

Support vector machines in the case of non-separable classes

↳ most common in practice

We introduce some auxiliary variables $\xi_1, \dots, \xi_n \geq 0$ (slacks), which express how far the related points (ξ_1 is for $x_1, \dots, \xi_n$ for x_n) are from the margin 'on the wrong side'

The constraint

$$y_i (\beta_0 + x_i^T \beta) \geq M$$

proportional amount in which the point is on the wrong side

is substituted by

$$y_i (\beta_0 + x_i^T \beta) \geq M (1 - \xi_i)$$

$M \xi_i$ → mathematically untractable

The optimization problem becomes

$$\min_{\beta, \beta_0} \|\beta\|^2 + C \sum_{i=1}^n \xi_i \quad \text{subject to} \quad \begin{cases} y_i (\beta_0 + x_i^T \beta) \geq M (1 - \xi_i) \\ \xi_i \geq 0 \end{cases} \quad \forall i=1, \dots, n$$

regularization parameter which controls the amount of penalization for an observation being misclassified (being on the wrong side of the margin)

$C > 0$

The solution of the optimization problem is

$$0 \leq \hat{\alpha}_i \leq C$$

$$\hat{\beta} = \sum_{i=1}^n \hat{\alpha}_i y_i x_i \quad (12.17) \quad \leftarrow \text{supporting vectors}$$

Where $\hat{\alpha}_i$ is related to the constraints and it is non-zero for observations that satisfy $y_i (\beta_0 + x_i^T \beta) \geq M (1 - \xi_i)$

As we saw for many techniques, it is often useful to transform x

$$h(x) = (h_1(x), h_2(x), \dots, h_M(x)) \quad x \in \mathbb{R}^p$$

with M not necessarily equal to p (number of dimensions of x)

Then, the separation rule is

$$f(x) = \beta_0 + h(x)^T \beta = 0$$

which leads to the solution

$$\hat{f}(x) = \hat{\beta}_0 + \sum_{i=1}^n \hat{\alpha}_i y_i \underbrace{\langle h(x_i), h(x) \rangle}_{\text{can lead to generalization,}} \\ \underbrace{K(x, x') = \langle h(x), h(x') \rangle}_{\text{inner product}}$$

Kernels that can be used:

kernel	$K(x, x')$
d^{th} -degree polynomial	$(1 + \langle x, x' \rangle)^d$
radial basis	$\exp(-\gamma \ x - x'\ ^2)$
Sigmoidal (Neural Network)	$\tanh(\kappa_1 \langle x, x' \rangle + \kappa_2)$

Example with $p=2$ and d (degree of polynomial) = 2

$$\begin{aligned} K(x, x') &= (1 + \langle x, x' \rangle)^2 \\ &= (1 + x_1 x'_1 + x_2 x'_2)^2 \\ &= (1 + (x_1 x'_1)^2 + (x_2 x'_2)^2 + 2x_1 x'_1 + 2x_2 x'_2 + 2x_1 x'_1 x_2 x'_2) \end{aligned}$$

This means, if I want to use $h(x)$, that $M=6$ and

$$\begin{aligned} h_1(x) &= 1 & h_2(x) &= \sqrt{2} x_1 & h_3(x) &= \sqrt{2} x_2 & K(x, x') &= \langle h(x), h(x') \rangle \\ h_4(x) &= x_1^2 & h_5(x) &= x_2^2 & h_6(x) &= \sqrt{2} x_1 x_2 \end{aligned}$$

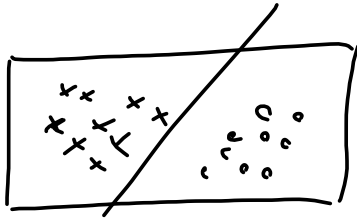
Unsupervised learning

- in contrast to supervised learning, here we do not have a response variable, so the goal is to detect patterns in the data only based on their characteristics ~~(x_i, y_i)~~

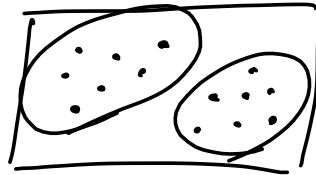
→ exploratory analysis

→ graphical exploration based on the principal components

→ cluster analysis (clustering)



classification
(supervised learning)



clustering
(unsupervised learning)

PCA

- compute the PC from x
- plot the first two PC

