1. **Principe**
   - Pulvérisation
   - dimension de Vapnik–Chervonenkis

2. **SVM**
   - SVM linéaire
   - SVM à kernel

Décomposition du risque

$$D(\tilde{F}) - ROPT = D(\tilde{F}) - \min_{\tilde{g} \in Z} \mathbb{E}(LF(\tilde{g}(X), Y))$$
$$+ \min_{\tilde{g} \in Z} \mathbb{E}(LF(\tilde{g}(X), Y)) - ROPT$$

Comment optimiser Z ?

Pulvérisation

Un ensemble $X \subset E$ est dit pulvérisé par une famille de sous-ensembles $H = (H_i \subset E)$ si :

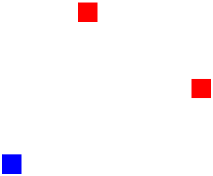
$$H \cap X \supset 2^X$$

Autrement dit, si tout sous-ensemble de X peut être associé à un élément de H.

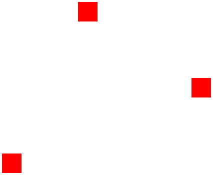
Exemple

Si X est formé de 3 points non alignés du plan E, il est pulvérisé par H l'ensemble des demi-plans.

Exemple



Exemple



exercice

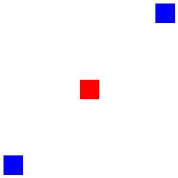
Exercice

Comment peut-on pulvériser un ensemble de trois points alignés ?

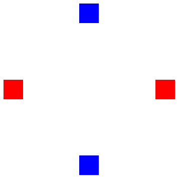
Exercice

Comment peut-on pulvériser un ensemble de quatre points situés aux sommets d'un carré ?

Exemple



Exemple



Pulvérisation par une famille d'estimateurs

Un ensemble $X \subset E$ est dit pulvérisé par une famille d'estimateurs $F = (f_i \in G^E)$ si il est pulvérisé par ses images réciproques $(f_i^{-1}(y))$

Autrement dit, si tout sous-ensemble de X peut être discriminé par un élément de F.

Perceptron

Le perceptron (mono-couche) à n entrées de coefficients w et de seuil θ est défini par :

$$f : E = \mathbb{R}^p \longrightarrow \{0, 1\}$$

$$f(x) = 1 \Leftrightarrow w \cdot x > \theta$$

Exercice

Exercice

Géométriquement, à quoi correspond un perceptron ? Combien de points non coplanaires est-il capable de pulvériser pour un p donné ?

1. **Principe**
   - Pulvérisation
   - dimension de Vapnik–Chervonenkis
2. **SVM**
   - SVM linéaire
   - SVM à kernel

dimension de Vapnik–Chervonenkis

$$VC(H) = \max\{i \in \mathbb{N}, \exists X, |X| = i \vee H \cap X \supset 2^X\}$$

Pour une famille d'estimateurs, la dimension de Vapnik–Chervonenkis correspond à la taille **maximale** d'un ensemble que cette famille peut pulvériser.

$$D(\tilde{F}) - ROPT = D(\tilde{F}) - \min_{\tilde{g} \in Z} \mathbb{E}(LF(\tilde{g}(X), Y)) + \min_{\tilde{g} \in Z} \mathbb{E}(LF(\tilde{g}(X), Y)) - ROPT$$

Comment optimiser Z ?

dimension et convergence (1)

Si $VC(Z)$ est bornée,

$$D(\tilde{F}) - \min_{\tilde{g} \in Z} \mathbb{E}(LF(\tilde{g}(X), Y)) \longrightarrow 0$$

dimension et convergence (1)

Si $VC(Z)$ est bornée,

$$D(\tilde{F}) - \min_{\tilde{g} \in Z} \mathbb{E}(LF(\tilde{g}(X), Y)) \longrightarrow 0$$

Si $VC(Z)$ n'est pas bornée,

$$\min_{\tilde{g} \in Z} \mathbb{E}(LF(\tilde{g}(X), Y)) - ROPT$$

est potentiellement meilleure.

dimension et convergence (1)

Si les (X_i, Y_i) sont iid et

$$n = \Theta\left(\frac{VC(Z) - \log(\delta)}{\epsilon^2}\right)$$

Alors :

$$\mathbb{P}\left(D(\tilde{F}) - \min_{\tilde{g}\in Z} \mathbb{E}(LF(\tilde{g}(X), Y)) < \epsilon\right) > 1 - \delta$$

Une situation idéale

On peut trouver des familles croissantes d'estimateurs Z_k tels que :

$$(X_i, Y_i)_{i \leq n}, n \to \infty$$

$$k \longrightarrow \infty$$

$$VC(Z_k) \to \infty$$

$$VC(Z_k)\frac{\log n}{n} \longrightarrow 0$$

ET

$$\min_{\tilde{g} \in Z_k} \mathbb{E}(LF(\tilde{g}(X), Y)) - ROPT \longrightarrow 0$$

1. Principe
   - Pulvérisation
   - dimension de Vapnik–Chervonenkis

2. SVM
   - SVM linéaire
   - SVM à kernel

Cas binaire

$$G = \{-1, 1\}$$

$$LF = \mathbf{1}_{Y \neq Y'}$$

$$Z = \{x \mapsto a + b \cdot x, (a, b) \in E\}$$

Un ensemble est séparable si :

$$\exists g \in Z, \forall i, g(X_i) > 0 \Leftrightarrow Y_i = 1$$

Marge optimale

On veut optimiser :

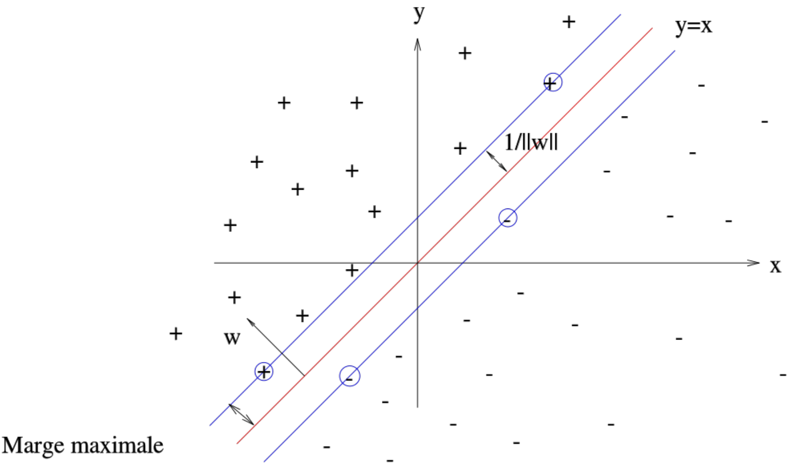
$$\max_{a,b} \min_i d(X_i, \Delta_g)$$

$$= \max_{a,b} \min_i \frac{a + b \cdot X_i}{|b|}$$

$$= \min_{\forall i, (a+b \cdot X_i) Y_i \geq 1} |b|^2$$

Exercice

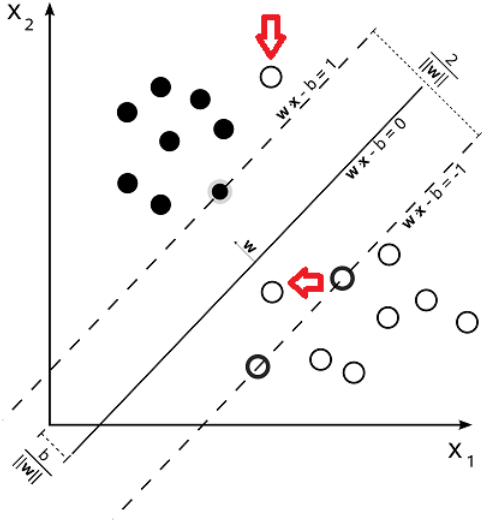
Retrouvez graphiquement ces formules


y
y=x
1/||w||
w
x
Marge maximale

Cas non séparable

On veut optimiser :

$$\min_{\forall i, (a + b \cdot X_i) Y_i \geq 1 - c_i} |b|^2 + p \sum c_i$$


X 2
X 1
w x - b = 1
w x - b = 0
w x - b = -1
2 / ||w||
b / ||w||
w

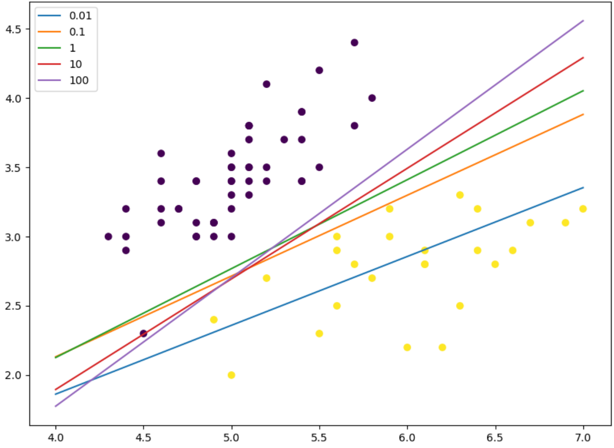
On se ramène donc à un problème d'optimisation classique :

- méthodes quadratiques,
- méthodes de gradient,
- complexité dépendante de la pénalisation

Exercice

A partir des données iris, en vous restreignant aux largeurs/longueurs de sépales et aux deux premières classes, écrivez un programme qui en fonction d'une pénalité p trouve la séparation optimale. Affichez la droite de séparation pour chaque pénalité.

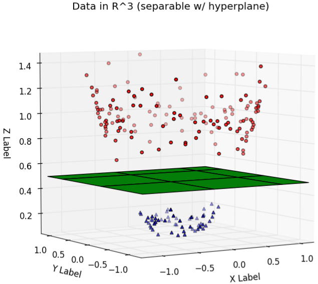
résultat

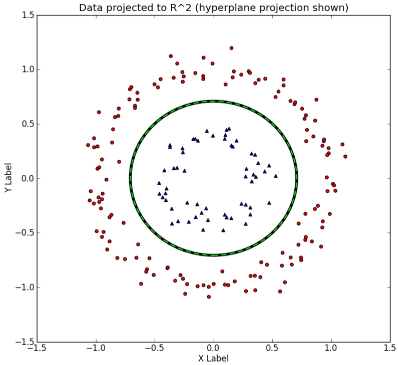


solution

```python
from sklearn import datasets,svm
from matplotlib import pyplot as plt
import numpy as np
iris = datasets.load_iris()
X,Y = iris.data[:,0:2], iris.target
X_train, X_test, Y_train, Y_test = X[:75],X[75:],Y[:75],Y[75:]
ref = list(np.linspace(4,7,100))
plt.scatter(X_train[:,0], X_train[:,1],c=Y_train)
for p in range(5):
    vm = svm.LinearSVC(C=10**(p-2))
    vm.fit(X_train, Y_train)
    b,a = list(vm.coef_[0]),vm.intercept_[0]
    plt.plot(ref,[-b[0]*x/b[1]-a/b[1] for x in ref],
        label=str(10**(p-2)))
plt.legend()
plt.show()
```

Séparabilité et dimension





Séparabilité et dimension

Comme pour la pulvérisation, l'espérance du nombre de points séparables vérifie $n \sim 2p$ où p est la dimension de E.

Séparabilité et dimension

Comme pour la pulvérisation, l'espérance du nombre de points séparables vérifie $n \sim 2p$ où p est la dimension de E.

On considère $\Phi : E \to E'$ où $dim(E') > dim(E)$.

Séparabilité et dimension

Comme pour la pulvérisation, l'espérance du nombre de points séparables vérifie $n \sim 2p$ où p est la dimension de E.

On considère $\Phi : E \rightarrow E'$ où $dim(E') > dim(E)$.

Et le problème devient

$$\min_{\forall i, (a+b \cdot \Phi(X_i)) Y_i \geq 1 - c_i} |b|^2 + p \sum c_i$$

Problème Dual

Problème primal :

$$\min_{\forall i,(a+b\cdot\Phi(X_i))Y_i\geq 1-c_i} |b|^2 + p\sum c_i$$

Problème dual :

$$\max_{\sum a_i Y_i=0,\forall i,0\leq a_i\leq p} 2\sum a_i - \sum\sum a_i a_j Y_i Y_j \Phi(X_i)\cdot\Phi(X_j)$$

Problème avec Kernel

Problème primal :

$$\min_{\forall i,(a+b\cdot\Phi(X_i))Y_i\geq 1-c_i} |b|^2 + p\sum c_i$$

Problème kernelisé :

$$\max_{\sum a_i Y_i=0,\forall i,0\leq a_i\leq p} 2\sum a_i - \sum\sum a_i a_j Y_i Y_j k(X_i, X_j)$$