données de documents médicaux classés pour chaque requête par des experts.

Ohsumed	Taille P	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Temps
P **C**omplète	349553	**0.38**	**0.44**	**0.40**	**0.48**	**0.42**	105s
P **R**eduite	**11108**	0.32	**0.43**	0.36	0.43	**0.42**	**0**.42s
Rapport	0.03	0.84	0.98	0.9	0.9	1	250

Tab.: Taille moyenne de P et P réduit et ndcg@10 pour chaque sous-ensemble « Fold » d'ohsumed avec un noyau linéaire.

- $\mathcal{I}_\alpha = \{i : f(\mathbf{x}_{k_i}) - f(\mathbf{x}_{l_i}) = 1, \quad 0 \leq \boldsymbol{\alpha}_i \leq 1\}$, Paires sur la Marge
- $\mathcal{I}_1 = \{i : f(\mathbf{x}_{k_i}) - f(\mathbf{x}_{l_i}) < 1, \quad \boldsymbol{\alpha}_i = 1\}$, **Paires Mal Classées**
- $\mathcal{I}_0 = \{i : f(\mathbf{x}_{k_i}) - f(\mathbf{x}_{l_i}) > 1, \quad \boldsymbol{\alpha}_i = 0\}$, Paires Bien Classées

$$
\begin{aligned}
f(\mathbf{x}) &= \frac{1}{\lambda}\boldsymbol{\alpha}^\top P\mathbf{k}(\mathbf{x}) = \frac{1}{\lambda}\left(\boldsymbol{\alpha}_{\mathcal{I}_\alpha}^\top P_{\mathcal{I}_\alpha}.\mathbf{k}(\mathbf{x}) + \boldsymbol{\alpha}_{\mathcal{I}_1}^\top P_{\mathcal{I}_1}.\mathbf{k}(\mathbf{x})\right) + \boldsymbol{\alpha}_{\mathcal{I}_0}^\top P_{\mathcal{I}_0}.\mathbf{k}(\mathbf{x}) \\
&= \frac{1}{\lambda}\left(\boldsymbol{\alpha}_{\mathcal{I}_\alpha}^\top P_{\mathcal{I}_\alpha}.\mathbf{k}(\mathbf{x}) + \mathbb{1}_{\mathcal{I}_1}^\top P_{\mathcal{I}_1}.\mathbf{k}(\mathbf{x})\right)
\end{aligned}
$$

Si $\mathcal{I}_0$, $\mathcal{I}_1$ et $\mathcal{I}_\alpha$ ne changent pas pour $\lambda^{t+1} \leq \lambda \leq \lambda^t$, on peut montrer :

$$\boldsymbol{\alpha}_i = \boldsymbol{\alpha}_i^t - (\lambda^t - \lambda)\eta_i \qquad i \in \mathcal{I}_\alpha.$$

$\boldsymbol{\alpha}_i \ \forall \ i \in \mathcal{I}_\alpha$ change de manière linéaire par morceaux par rapport à λ.

avec $\boldsymbol{\eta} = (P_{\mathcal{I}_\alpha} K P_{\mathcal{I}_\alpha}^\top)^{-1}\mathbb{1}_{\mathcal{I}_\alpha}$ et $P_{\mathcal{I}}$ la sous-matrice avec les $\mathcal{I}$ lignes et toutes les colonnes de P (complète ou reduite).

Ordonnancement : Chemin de Régularisation

Linéarité par morceaux $\Longrightarrow$ Chemin de régularisation [Zapi 08b].

$$\boldsymbol{\alpha}_i = \boldsymbol{\alpha}_i^t - (\lambda^t - \lambda)\eta_i \qquad i \in \mathcal{I}_\alpha.$$

Événement : quelle valeur de λ^{t+1} provoque un changement de I_α, I_0 ou I_1 ?

Détection des événements

- Une paire i en $\mathcal{I}_\alpha$ passe à $\mathcal{I}_1$ ou $\mathcal{I}_0$: α_i devient 0 or 1.
- Une paire i en $\mathcal{I}_1$ or $\mathcal{I}_0$ passe à $\mathcal{I}_\alpha$: $f(\mathbf{x}_{k_i}) - f(\mathbf{x}_{l_i}) = 1$.

Initialisation

- Choisir λ grand : toutes les paires appartiennent à $\mathcal{I}_1$.
- Décroître λ pour avoir au moins une paire en $\mathcal{I}_\alpha$.

Le reste du chemin peut être déduit par la **détection des événements** comme λ décroît et **la mise à jour efficace** de η_i.

Sélection de modèle pour le RankSVM avec le graphe réduit en utilisant le chemin de régularisation [Zapi 08a].

Dataset	Bootstrap		Final Train		Test		Test		Param	
	Grid	Path	CA	Path	Grid	Path	Grid	Path	Grid	Path
	Time (sec)		Time (sec)		Error		ndcg@10		Std dev λ	
mixture	19	**14**	**0.1**	4	15%	**13%**	0.96	**0.98**	7.5	**0.4**
3mixture	29	**23**	**0.2**	19	11%	**9%**	**1**	**1**	7	**0.7**
3moons	31	**20**	**0.1**	12	1%	**0%**	**1**	**1**	7.5	**1.2**
cancer	18	**10**	**0.2**	22	**2%**	**2%**	0.97	**1**	**0.5**	112
diabetes	17	**11**	**1.5**	108	22%	**18%**	0.81	**0.9**	12	**0.6**
auto	26	**17**	**0.3**	19	6%	**5%**	0.95	**1**	9.2	**6.1**
housing	27	**18**	**0.8**	56	16%	**8%**	0.97	**0.98**	11	**0.2**
spam	48	**12**	**92**	9770	18%	**6%**	**1**	**1**	0.3	**0.1**
ad	44	**39**	**11**	2183	7%	**3%**	0.93	**0.94**	1.4	**1**

Tab.: Comparaison de la recherche sur une grille « Grid » vs. Chemin de Regularisation « Path ». L'apprentissage du modèle final avec toutes les données est noté « Final Train ». « CA » dénote la méthode de contraintes actives.

Ordonnancement : Récapitulatif

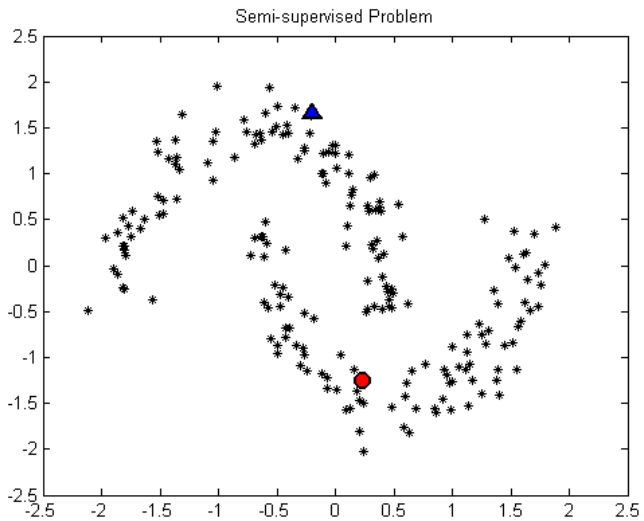
- Le **graphe réduit** aide à **briser la complexité** des problèmes d'ordonnancement en gardant une **performance équivalente**.
- Le chemin de régularisation **réduit le temps de recherche** du modèle et **augmente la performance**.

Feuille de route

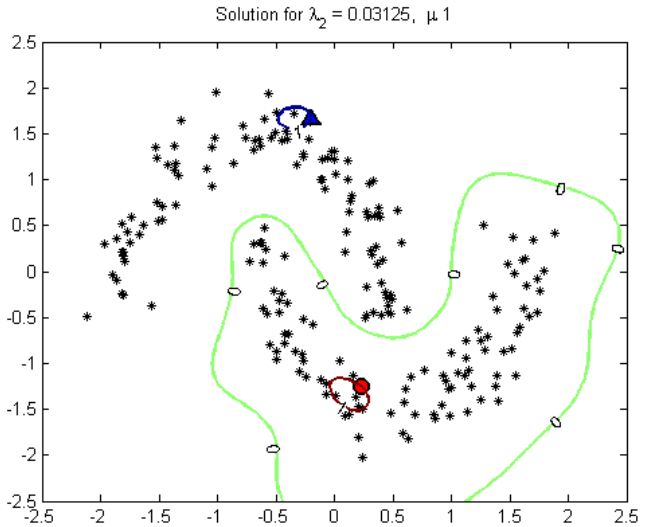
Cadre général

- $S_{\mathcal{L}} = \{(x_i, y_i), \quad i = 1, \ldots, \ell\}$ points étiquetés
- $S_{\mathcal{U}} = \{x_i, \quad i = \ell + 1, \ldots, \ell + u\}$ points sans étiquettes

Objectif : Déterminer la fonction de décision f en se basant sur les points étiquetés et non-étiquetés.



(e) Un point étiqueté par classe



(f) Solution du problème

Contexte

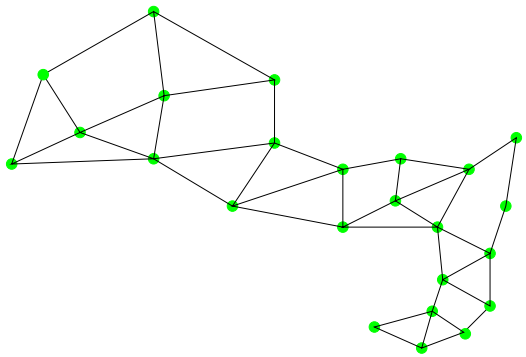
- **Hypothèse** : les données résident sur une **sous-variété régulière** de dimension réduite par rapport à la dimension intrinsèque

Laplacien SVM [Belk 06]

- Généralisation au cadre des SVM

$$\begin{aligned} \min_{f} \quad & \sum_{i=1}^{l} \xi_i + \lambda \|f\|_{\mathcal{H}}^2 + \mu \|f\|_{\mathcal{M}}^2 \\ \text{s. c.} \quad & y_i f(x_i) \geq 1 - \xi_i \quad i = 1, ..., \ell \\ & \xi_i \geq 0 \quad i = 1, ..., \ell \end{aligned}$$

- Pénalisation supplémentaire $\mu \|f\|_{\mathcal{M}}^2$: **éviter que la frontière de décision** donnée par f **passe à travers les variétés**



Graphe de similarité

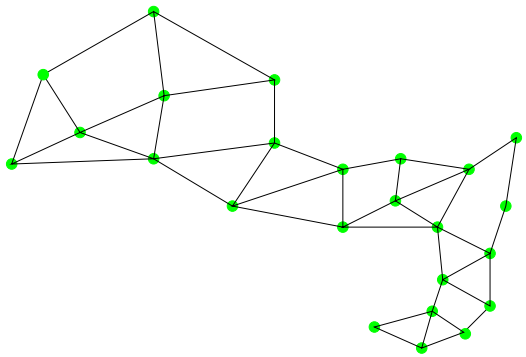
- **Graphe** : $G(V, E)$, V : points $\{x_i, i = [\![n]\!]\}$, $n = \ell + u$ et E : arcs.
- **Matrice d'adjacence** : $W \in \mathbb{R}^{n \times n}$:

$$W_{ij} = \begin{cases} 1 & \text{si } (x_i \sim x_j) \\ 0 & \text{autrement.} \end{cases}$$

- Contrainte de régularité sur la variété : $x_i \sim x_j$ $\Rightarrow$ faibles variations des étiquettes :

$$\begin{aligned} \sum_{x_i \sim x_j} (f_i - f_j)^2 &= 2 \sum_i f_i^2 D_i - 2 \sum_{x_i \sim x_j} f_i f_j \\ &= 2\mathbf{f}^\top \underbrace{(D - W)}_{L} \mathbf{f} \end{aligned}$$

où $x_i \sim x_j$ si x_i et x_j sont voisins , $D_i = \sum_j W_{ij}$. L est le Laplacien du graphe et $f_i = f(x_i) = \sum_{k=1}^{n} \beta_k \mathbf{k}(x_i, x_k) + b$.



Graphe de similarité

- **Graphe** : $G(V, E)$, V : points $\{x_i, i = [\![n]\!]\}$, $n = \ell + u$ et E : arcs.
- **Matrice d'adjacence** : $W \in \mathbb{R}^{n \times n}$:

$$W_{ij} = \begin{cases} 1 & \text{si } (x_i \sim x_j) \\ 0 & \text{autrement.} \end{cases}$$

Pénalisations du Laplacien SVM

- Complexité : $\|f\|^2_{\mathcal{H}} = \beta^T K \beta$,
- Régularité sur la variété : $\|f\|^2_{\mathcal{M}} = \beta^T K L K \beta$,

avec K matrice de Gram $K_{ij} = k(x_i, x_j)$ et L Laplacien du graphe.

Problème dual

A partir des conditions d'optimalité, on obtient :

$$\begin{aligned} \max_{\boldsymbol{\alpha}\in\mathbb{R}^{l}} \quad & -\boldsymbol{\alpha}^\top Q\boldsymbol{\alpha} + \boldsymbol{\alpha}^\top \mathbb{1} \\ \text{s.t.} \quad & \boldsymbol{\alpha}^\top \mathbf{y} = 0, \\ & 0 \leq \alpha_i \leq 1, \quad i = 1, \ldots, \ell \end{aligned}$$

avec $Q = YJK(\lambda I_n + \mu LK)^{-1} J^\top Y$, $J = \begin{bmatrix} I_\ell & 0_{\ell \times u} \end{bmatrix}$ et $n = \ell + u$.

La solution du problème primal est :

$$\boldsymbol{\beta} = (\lambda I + \mu LK)^{-1} J^\top Y \boldsymbol{\alpha}$$

- Problème quadratique avec contraintes dépendant seulement des points étiquetés
- Solution duale : $\boldsymbol{\alpha}^* \in \mathbb{R}^\ell$

Problème

$\boldsymbol{\beta}^* \in \mathbb{R}^{\ell+u}$ implique tous les points (étiquetés ou non)
Généralement $\boldsymbol{\beta}^*$ **n**'est pas parcimonieux

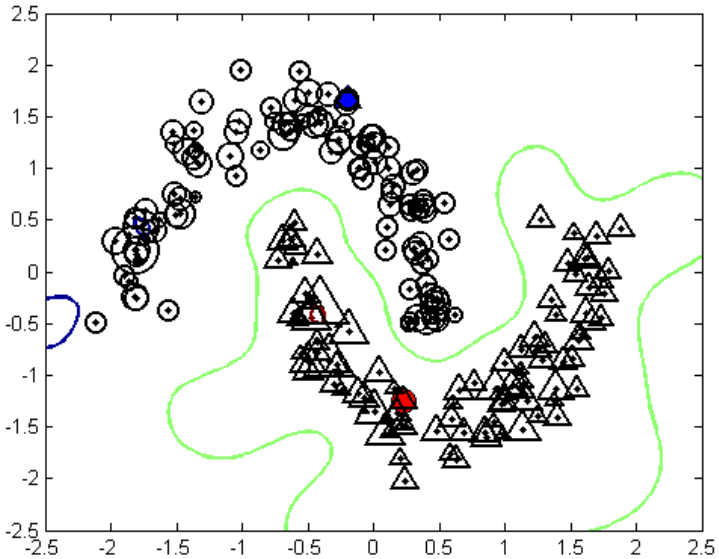


Fig.: Taille des symboles proportionnelle à β_i

Laplacien SVM avec une norme L_1 [Gass 07, Gass 08a, Gass 08b]

Introduction d'une contrainte de parcimonie via une norme L_1

$$\begin{array}{ll} \min_{\beta,b,\xi} & \sum_{i=1}^{\ell} \xi_i + \beta^\top P \beta \\ \text{s. c.} & y_i f(x_i) \geq 1 - \xi_i, \quad \forall i = 1, \ldots, \ell \\ & \xi_i \geq 0, \qquad \forall i = 1, \ldots, \ell \\ & \sum_{j=1}^{\ell+u} |\beta_j| \leq s \end{array} \tag{1}$$

où $P = \lambda K + \mu KLK$.

Si s est petit, certains coefficients β_j sont forcés à 0.

Question

Comment évolue le degré de parcimonie du modèle $f(x)$ si l'hyper-paramètre s varie dans l'intervalle $[0, \infty]$?

Laplacien SVM avec une norme L_1 [Gass 07, Gass 08a, Gass 08b]

Introduction d'une contrainte de parcimonie via une norme L_1

$$\begin{array}{ll} \min_{\beta,b,\xi} & \sum_{i=1}^{\ell} \xi_i + \beta^\top P \beta \\ \text{s. c.} & y_i f(x_i) \geq 1 - \xi_i, \quad \forall i = 1, \ldots, \ell \\ & \xi_i \geq 0, \qquad \forall i = 1, \ldots, \ell \\ & \sum_{j=1}^{\ell+u} |\beta_j| \leq s \end{array} \tag{1}$$

où $P = \lambda K + \mu KLK$.

Si s est petit, certains coefficients β_j sont forcés à 0.

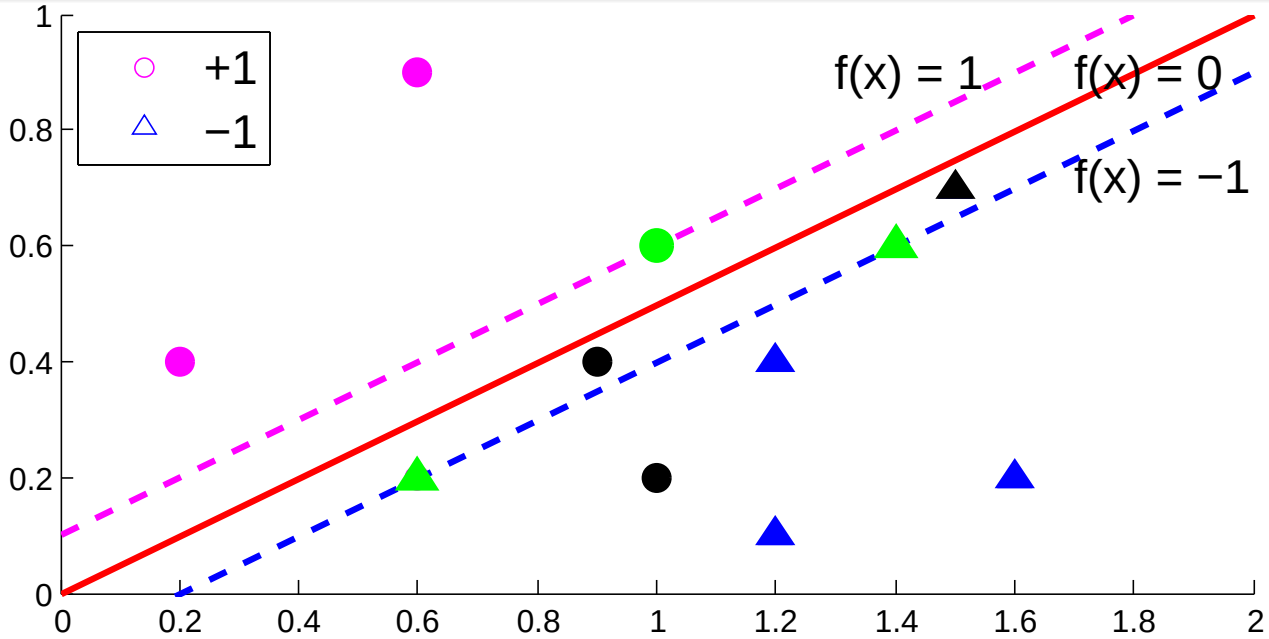
Bonnes nouvelles

Ce problème a une solution linéaire par morceaux quand s varie $\Rightarrow$ chemin de régularisation linéaire par morceaux.

Pour $\boldsymbol{\beta}^* \Leftrightarrow \boldsymbol{\alpha}^*$, on a la répartition suivante :

Ensembles (portant uniquement sur les points étiquetés)

- $\mathcal{I}_\alpha$: $y_i f(x_i) = 1, \quad 0 \leq \alpha_i \leq 1 \rightarrow$ Points sur la Marge
- $\mathcal{I}_1$: $y_i f(x_i) < 1, \quad \alpha_i = 1 \quad \rightarrow$ Points Mal Classées
- $\mathcal{I}_0$: $y_i f(x_i) > 1, \quad \alpha_i = 0 \quad \rightarrow$ Points Bien Classées



Analyse de la structure de la solution

Pour $\boldsymbol{\beta}^* \Leftrightarrow \boldsymbol{\alpha}^*$, on a la répartition suivante :

Ensembles (portant uniquement sur les points étiquetés)

- $\mathcal{I}_\alpha$: $y_i f(x_i) = 1, \quad 0 \leq \alpha_i \leq 1 \rightarrow$ Points sur la Marge
- $\mathcal{I}_1$: $y_i f(x_i) < 1, \quad \alpha_i = 1 \quad \rightarrow$ Points Mal Classées
- $\mathcal{I}_0$: $y_i f(x_i) > 1, \quad \alpha_i = 0 \quad \rightarrow$ Points Bien Classées

Points actifs (portant sur tous les points)

- $\mathcal{A}$: $\{j \mid \beta_j \neq 0\}$

$f(x) = \sum_{i=1}^{n} \beta_i k(x, x_i)$ est entièrement déterminée par la connaissance de ces ensembles.

Avec les conditions nécessaires d'optimalité, un système linéaire en fonction du changement de s peut être obtenu. $\exists\ \Delta s$ tel que pour $s = s^t + \Delta s$, les ensembles $\mathcal{I}_0, \mathcal{I}_\alpha, \mathcal{I}_1$ et $\mathcal{A}$ restent inchangés et on a :

$$\Theta = \Theta^t + (s - s^t)\boldsymbol{\eta}$$

- Θ représente **l'ensemble des paramètres** : $\Theta = \begin{bmatrix}\boldsymbol{\beta}_{\mathcal{A}}^\top & \boldsymbol{\alpha}^\top & b & \lambda\end{bmatrix}^\top$,
- $\boldsymbol{\eta} = H^{-1}\mathbf{z}$
- $\mathbf{z} = \begin{bmatrix}\mathbf{0}_{\mathcal{A}}^\top & \mathbf{0}_{\mathcal{I}_\alpha}^\top & 0 & 1\end{bmatrix}^\top$,
- $H = \begin{bmatrix} P_{\mathcal{A},\mathcal{A}} & -K_{\mathcal{A},\mathcal{I}_\alpha} Y_{\mathcal{I}_\alpha} & \mathbf{0}_{\mathcal{A}} & sign(\boldsymbol{\beta}_{\mathcal{A}}) \\ -Y_{\mathcal{I}_\alpha} K_{\mathcal{I}_\alpha,\mathcal{A}} & \mathbf{0}_{\mathcal{I}_\alpha} & -\mathbf{y}_{\mathcal{I}_\alpha} & \mathbf{0}_{\mathcal{I}_\alpha} \\ \mathbf{0}_{\mathcal{A}}^\top & -\mathbf{y}_{\mathcal{I}_\alpha}^\top & 0 & 0 \\ sign(\boldsymbol{\beta}_{\mathcal{A}})^\top & \mathbf{0}_{\mathcal{I}_\alpha}^\top & 0 & 0 \end{bmatrix}$

Un événement dans $\mathcal{I}_0, \mathcal{I}_\alpha, \mathcal{I}_1$ et $\mathcal{A}$ implique un changement en $\boldsymbol{\eta}$.

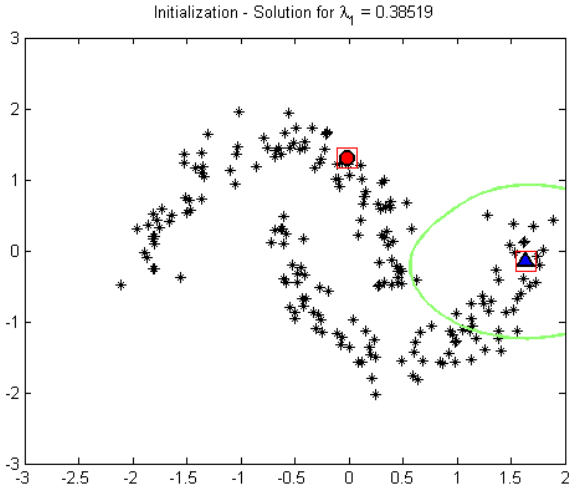
Événements et critère d'arrêt

Avec le système : $s = s^t + \frac{\theta - \theta^t}{\eta_\theta}$ On détecte les événements suivants :

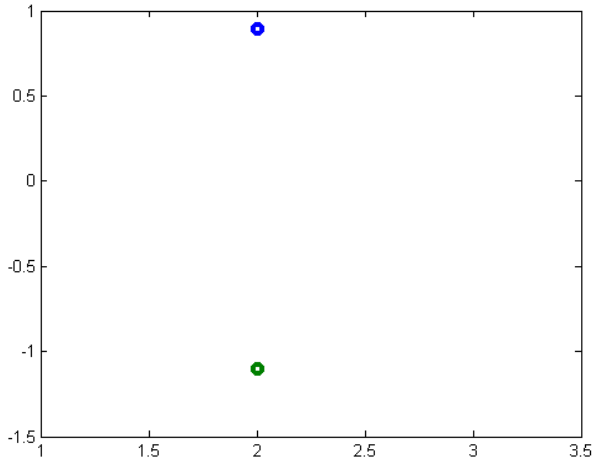
1. $x_i \in \mathcal{I}_\alpha \longrightarrow \mathcal{I}_0 \cup \mathcal{I}_1 \quad (\alpha_i \rightarrow \{0, 1\})$,
2. $x_i \in \mathcal{I}_0 \cup \mathcal{I}_1 \longrightarrow \mathcal{I}_\alpha \quad (y_i f(x_i) = 1)$,
3. $\beta_i, i \in \mathcal{A} \longrightarrow \bar{\mathcal{A}} \quad (\beta_i \neq 0 \rightarrow 0)$,
4. $\beta_i, i \in \bar{\mathcal{A}} \longrightarrow \mathcal{A} \quad (\beta_i = 0 \rightarrow \beta_i \neq 0)$,
5. $\lambda \longrightarrow 0$ (Critère d'arrêt)

Prendre la plus petite valeur $s^{t+1} = s$ telle que $s > s^t$

Application jouet : Two Moons

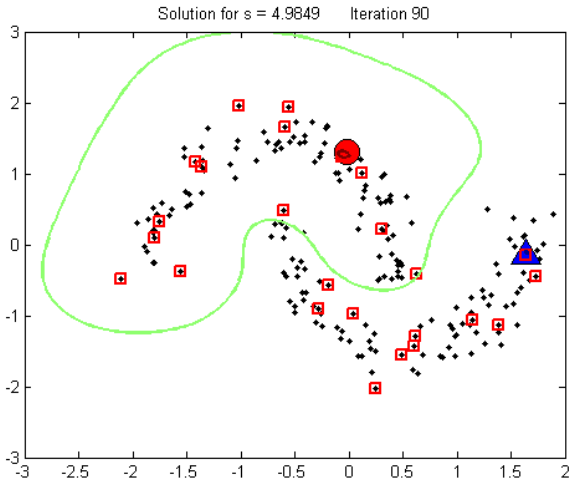


(a) Solution initiale

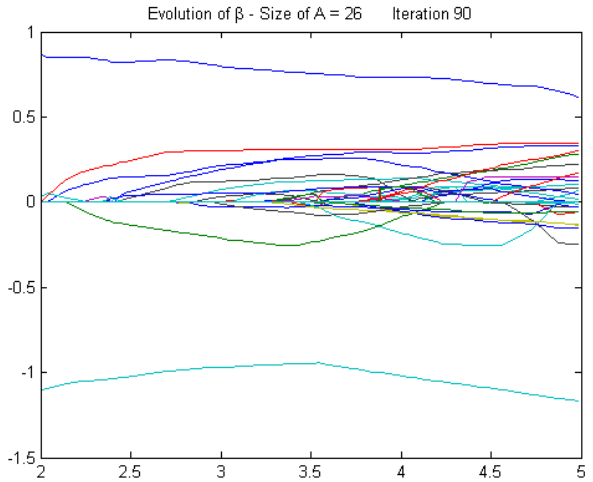


(b) Ensemble actif

Application jouet : Two Moons

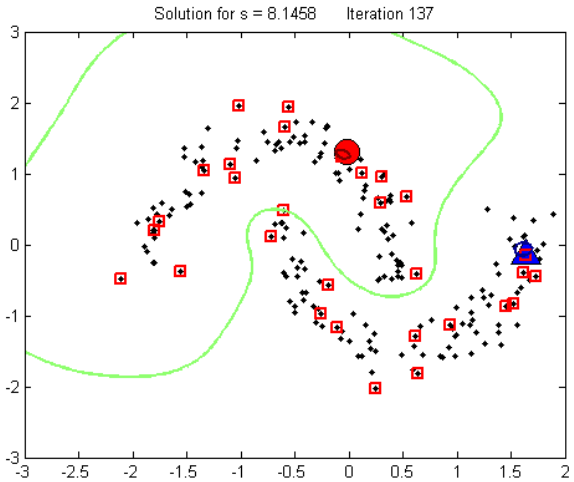


(c) Solution intermédiaire

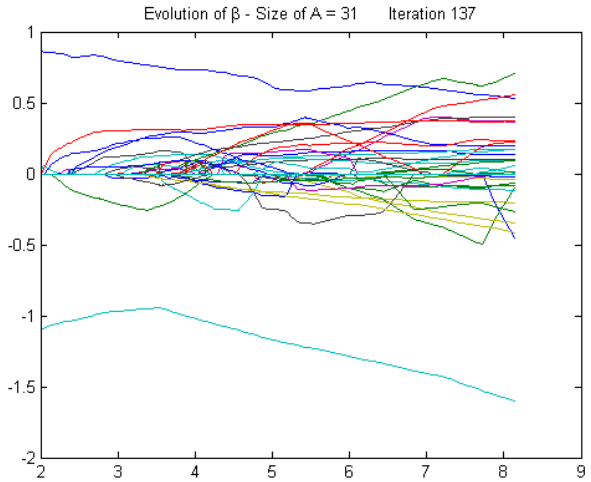


(d) Ensemble actif

Application jouet : Two Moons

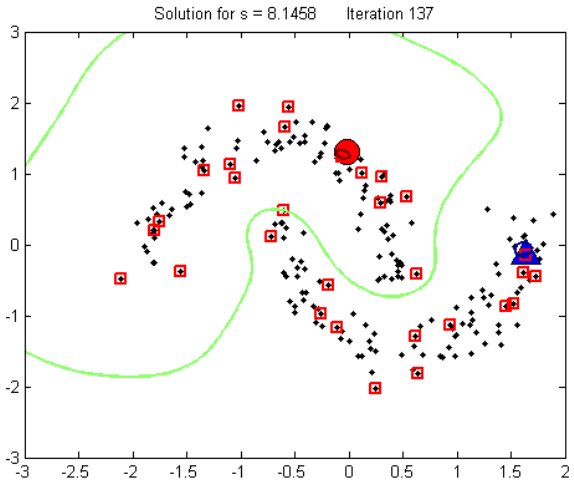


(e) Solution finale

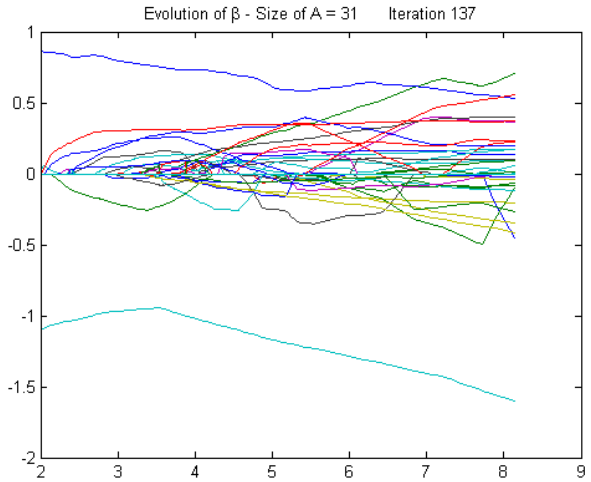


(f) Ensemble actif

Application jouet : Two Moons



(g) Solution finale

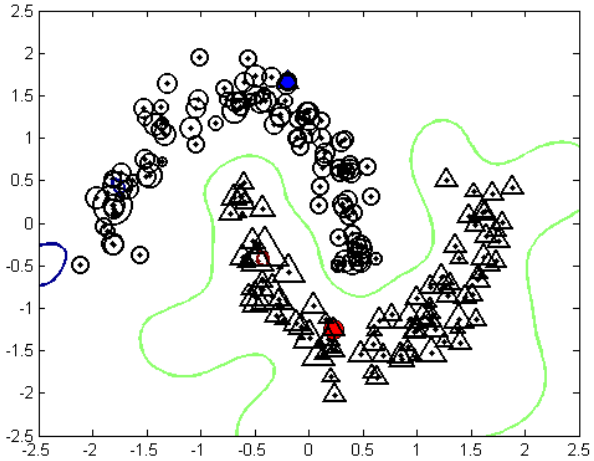


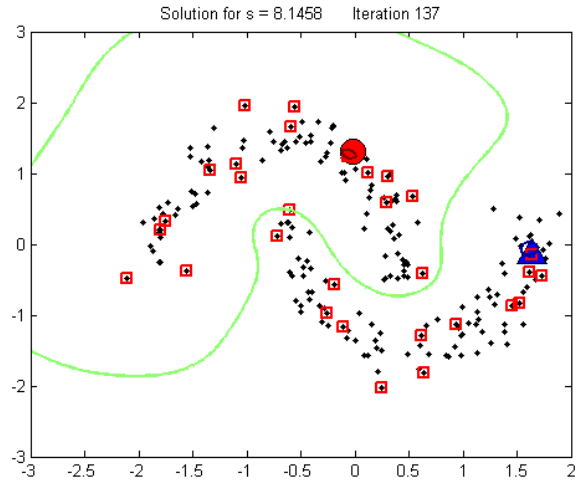
(h) Ensemble actif

ℓ	4	8	16	32	64
$\lvert\mathcal{A}\rvert$	48.87	32.2	28.4	23.8	22.5
$\widehat{\sigma}_{\mathcal{A}}$	12.58	11.69	12.81	6.54	2.27

Tab.: Nombre $|\mathcal{A}|$ de variables sélectionnées en fonction du nombre ℓ de points étiquetés pour une erreur de classification nulle. Paramètres utilisés : noyau Gaussien avec $\sigma = 0.5$, $\lambda = .001$, $\mu = 1$. Les résultats ont été moyennés sur 10 expériences. Le Laplacien SVM de base utilise $n = \ell + u = 200$ variables.

Expériences





Données	Dim.	Erreur		$\lvert\mathcal{A}\rvert$		Rapport
		LapSVM	L_1-LapSVM	LapSVM	L_1-LapSVM	
moons	2	0	0	200	22.5	0.11
1 vs 2	1024	5.6%	4.9%	108	83	0.77
1 vs 3	1024	17.4%	18.1%	108	90	0.83
2 vs 3	1024	29.2%	29.2%	108	88	0.81
Text	7511	10.4%	10.4%	1458	1027	0.7

Tab.: L_1-LapSVM sur les données 2moons, coil et text.

Apprentissage Semi-Supervisé : Récapitulatif

- **Introduction** d'une pénalisation de type L_1 pour obtenir la **parcimonie**
- Calcul du **chemin de régularisation** pour régler cette parcimonie
- **Performances** en classification **aussi bonnes** que le Laplacien SVM basique $\Rightarrow$ classification rapide de nouveaux points !

Conclusion

- Dérivation de la solution optimale en utilisant des sous-différentielles
- Chemin de régularisation linéaire par morceaux :
  - **calcul rapide et efficace de toutes les solutions**
  - partition efficace de l'espace des paramètres
  - **Bonne méthode de recherche de modèle optimal**
- **Diminution de la complexité** de problèmes d'**ordonnancement** avec un graphe réduit
- **Introduction** d'une pénalisation de type L_1 pour obtenir la parcimonie en problèmes semi-supervisés
- Extension du **chemin de régularisation** pour **RankSVM** et L_1-**LapSVM**
- Dérivation d'un algorithme pour le **nettoyage du graphe** des voisins

Perspectives

- Réalisation des expériences sur d'autres données
- Dérivation des chemins multi-dimensionels pour différents paramètres
- Utilisation des méthodes d'optimisation non-régulière et stochastique pour la recherche du modèle

- Réalisation des expériences sur d'autres données
- Dérivation des chemins multi-dimensionels pour différents paramètres
- Utilisation des méthodes d'optimisation non-régulière et stochastique pour la recherche du modèle

Merci !

References I

M. Belkin, P. Niyogi, and V. Sindhwani.
"Manifold Regularization : A Geometric Framework for Learning from Labeled and Unlabeled Examples.".
Journal of Machine Learning Research, Vol. 7, pp. 2399–2434, 2006.

L. Bottou and O. Bousquet.
"Learning Using Large Datasets".
In : *Mining Massive DataSets for Security*, IOS Press, Amsterdam, 2008.
to appear.

O. Chapelle.
"Training a Support Vector Machine in the Primal".
Neural Computation, Vol. 19, No. 5, pp. 1155–1178, May 2007.

O. Chapelle and S. S. Keerthi.
"Efficient algorithms for ranking with SVMs".
Information Retrieval Journal, Special Issue on Learning to Rank, 2009.

F. H. Clarke, Y. S. Ledyaev, R. J. Stern, and P. R. Wolenski.
Nonsmooth analysis and control theory.
Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1998.

References II

T.-M.-T. Do and T. Artières.
"A Fast Method for Training Linear SVM in the Primal".
In : *ECML PKDD '08 : Proceedings of the 2008 European Conference on Machine Learning and Knowledge Discovery in Databases - Part I*, pp. 272–287, Springer-Verlag, Berlin, Heidelberg, 2008.

G. Gasso, K. Zapién, and S. Canu.
"Sparsity Regularization Path for Semi-Supervised SVM".
In : *Proceedings of 4th International Conference in Machine Learning and Applications*, pp. 25–30, IEEE Computer Society, Los Alamitos, CA, USA, 2007.

G. Gasso, K. Zapién, and S. Canu.
"Apprentissage semi supervisé via un SVM parcimonieux : calcul du chemin de régularisation".
Revue I3 - Information Interaction Intelligence, Vol. 8, No. 2, 2008.

G. Gasso, K. Zapién, and S. Canu.
"Apprentissage semi supervisé via un SVM parcimonieux : calcul du chemin de régularisation".
In : *RFIA 2008*, Amiens, France, 2008.

References III

T. Hastie, S. Rosset, R. Tibshirani, and J. Zhu.
"The Entire Regularization Path for the Support Vector Machine".
Journal of Machine Learning Research, Vol. 5, pp. 1391–1415, October 2004.

T. Joachims.
"Training linear SVMs in linear time".
In : *KDD '06 : Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 217–226, ACM, New York, NY, USA, 2006.

G. S. Kimeldorf and G. Wahba.
"Some Results on Tchebycheffian Spline Functions".
Journal of Mathematical Analysis and Applications, Vol. 33, No. 1, pp. 82–95, 1971.

G. Loosli.
Méthodes à noyaux pour la détection de contexte.
PhD thesis, INSA de Rouen, Saint Etienne du Rouvray, October 2007.

H. M. Markowitz.
"The utility of wealth".
Journal of Political Economy, Vol. 60, pp. 151–158, 1952.

References IV


M. R. Osborne, B. Presnell, and B. Turlach.
"A New Approach to Variable Selection in Least Squares Problems".
IMA journal of numerical analysis, Vol. 20, No. 3, pp. 389–403, 1999.


J. Platt.
"Fast training of support vector machines using sequential minimal optimization".
In : *Advances in Kernel Methods : Support Vector Learning*, pp. 185–208, MIT Press, Cambridge, MA, USA, 1999.


S. Rosset and J. Zhu.
"Piecewise Linear Regularized Solution Paths".
Annals of Statistics, Vol. 35, No. 3, pp. 1012–1030, 2007.


W. Schirotzek.
Nonsmooth Analysis.
Springer-Verlag, Berlin, Heidelberg, New York, 2007.


S. Shalev-Shwartz, Y. Singer, and N. Srebro.
"Pegasos : Primal Estimated sub-GrAdient SOlver for SVM".
In : *ICML '07 : Proceedings of the 24th international conference on Machine learning*, pp. 807–814, ACM, New York, NY, USA, 2007.

References V

K. Zapién, T. Gärtner, G. Gasso, and S. Canu.
"Model Selection for Ranking SVM Using Regularization Path".
In : *Proceedings of CAp, Conférence d'apprentissage*, Porquerolles, France, 2008.

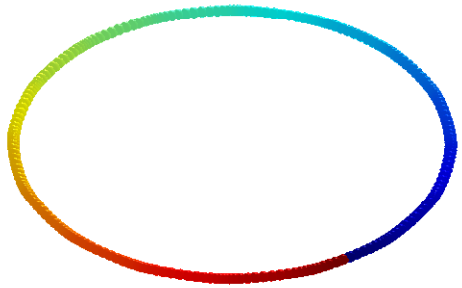
K. Zapién, T. Gärtner, G. Gasso, and S. Canu.
"Regularisation Path for Ranking SVM".
In : *Proceedings of ESANN 2008*, Bruges, Belgium, 2008.

K. Zapién, G. Gasso, T. Gärtner, and S. Canu.
"Model Selection for Ranking SVM Using Regularization Path".
2009.
To appear.

Données Originales :

Données Originales :

Reduction de dimension :

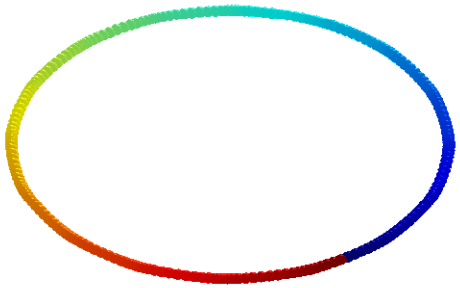


Données Originales :

Topologie locale :

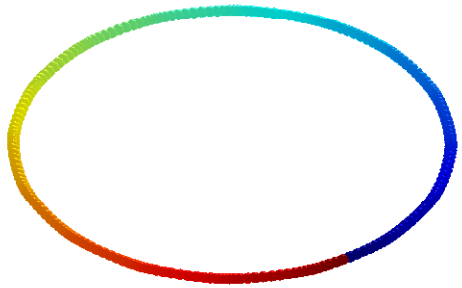
Graphe de voisinage

Reduction de dimension :



Données Originales : Topologie locale : Reduction de dimension :

Graphe de voisinage



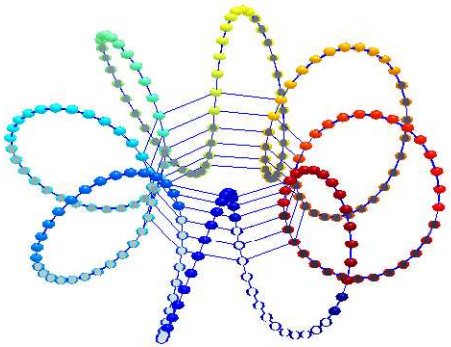
On cherche Diminuir la dimension originelle en gardant la topologie locale : On utilise une graphe de voisinage que la nouveau représentation doit respecter.

Données Originales :

Données Originales :

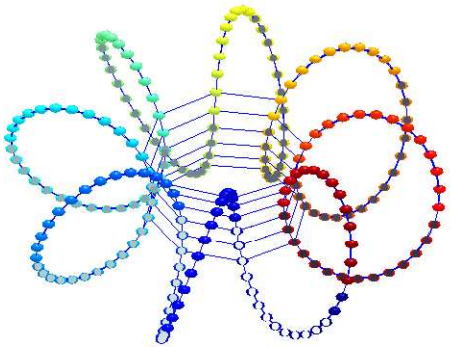
Topologie locale :

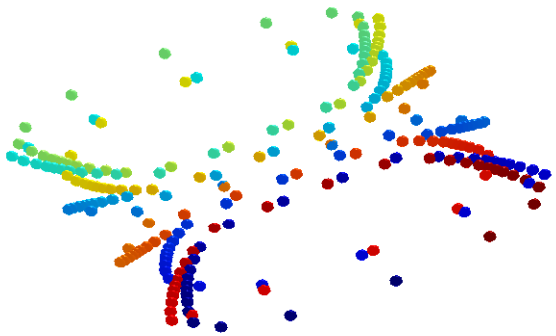
Graphe de voisinage



Données Originales : Topologie locale : Reduction de dimension :

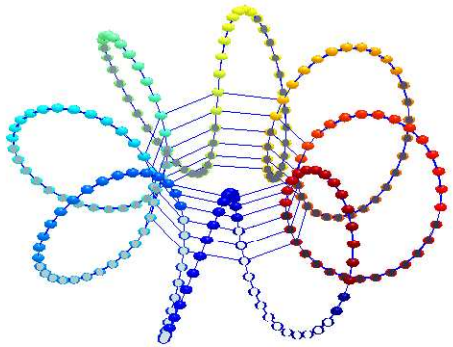
Graphe de voisinage

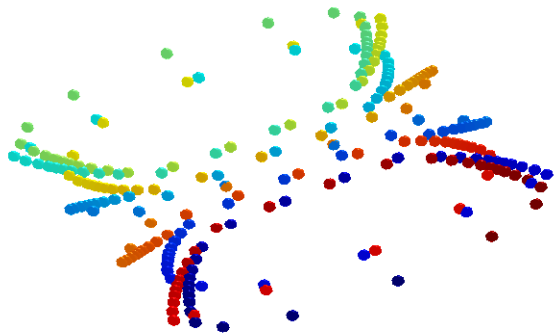




Données Originales : Topologie locale : Reduction de dimension :

Graphe de voisinage

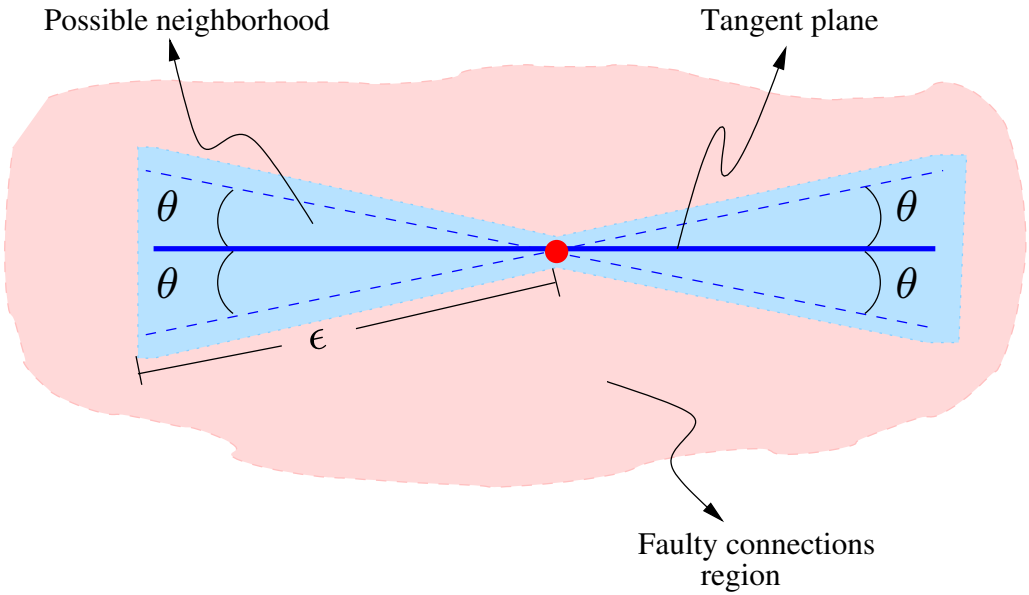


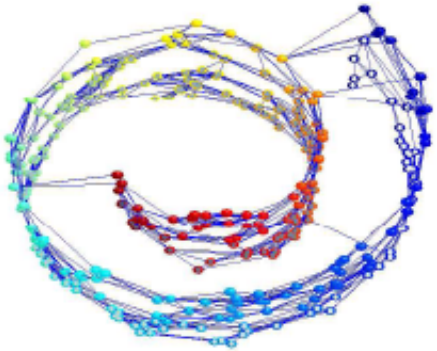


Si le graphe de voisinage donne une **a**ssignation erroné des voisins.
La topologie locale n'est pas respecté :
mauvaise nouvelle représentation.

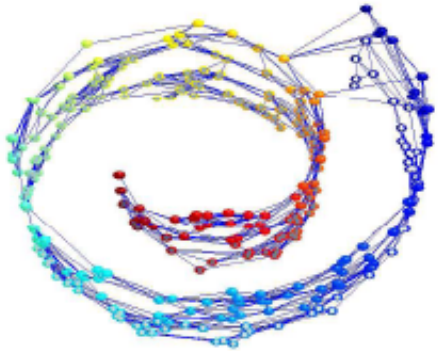
Successive Approximation phase pseudo-algorithm

Input : deviation angles Φ_{ij}, Actual neighborhood $\mathcal{N}_i$, I_{done}
Output : Clean Neighborhood for points $\mathbf{x}_j, j \in \mathcal{N}_i$.
Choose $t = \text{argmin}_j\{\Phi_{ij} |\ i \in I_{done},\ j \in I_i\}$, so that for a point $\mathbf{x}_s,\ s \in I_{done}$, $\mathbf{x}_t \in \mathcal{N}_s$ has the minimum deviation to tangent subspace $\mathcal{P}_s$, let $s = \text{argmin}_i\{\Phi_{it},\ i \in I_{done}\}$.
Select nearest neighbors $\mathcal{N}_t \in \mathcal{N}_t^0$ using Eqs. (??) with $\mathcal{P}_t = \mathcal{P}_s$ and $\theta = \theta + \angle(\mathbf{x}_t - \mathbf{x}_s, \mathcal{P}_s)$.
Update $\mathcal{P}_t(\mathcal{N}_t)$ with Eqs. (??).
Set $I_{done} = I_{done} \cup \{\mathbf{x}_t\}$.
Set $\Phi_{tj} = \angle(\mathbf{w}_t, \mathbf{x}_j - \mathbf{x}_t),\ j \notin I_{done}$. Set $\Phi_{jt} = \infty,\ j \in I_{done}$ to avoid selecting again $\mathbf{x}_t$.
If $|I_{done}| < n$, go to step (3), else go to (9).
If the graph is not connected, make a single component based on $\frac{\Phi_{ij}}{\epsilon_{ij}}$.

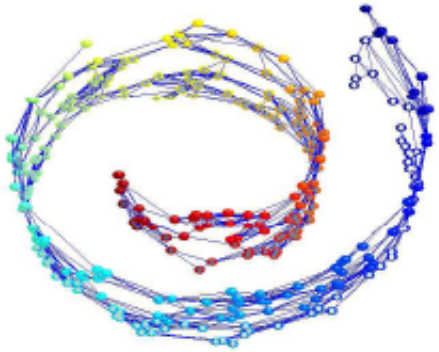

Possible neighborhood
Tangent plane
θ
θ
θ
θ
ϵ
Faulty connections region



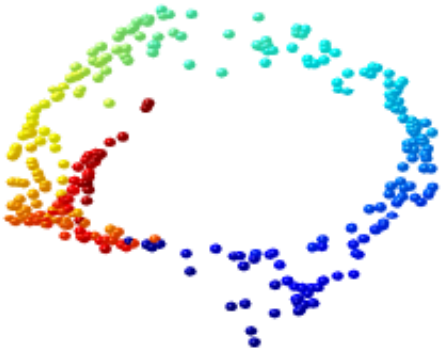
(a) k-NN, $k = 10$



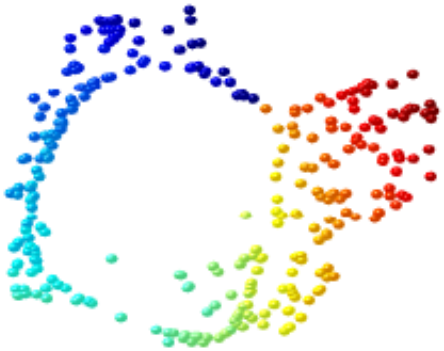
(b) Flows use



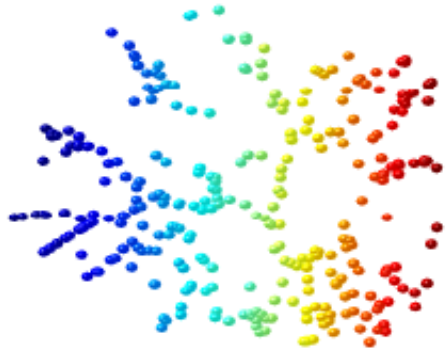
(c) Tangents Propagation



(d) Embedding using k-NN, $k = 10$, Isomap method

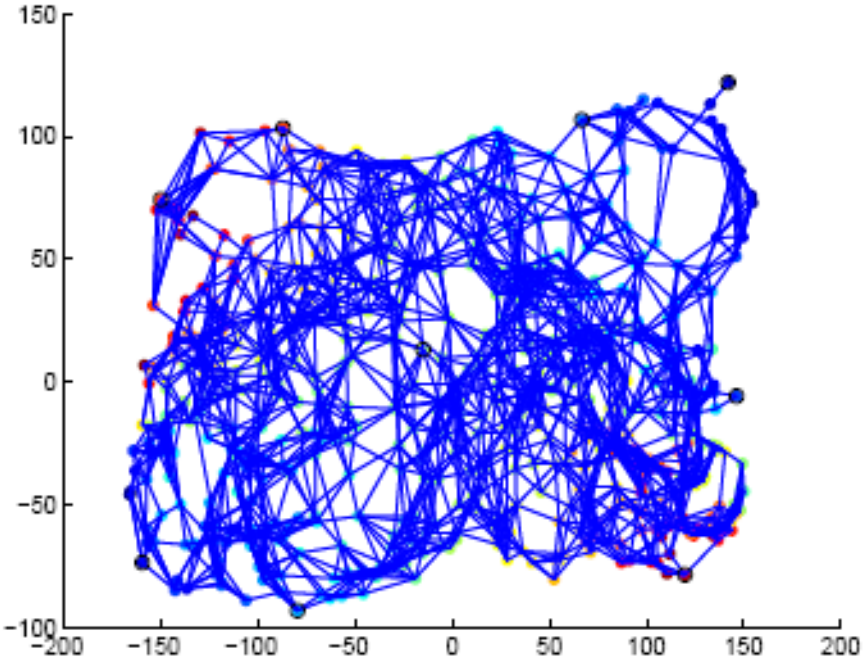


(e) Embedding using Flows use, Isomap method

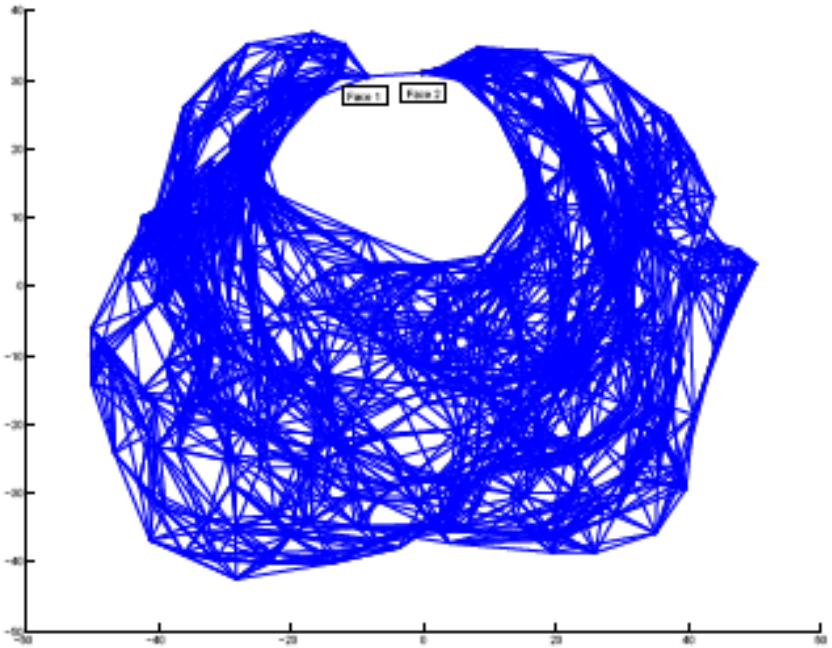


(f) Embedding using Tangents Propagation, Isomap method

Expériences



Exemple avec la base de données *faces*. Used $k = 10$ plus la méthode de planes tangentes. Méthode Isomap.



Exemple avec la base de données *faces*. Used $k = 10$. Méthode Isomap.