# Machine Learning and Pattern Recognition Practice Session IV

Martin Palazzo

Universite de Technologie de Troyes
Universidad Tecnologica Nacional Buenos Aires
Biomedicine Research Institute of Buenos Aires - Max Planck Partner

*martin.palazzo@utt.fr*

November 2, 2020

# Overview

1. Support Vector Machines - Binary Case

# Disclaimer

The following document has been created as supporting and guiding material during the practical lessons during the Pattern Recognition course of the Master OSS. The official bibliography and theory materials have been distributed previously by the organization committee of the Master OSS.

# Support Vector Machines - Binary Case

# Support Vector Machines

- Support Vector Classification [1] can build a nonlinear rule by constructing a linear boundary in a transformed and high dimensional version of the feature space.
- In binary classification the goal is to estimate a function $f : \mathbb{R} \rightarrow \{+1, -1\}$ from training data samples $x_i$ with label $Y_i$. SVC aims to estimate a hyperplane

$$x : f(x) = x^T \omega + \omega 0 = 0 \tag{1}$$

corresponding to the decision function:

$$D\left(x\right) = sign\left[x^T \omega + \omega_0\right] \tag{2}$$

# Support Vector Machines

The decision function $D$ obtained corresponds to the hyperplane which maximizes the separating margin $M$ between the two classes where $M = 1/\|\omega\|$. Now supose that both classes overlap and are not linearly separable. A set of slack variables $\xi = (\xi_1, ..., \xi_m)$ are defined to allow some miss classifications when a sample fall on the wrong side of the margin [1].

Then a convex optimization problem is expressed in equation 10 where the cost $C$ parameter penalizes every miss classification.

$$min\frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{m}\xi_i \tag{3}$$

$$s.t.\xi_i \geq 0; y_i\left(x^T\omega + \omega 0\right) \geq 1 - \xi_i \tag{4}$$

# Support Vector Machines

After a quadratic programming solution applying Lagrange Multipliers the solution of $\omega$ is expressed as

$$\hat{\omega} = \sum_{1}^{m} \hat{\alpha}_i y_i x_i \tag{5}$$

with non zero coefficient $\hat{\alpha}_i$ only for the samples lying on the edge of the margin or for the miss classified ones. These samples are known as support vectors [1].

# Support Vector Machines and the Kernel trick

The core idea is to apply a transformation/mapping to the input feature vector $X$ and then use linear models in the new space [1]. The transformation is denoted as $\phi$ where $\phi_m$ corresponds to the $m_{th}$ transformation of $X$ and $m = 1...M$. Then the decision function can be written as

$$f(x) = \phi(x)^T \beta + \beta_0 = \sum_{i=1}^{M} \alpha_i y_i \left\langle \phi(x), \phi(x') \right\rangle + \beta_0 \tag{6}$$

where $\phi(x)$ is used only for inner products. For this reason it is not necessary to determine the transformation $\phi(x)$ but it is required to know the positive and semi-definite kernel function $K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$ responsible to compute the inner products in the transformed space.

- Palazzo, M., Beauseroy, P., Yankilevich, P. (2019). Hepatocellular Carcinoma tumor stage classification and gene selection using machine learning models. Electronic Journal of SADIO (EJS), 18(1), 26-42.
- Bishop, C. M. (2006). Pattern recognition and machine learning. springer.
- Friedman, J., Hastie, T., Tibshirani, R. (2001). The elements of statistical learning (Vol. 1, No. 10). New York: Springer series in statistics.
- Shawe-Taylor, J., Cristianini, N. (2004). Kernel methods for pattern analysis. Cambridge university press.