MACHINE LEARNING

With content adapted from the thesis PhD “Desenvolvimento de Modelos Analíticos de Apoio à Gestão de Instituições do Ensino Superior, com Recurso a Data Mining”, of Maria Martins, 2020

Support Vector Machine

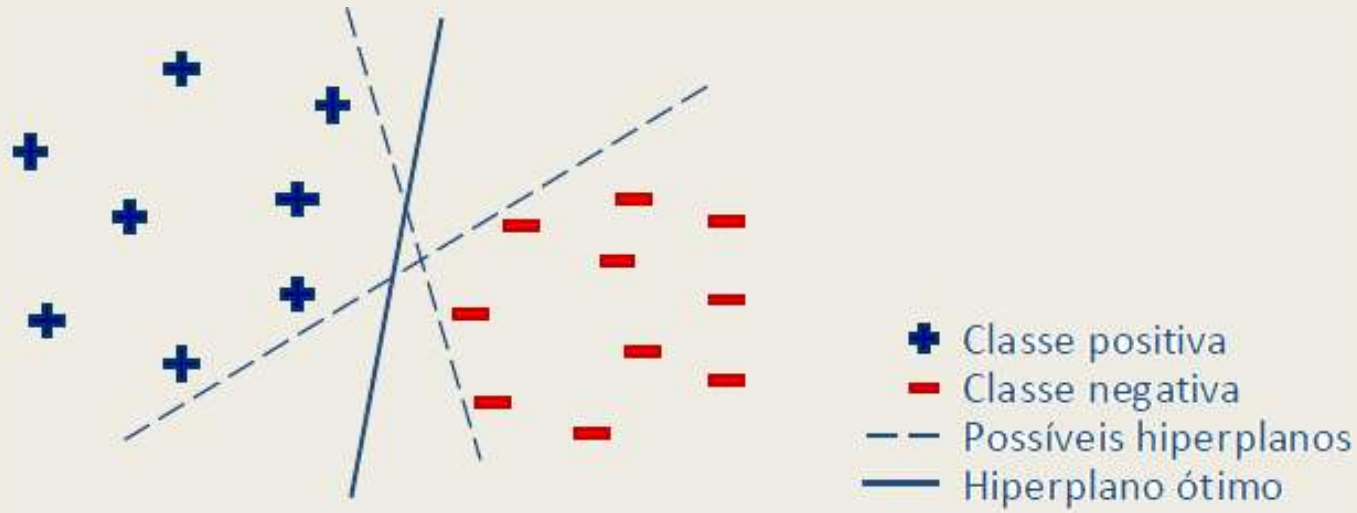
- Support Vector Machine – SVM, is an algorithm developed in 1995 by Vladimir Vapnik[1], Initially for binary classification.
  - *Although nowadays SVM can also be used for regression problems (predict the value of a continuous variable instead of classifying them) and to get the solutions for multiclass issues, We stick only to its original formulation, which already covers an essential category of predicting problems.*

- This is another algorithm that is very popular among the ML community, either for classification or for regression
- So, let's try understanding how it works based on binary classification...

[1] *Vladimir Vapnik. The nature of statistical learning theory. Springer science & business media, 2013*

Optimal Hyperplan

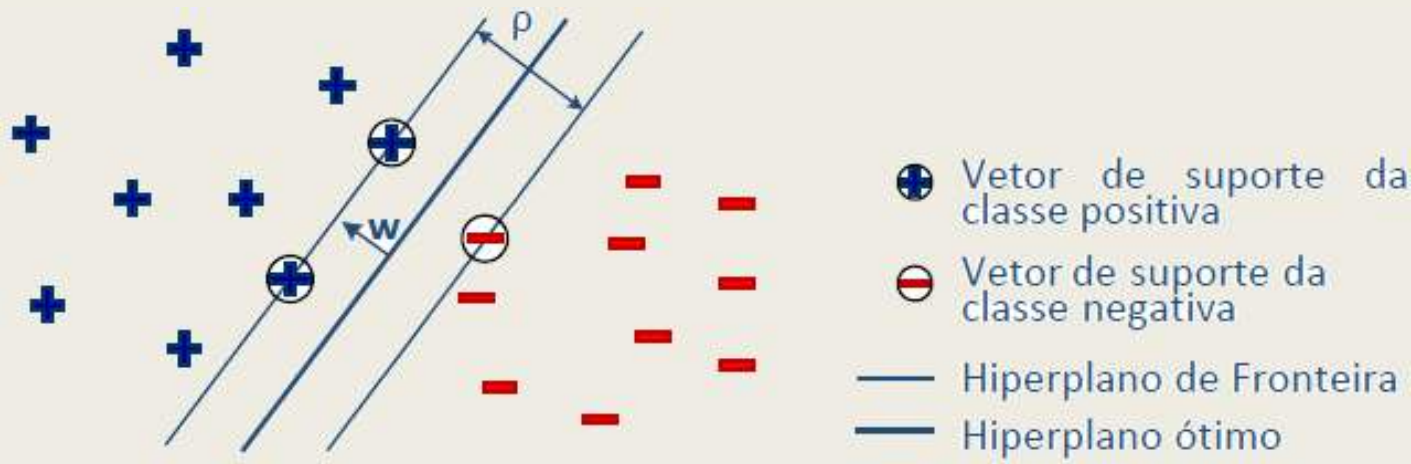
- With two classes, one negative and one positive, the goal of an SVM is to find the optimal hyperplane of separation between them.
  - *To this end, the training of SVMs involves solving a quadratic optimisation problem to maximise the margin of separation between the items of the two different classes.*
  - *As illustrated in the following image, through a representative scheme of two predictive variables, many hyperplanes can separate the point sets of the two classes. Still, it will be the one that allows the most significant margin of separation between the two classes and offers the greatest capacity for generalisation from the outset — optimal hyperplane.*



Possíveis planos de separação das classes positiva e negativa

Support Vectors

- The following image illustrates a geometric construction of the corresponding optimal hyperplane in a two-dimensional space.
  - *In this case, we have an optimal plan that guarantees the maximum separation at the expense of 3 training examples, 1 negative and 2 positive, commonly referred to as* ***support vectors****.*



Hiperplano ótimo de separação e respetivos vetores de suporte

SVM for Linear Problems

The first formulation of SVMs was developed to deal with linearly separable data.

- A linear classification, capable of dealing with this type of problem, can be represented by a linear function ***f(x)*** that, from the set of explanatory variables **x** (dimension array **d**), produces as a result:
  - *a value greater than zero, whenever the observed values of **X** are associated with a positive class (Agreed designation for one of the two categories of the classifier and symbolically represented +1)*
  - *a value less than zero, whenever it is associated with the negative class (−1).*
- This linear function can take the form:

$$f(\mathbf{x}) = \mathbf{w}^t \cdot \mathbf{x} + b = \sum_{i=1}^{d} w_i x_i + b$$

*where **w** is the array of weights (of dimension d) and **b** is the value of the bias, which, together, characterize the **optimal hyperplan**.*

(The array of weights **w** defines the direction perpendicular to the hyperplan, as illustrated in the image, and the parameter **b** is influenced by the displacement of the hyperplan towards one of the classes, moving parallel to itself)

Hard Margin SVM

- Considering that the data are linearly separable, a hard margin SVM can be used for classification. The **optimal hyperplane** is defined as

$$\mathbf{w}^t \cdot \mathbf{x} + b = 0$$

- Being the two classes of data completely separable by a hyperplane, It is then possible to find a pair (W, B) that guarantees the verification of the following 2 equations, for whatever the set of observations from the explanatory variables $\mathbf{x}_i$,

$$\mathbf{w}^t \cdot \mathbf{x}_i + b \geq +1, \text{ para } y_i = +1$$

$$\mathbf{w}^t \cdot \mathbf{x}_i + b \leq -1, \text{ para } y_i = -1$$

or the verification of the equivalent combined restriction

$$y_i(\mathbf{w}^t \cdot \mathbf{x}_i + b) \geq 1, \forall i = 1, \ldots, n$$

*Where **n** is the number of observations in the training set.*

Hard Margin SVM

- The linear classifiers that separate the training set into two groups have a **positive margin** (which we represent by **ρ**)
  - *That is, they ensure that there is **no example of training** between the frontier hyperplans (**w**^{t} · **x** + *b* = +1) and (**w**^{t} · **x** + *b* = −1)*
  - *It's due to this **exclusion margin** (from training examples), this type of classifier is called **Hard Margin SVM.***
- **The training of the SVM aims to maximize the separation margin ρ**
  - *It is possible to demonstrate that this goal is achieved by minimizing the norm of the weight array **w**. This way, the training involves the following optimization problem:*

$$\underset{\mathbf{w},b}{\text{Minimizar}} \ \|\mathbf{w}\|^2 ,$$

$$\text{sob as restrições: } y_i(\mathbf{w}^t \cdot \mathbf{x}_i + b) \geq 1, \forall i = 1, \ldots, n$$

- Once we've found the optimal match (w*, b*) during the training phase, the Linear Function f(**x**) = **w**^{t} · **x** + *b* will be used to classify any example $\mathbf{z}_j$ of the test set

$$y_j = \begin{cases} +1, & \text{se } \mathbf{w}^{*t} \cdot \mathbf{z}_j + b^* \geq 0 \\ -1, & \text{se } \mathbf{w}^{*t} \cdot \mathbf{z}_j + b^* < 0 \end{cases}$$

Soft Margin SVM

Unfortunately, in the overwhelming majority of classification problems, the data is not linearly separable

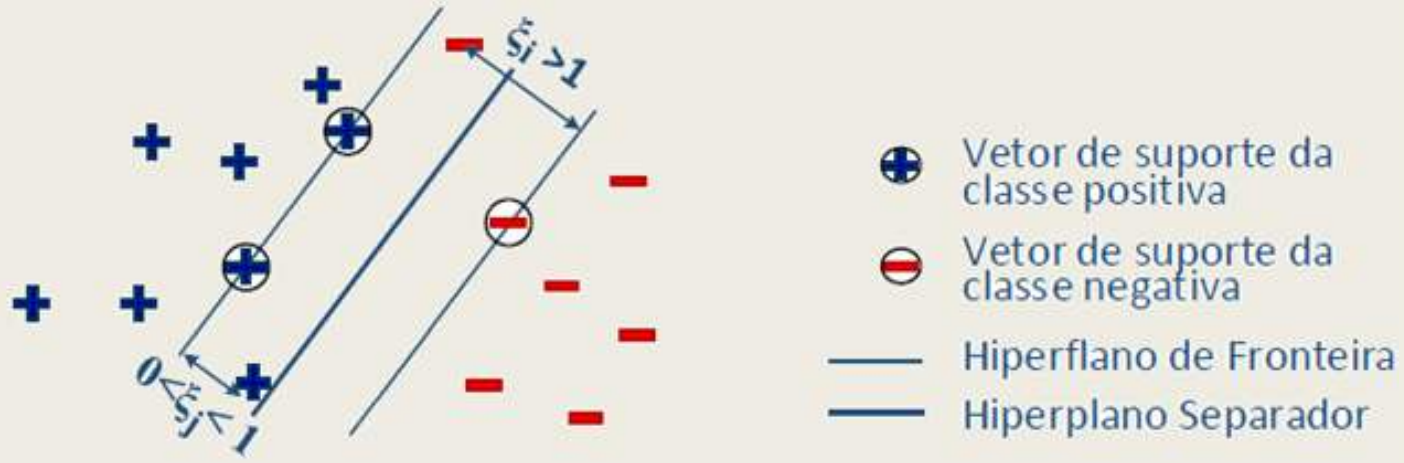
- Linear SVMs are unable to handle this more general training data.
- However, they can be adapted to be able to deal with these types of problems as well.
  - *That's what happens with Soft Edge SVMs, where some examples of the training data are allowed to stay within that margin or even to be on the wrong side of the separation hyperplan.*
    - This is achieved by introducing tolerance variables ξ_i:

$$y_i(\mathbf{w}^t \cdot \mathbf{x}_i + b) \geq 1 - \xi_i, \;\; \xi_i \geq 0, \;\; \forall i = 1, \ldots, n$$

  - *Even so, Soft Edge SVMs will tolerate that some of the training examples go against the exclusion margin constraint, they will naturally try to minimize their occurrence.*

Soft Margin SVM

- Each variable ξ_i represents the **distance from the "badly behaved" example** to the boundary hyperplan of its class



Posicionamento de exemplos de treino numa SVM de margem suave

- *If its value is between 0 and 1, it means that the example is positioned within the "exclusion" margin,*
- *But if it exceeds the unit, it will be a classification error since the training example will already be on the wrong side of the separation hyperplan.*

Soft Margin SVM

- In order to accommodate the possibility that there may be some misclassification in the training data, The quadratic optimization problem takes into account, the sum of all deviations ξ_i, is now formulated as follows[1]:

$$\underset{\mathbf{w},b}{\text{Minimizar}} \ \frac{1}{2}\|\mathbf{w}\|^2 + C\left(\sum_{i=1}^{n}\xi_i\right),$$

$$\text{sob as restrições: } y_i(\mathbf{w}^t \cdot \mathbf{x}_i + b) \geq 1 - \xi_i, \ \ \xi_i \geq 0, \ \ \forall i = 1, \ldots, n$$

where parameter C is a regularization term that assigns a weight to the minimization of classification errors in relation to another objective of optimization: that one of maximizing the separation margin.

In other words, this parameter C (also called the 'penalty error parameter') allows the analyst to control the relative importance of each of these two objectives

- Minimization of errors
- Margin Maximization

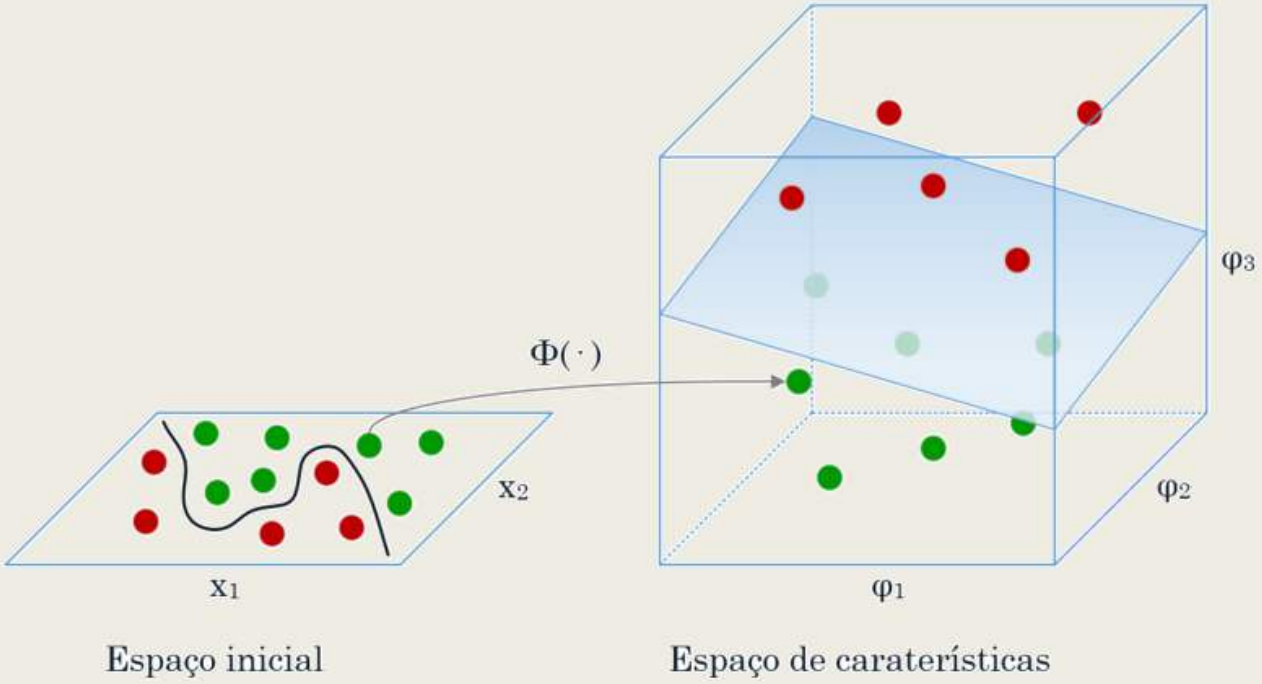
With this, it also controls, in some way, the generalization capacity of the classifier

As for small values of C the margins tend to be larger, it is to be expected that the classifier will be more capable of generalization, although in the training phase it allows a greater number of classification errors.

[1] *Quanto à forma de resolver este problema de otimização, refira-se apenas que, à semelhança das SVM de Margem Rígida, passa pela utilização de multiplicadores de Lagrange.*

Nonlinear SVM

- A large part of real-world problems involve data for which there is no separating hyperplan, because they have inherently nonlinear structures.
  - *Even though soft-margin SVMs can mitigate this difficulty in part, they can't handle datasets that have a highly nonlinear distribution well.*
  - *Fortunately, an attractive feature of SVMs is that they can easily be transformed into non-linear learning mechanisms.*
    - To do this, the input instances are usually mapped to a larger space, called a feature space, where it will be possible to define hyperplans that linearly separate them.



Kernel Functions

- In general, the transformations responsible for mapping the training sets, from their initial space to a new space of a higher dimension, which makes the observations linearly separable into two classes, can be of great complexity or even unfeasible.
- SVMs get around this difficulty by perceiving 2 things:
  - *that the only operation that needs to be performed in the characteristic space* ***is the calculation of internal products***
  - *and that for certain mappings, these internal products can be easily accomplished through known functions, called* ***kernel functions***
    - Representing by Φ(·) the transformation responsible for one of these mappings, the internal product between any two arrays xi and xj, once mapped, can be obtained by applying the Kernel function K(·) directly to these two arrays of the initial space

$$\Phi(\mathbf{x_i})^t \cdot \Phi(\mathbf{x}_j) = K(\mathbf{x_i}, \mathbf{x}_j)$$

Typical Kernel Functions

In the implementation of nonlinear SVMs, we usually chose a mapping to which a known kernel function is associated

- Among the most commonly used kernel functions are
  - *the Polynomial*

$$K_{\text{Polinomial}}(\mathbf{x_i}, \mathbf{x}_j) = \left(\gamma\, \mathbf{x_i}^t \cdot \mathbf{x}_j + k\right)^d$$

  - *A Radial Base (RBF – radial basis function)*

$$K_{\text{RBF}}(\mathbf{x_i}, \mathbf{x}_j) = \exp\left(-\gamma \left\|\mathbf{x_i} - \mathbf{x}_j\right\|^2\right)$$

  - *and Sigmoidal*

$$K_{\text{Sigmoidal}}(\mathbf{x_i}, \mathbf{x}_j) = \tanh\left(\gamma\, \mathbf{x_i}^t \cdot \mathbf{x}_j + k\right)$$

- The gamma (γ) is another important parameter for tuning SVM,
  - *in addition to the C regularization term and the kernel function itself to be used.*
  - *But it is only considered in the case of not using the linear kernel.*
  - *Defines the scope of influence of each training example. The higher the γ, the less the reach (and the greater the overfitting).*

Advantages and Disadvantages of SVM

- Advantages
  - *They have characteristics that favor their robustness and good predictive performance compared to other classifier algorithms*
  - *They have a good tolerance to noise,*
  - *The convexity of the optimization problem formulated for his training (which implies the existence of a single global minimum)*
  - *the accuracy does not depend on the size and dimensionality of the data*
  - *Good generalization ability*
- Disadvantages
  - *Not always easy to set up*
  - *Created a model that is difficult to understand (seen as a black box technique)*
  - *can't handle categorical attributes (requires the application of the "one hot encoding" technique)*
  - *predictors with different scales compromise their performance (requires normalization)*
  - *involves a high computational effort*

Solve exercise #19, #20

from the book of exercises