

MLEA : Machine Learning

Linear Support Vector Machines

Réda DEHAK
reda.dehak@epita.fr

EPITA

22 april 2024

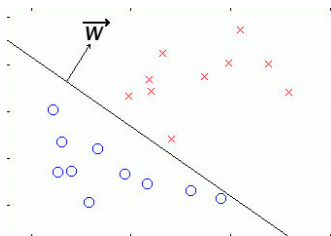
- 1 Linear Support Vector Machines
- 2 Linear Learning Machines
 - Perceptron
- 3 Soft Margin Linear SVM
- 4 Multiclass vs. Binary Classifier

History

- Classifier derived from statistical Learning theory by Vapnick and Chervonenkis.
- Introduced in COLT¹-1992 by Boser, Guyon and Vapnik
- Kernel Machines: Large class of learning algorithms, SVMs a particular instance
- Centralized website: www.kernel-machines.org
- Successful applications in many fields (text, image recognition, bioinformatics, ...)
- An important and active field of all Machine Learning research: A large and diverse community, machine learning, optimization, statistics, neural networks, functional analysis, etc.

¹Annual Conference on Learning Theory

Linear Algebra



- Inner product between vectors :

$$\langle x, z \rangle = x^T z = \sum_i x_i z_i$$

- Hyperplane:

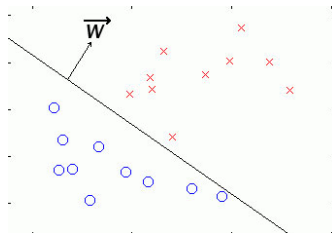
$$\langle w, x \rangle + b = 0$$

- 1 Linear Support Vector Machines
- 2 Linear Learning Machines
 - Perceptron
- 3 Soft Margin Linear SVM
- 4 Multiclass vs. Binary Classifier

Linear Learning Machines

- Classification : Decision Function is a hyperplane in input space
- The Perceptron algorithm (Rosenblatt, 57)
- Useful to analyse the Perceptron algorithm, before looking at SVMs and Kernel Methods in general

Perceptron

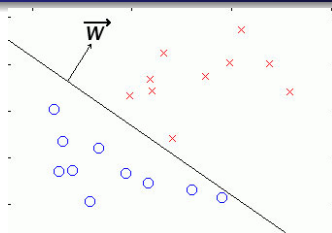


Linear Separation of the input space

$$f(x) = \langle w, x \rangle + b$$

$$h(x) = \text{sign}(f(x))$$

Perceptron Algorithm



Optimisation criteria:

$$\arg \min_{w,b} \|f(x) - y\|^2$$

Update rule :

if $y_i (< w_k, x_i > + b_k) \leq 0$ then

$$w_{k+1} \leftarrow w_k + \eta y_i x_i$$

Observations

- Solution is a linear combination of training points

$$w = \sum_i \alpha_i y_i x_i$$

$$\alpha_i \geq 0$$

- Only used informative points (mistake driven)
- The coefficient of a point in combination reflects its difficulty

Dual representation

- The decision function can be re-written as follows

$$f(x) = \langle w, x \rangle + b = \sum_i \alpha_i y_i \langle x_i, x \rangle + b$$

avec $w = \sum_i \alpha_i y_i x_i$

- and the update rule

$$\text{if } y_i \left(\sum_i \alpha_i y_i \langle x_i, x \rangle + b \right) \leq 0 \text{ then } \alpha_i \longrightarrow \alpha_i + \eta$$

Dual representation

- The decision function can be re-written as follows

$$f(x) = \langle w, x \rangle + b = \sum_i \alpha_i y_i \langle x_i, x \rangle + b$$

avec $w = \sum_i \alpha_i y_i x_i$

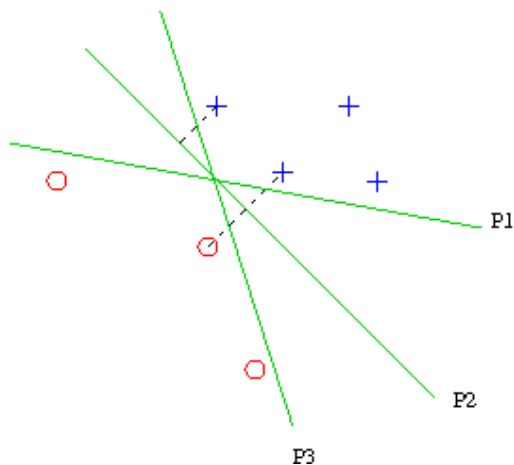
- and the update rule

$$\text{if } y_i \left(\sum_i \alpha_i y_i \langle x_i, x \rangle + b \right) \leq 0 \text{ then } \alpha_i \longrightarrow \alpha_i + \eta$$

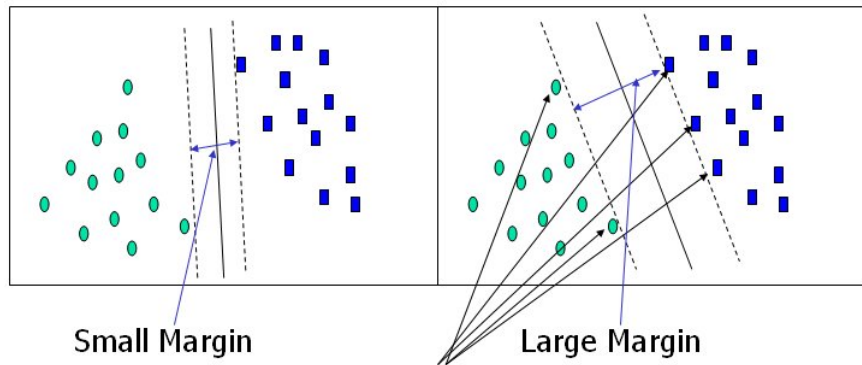
Note:

In dual representation, data appears only inside dot products

Separating Hyperplane



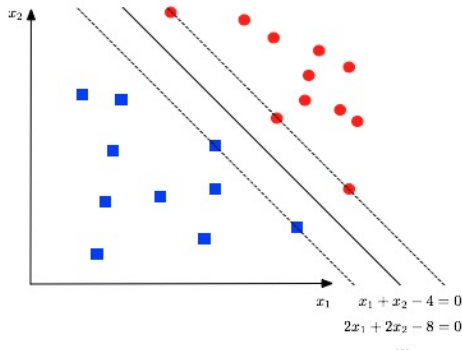
Margin and Support Vectors



Support Vectors

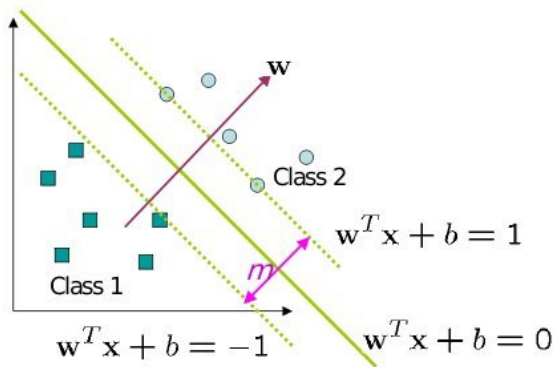
Margin should be large

The decision boundary should be as far from the data as possible
 \Rightarrow the margin m should be maximized



Margin should be large

The decision boundary should be as far from the data as possible
 \Rightarrow the margin m should be maximized

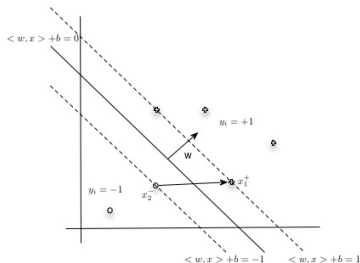


Margin as a function of the weight vector

Let's consider two support vectors x_1^+ et x_2^- :

$$\langle w, x_1^+ \rangle + b = 1 \quad (1)$$

$$\langle w, x_2^- \rangle + b = -1 \quad (2)$$



$$\text{Then } \langle w, x_1^+ \rangle - \langle w, x_2^- \rangle = 2$$

$$\langle w, x_1^+ - x_2^- \rangle = 2$$

$$\|w\| \|x_1^+ - x_2^-\| \cos(\vec{w}, \overrightarrow{x_2^- x_1^+}) = 2$$

$$\|x_1^+ - x_2^-\| \cos(\vec{w}, \overrightarrow{x_2^- x_1^+}) = \frac{2}{\|w\|}$$

$$m = \frac{2}{\|w\|}$$

Hard Margin SVM Problem

- Let x_1, \dots, x_n be our dataset, and $y_i \in \{-1, 1\}$ the class label of x_i
- $\forall i, y_i(w^T x_i + b) \geq 1$
- Resolve the optimization problem:
 - **Maximize margin** $m = \frac{2}{\|w\|} \Rightarrow$ **Minimize norm** of the separating hyperplane $\frac{1}{2} \|w\|^2$
 - **Subject to** $y_i(w^T x_i + b) \geq 1$

Reformulation

- The problem could be reformulated using Lagrange Multiplier method as find w , b , and $\alpha = (\alpha_1, \dots, \alpha_n) \neq 0$ and ≥ 0 minimizing:

$$L(\alpha, w, b) = \frac{1}{2} \|w\|^2 - \sum_i \alpha_i (y_i (w^T x_i + b) - 1)$$

- The maximum margin classifier will be found for $\frac{\partial L}{\partial w} = 0$ and $\frac{\partial L}{\partial b} = 0$
- Reformulate the problem

Reformulation

- The problem could be reformulated using Lagrange Multiplier method as find w , b , and $\alpha = (\alpha_1, \dots, \alpha_n) \neq 0$ and ≥ 0 minimizing:

$$L(\alpha, w, b) = \frac{1}{2} \|w\|^2 - \sum_i \alpha_i (y_i (w^T x_i + b) - 1)$$

- The maximum margin classifier will be found for $\frac{\partial L}{\partial w} = 0$ and $\frac{\partial L}{\partial b} = 0$
- Reformulate the problem
- $\frac{\partial L}{\partial w} = 0 \Rightarrow w = \sum \alpha_i y_i x_i$
- $\frac{\partial L}{\partial b} = 0 \Rightarrow \sum y_i \alpha_i = 0$

Reformulation

- The problem could be reformulated using Lagrange Multiplier method as find w , b , and $\alpha = (\alpha_1, \dots, \alpha_n) \neq 0$ and ≥ 0 minimizing:

$$L(\alpha, w, b) = \frac{1}{2} \|w\|^2 - \sum_i \alpha_i (y_i (w^T x_i + b) - 1)$$

- The maximum margin classifier will be found for $\frac{\partial L}{\partial w} = 0$ and $\frac{\partial L}{\partial b} = 0$
- Reformulate the problem
- $\frac{\partial L}{\partial w} = 0 \Rightarrow w = \sum \alpha_i y_i x_i$
- $\frac{\partial L}{\partial b} = 0 \Rightarrow \sum y_i \alpha_i = 0$
- $L(\alpha, w, b) = Q(\alpha) = \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle$

Proof

$$\begin{aligned}
 L(\alpha, w, b) &= \frac{1}{2} \|w\|^2 - \sum_i \alpha_i (y_i (w^T x_i + b) - 1) \\
 &= \frac{1}{2} \|w\|^2 - w^T \sum_i \alpha_i y_i x_i - \sum_i \alpha_i y_i b + \sum_i \alpha_i \\
 &= \frac{1}{2} \|w\|^2 - \|w\|^2 - b * 0 + \sum_i \alpha_i \\
 &= \sum_i \alpha_i - \frac{1}{2} \|w\|^2
 \end{aligned}$$

Since

$$\left(\sum_{i=1}^N x_i \right)^2 = \sum_i \sum_j x_i x_j$$

then

$$L(\alpha, w, b) = Q(\alpha) = \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle$$

Support Vector

- Only the data near the hyperplane is usefull to train the classifier
- We don't need to take into account the data far from the hyperplane
- the elements that we will consider to build the hyperplane are called the support vectors
- We want to maximize

$$Q(\alpha) = \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle$$

with constraints

$$\alpha_i \geq 0$$

and

$$\sum_i \alpha_i y_i = 0$$

Support Vector

- this is a recurrent mathematical optimisation problem, many specialized heuristics have been designed to resolve it (SMO²)
- the resulting decision function will be

$$f(x) = w^T x + b = \sum_i \alpha_i y_i \langle x_i, x \rangle + b$$

²SMO: Sequential Minimal Optimization

Finding b

Once the α_i are computed, you know which elements are the support vectors.

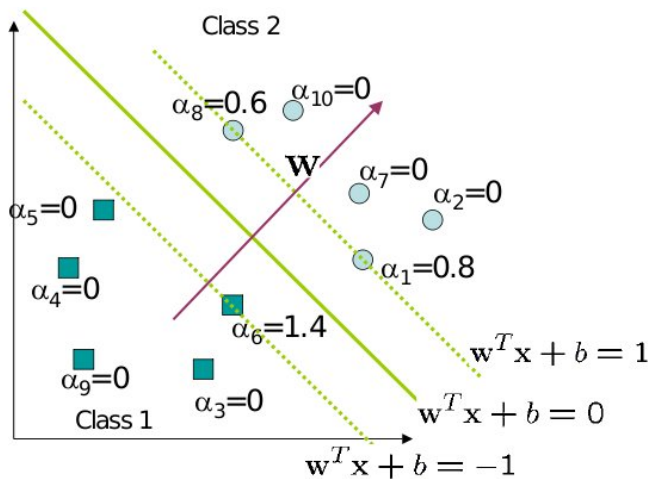
For all support vector x_{SV} , $y_{SV} = w^T x_{SV} + b \Rightarrow b = y_{SV} - w^T x_{SV}$

For a better numerical stability, b can be averaged as:

$$b = \frac{1}{N_{SV}} \sum_{i \in \text{Support Vectors}} y_i - w^T x_i$$

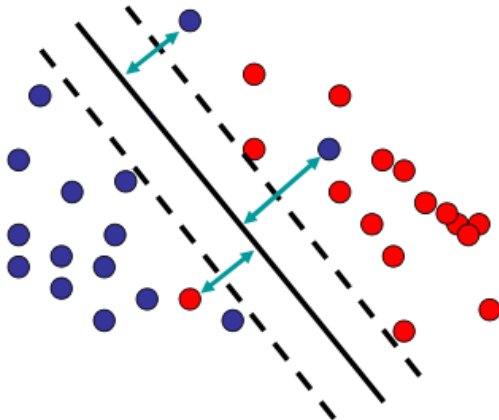
with N_{SV} the number of support vectors

Considerations

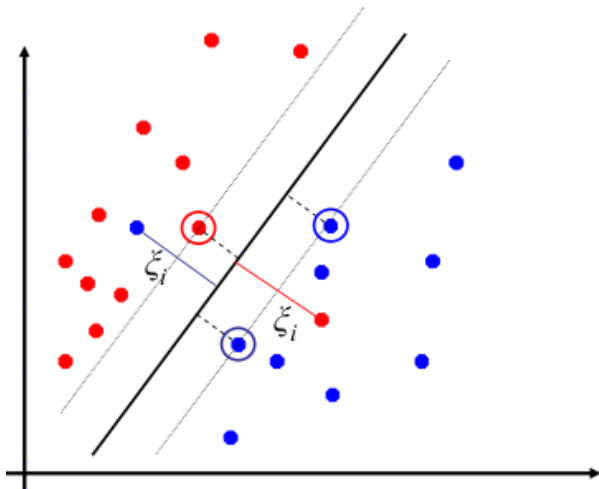


- 1 Linear Support Vector Machines
- 2 Linear Learning Machines
 - Perceptron
- 3 Soft Margin Linear SVM
- 4 Multiclass vs. Binary Classifier

Dealing with noise



Minimizing Errors



Soft Margin SVM formulation

- Minimize $\frac{1}{2}||w||^2 + C \sum \xi_i$
- with $y_i(w^T x_i + b) \geq 1 - \xi_i$
- $\xi_i \geq 0$

Equivalent to maximizing

$$Q(\alpha) = \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j x_i^T x_j$$

with constraints

$$0 \leq \alpha_i \leq C$$

and

$$\sum_i \alpha_i y_i = 0$$

C intuitive meaning

- tradeoff parameter between error and margin

C intuitive meaning

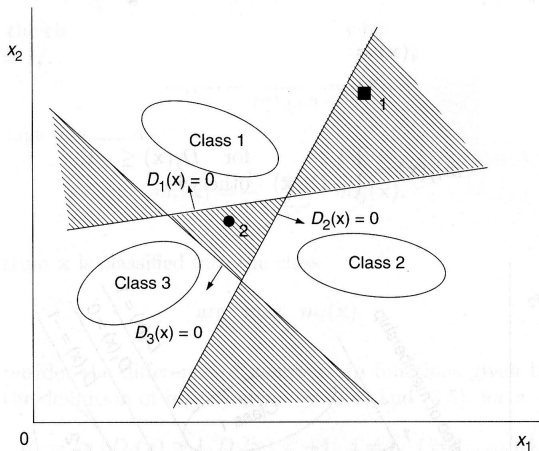
- tradeoff parameter between error and margin
- <http://www.svms.org/parameters/>
- weighting data
- Deal with unbalanced dataset

- 1 Linear Support Vector Machines
- 2 Linear Learning Machines
 - Perceptron
- 3 Soft Margin Linear SVM
- 4 Multiclass vs. Binary Classifier

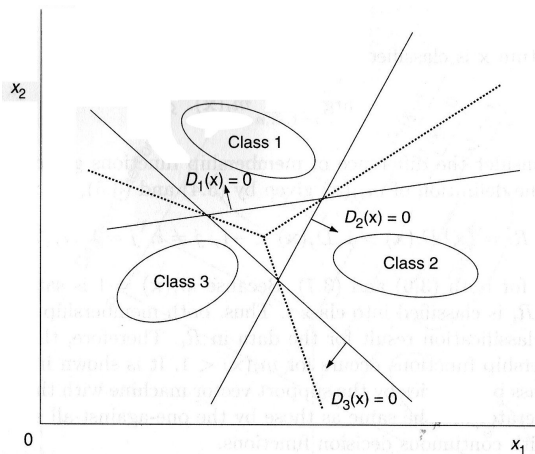
One Against Rest (OAR)

- Train N binary SVM with the whole dataset
- each SVM should discriminate one class versus all other

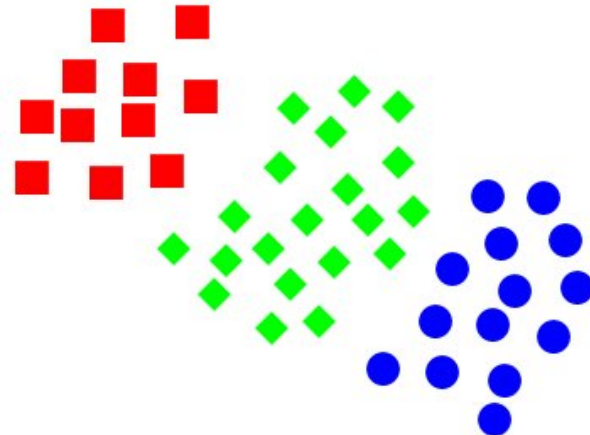
Unclassifiable Regions



Using Decision Function



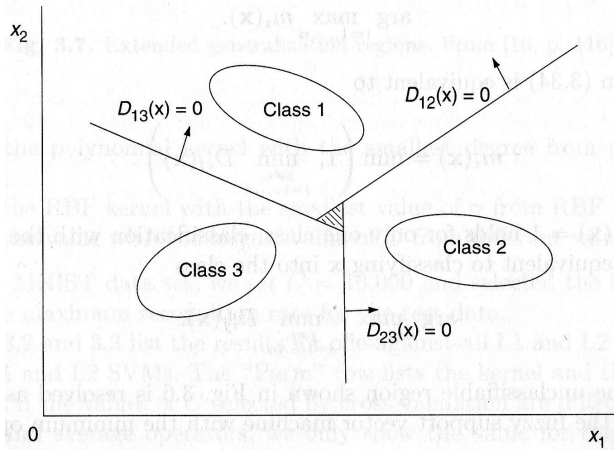
Unadapted Paradigm?



One Against One (OAO)

- build pairwise classifiers
- $\frac{N(N-1)}{2}$ binary classifiers
- decision done through a vote
- smaller training time (the observed speedup is between 3 and 9)
- bigger classification time

Draw...



Fuzzy OAO

Membership function

$$m_{ij}(x) = 1 \text{ for } D_{ij}(x) \geq 1$$
$$m_{ij}(x) = D_{ij}(x) \text{ otherwise}$$

Minimum Operator

$$m_i(x) = \min_{j \neq i} m_{ij}(x)$$

Average Operator

$$m_i(x) = \frac{1}{n-1} \sum_{j \neq i} m_{ij}(x)$$

Classification

$$\operatorname{argmax}(m_i(x))$$

Resolution

