Máquina de Vetores de Suporte

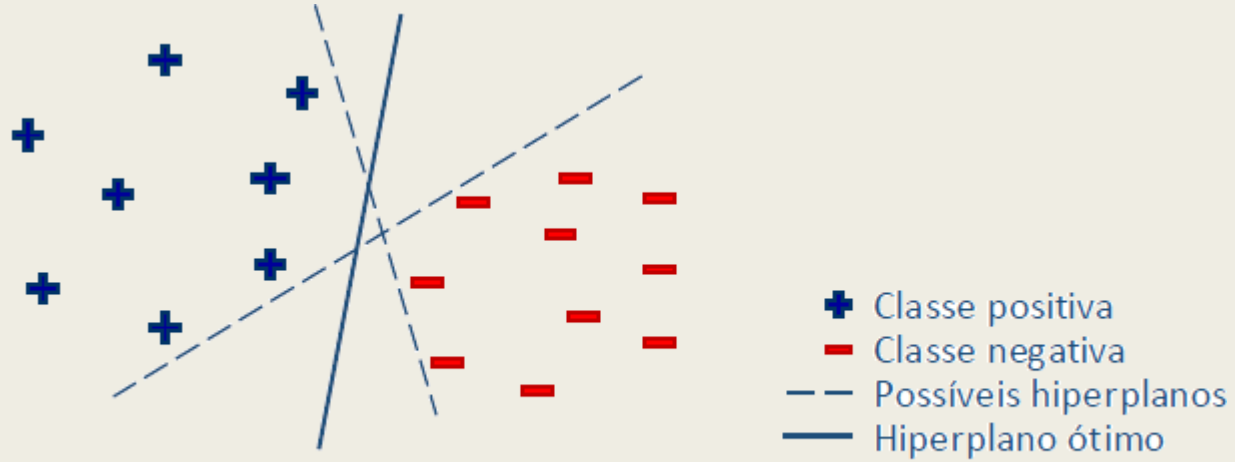
- Uma máquina de vetores de suporte (Support Vector Machine – SVM) é um algoritmo de Machine Learning desenvolvido em 1995 por Vladimir Vapnik[1], inicialmente para classificação binária.
  - *Embora atualmente as SVM possam também ser usadas para problemas de regressão (prever o valor de uma variável contínua em vez de classificar) e para solucionar problemas multiclasse, cingimo-nos somente à sua formulação original, a qual já abrange uma categoria muito importante dos problemas de previsão.*

- Trata-se de mais um algoritmo bastante popular entre a comunidade de ML, quer para classificação, quer para regressão
- Tentemos então perceber o seu funcionamento base na classificação binária…

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

Hiperplano ótimo

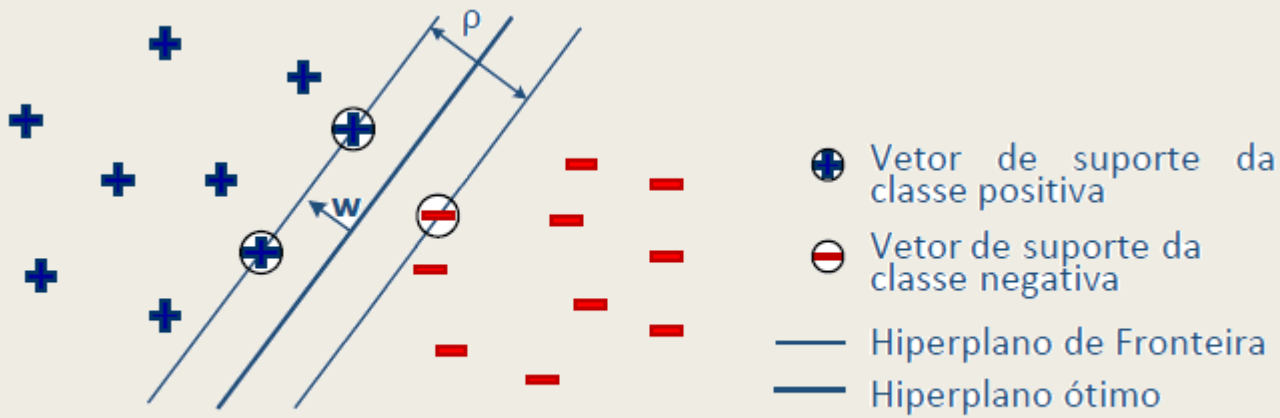
- Perante duas classes, uma negativa e outra positiva, o objetivo de uma SVM é encontrar o hiperplano ótimo de separação entre elas.
  - *Para o efeito, o treino das SVM envolve a resolução de um problema de otimização quadrática, formulado com o objetivo de maximizar a margem de separação entre os itens das duas diferentes classes.*
  - *Como se ilustra na figura, através dum esquema representativo dum espaço de entrada de duas dimensões (duas variáveis preditivas), existem muitos hiperplanos que podem separar os conjuntos de pontos das duas classes, mas será aquele que possibilitar a maior margem de separação entre as duas classes que à partida oferecerá maior capacidade de generalização — hiperplano ótimo.*



Possíveis planos de separação das classes positiva e negativa

Vetores de Suporte

- A figura ilustra uma construção geométrica do correspondente hiperplano ótimo num espaço de duas dimensões.
  - *Neste caso temos um plano ótimo que garante a máxima separação à custa de 3 exemplos de treino, 1 negativo e 2 positivos, normalmente designados por vetores de suporte (VS).*



Hiperplano ótimo de separação e respetivos vetores de suporte

- Os VS são os pontos (vetores) que determinam a posição e a orientação do hiperplano de decisão
  - *são os pontos mais próximos da fronteira de decisão entre as classes,*
  - *a quantidade de VS depende do dataset e dos hiperparâmetros escolhidos.*

SVM para problemas lineares

A primeira formulação das SVM foi desenvolvida para lidar com dados linearmente separáveis.

- Uma classificação linear, capaz de lidar com esse tipo de problemas, pode ser representado por uma função linear f(**x**) que, a partir do conjunto de variáveis explicativas **x** (vetor de dimensão *d*), produza como resultado:
  - *um valor maior que zero, sempre que aos valores observados de* **x** *esteja associada a classe positiva (designação convencionada para uma das duas categorias do classificador e simbolicamente representada +1)*
  - *e um valor menor que zero, sempre que, pelo contrário, esteja associada a classe negativa (−1).*
- Essa função linear pode tomar a forma:

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

em que **w** *é o vetor de pesos (de dimensão d) e b o valor do enviesamento (bias), os quais, em conjunto, caracterizam o hiperplano ótimo.*

(O vetor de pesos **w** define a direção perpendicular ao hiperplano, tal como ilustrado na figura, e o parâmetro *b* tem como influência o deslocamento do hiperplano em direção a uma das classes, movendo-se paralelamente a si próprio)

SVM de margem rígida

- Considerando que os dados são linearmente separáveis pode usar-se uma SVM de margem rígida para a classificação. O hiperplano ótimo é definido como

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

- Sendo as duas classes de dados completamente separáveis por um hiperplano, é então possível encontrar um par (**w**, *b*) que garanta a verificação das 2 inequações que se seguem, para qualquer que seja o conjunto de observações das variáveis explicativas $\mathbf{x}_i$,

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

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

ou a verificação da restrição combinada equivalente

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

em que n é o número de observações do conjunto de treino.

SVM de margem rígida

- Os classificadores lineares que separam em dois grupos o conjunto de treino **possuem margem positiva** (que representamos por ρ)
  - *ou seja,* ***garantem que não há nenhum exemplo de treino*** *entre os hiperplanos de fronteira* ($\mathbf{w}^t \cdot \mathbf{x} + b = +1$) *e* ($\mathbf{w}^t \cdot \mathbf{x} + b = -1$)
  - *É devido à existência desta* ***margem de exclusão*** *(de exemplos de treino) que este tipo de classificador se designa por SVM de* ***Margem Rígida****.*
- **O treinamento das SVM tem como objetivo a maximização da margem de separação ρ**
  - *é possível demonstrar que se atinge esse mesmo objetivo minimizando a norma do vetor de pesos* **w***. Dessa forma, o treinamento envolve o seguinte problema de otimização:*

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

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

- Depois de encontrado o par ótimo (w^*, b^*) durante a fase de treino, a função linear $f(\mathbf{x}) = \mathbf{w}^t \cdot \mathbf{x} + b$ será usada para classificar um qualquer exemplo $\mathbf{z}_j$ do conjunto de teste

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

SVM de margem suave

Infelizmente, na esmagadora maioria dos problemas de classificação, os dados não são linearmente separáveis

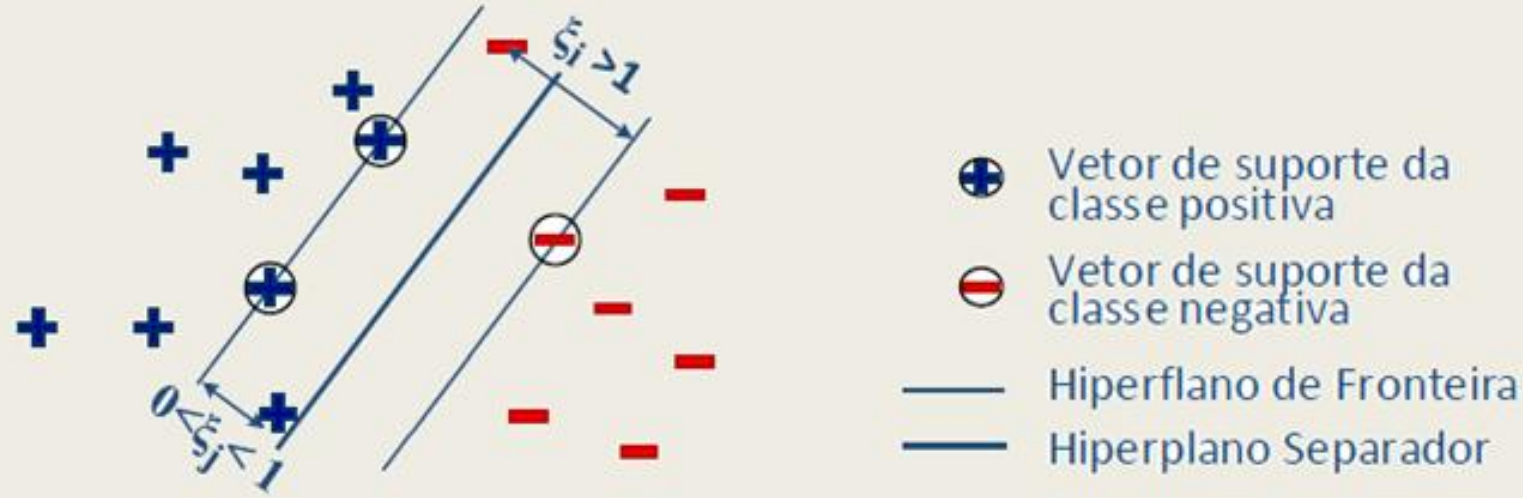
- As SVM lineares são incapazes de lidar com esses dados de treino mais gerais.
- No entanto, podem ser adaptadas de forma a conseguirem lidar também com esse tipo de problemas.
  - *É o que acontece com as SVM de Margem Suave, em que se permite que alguns exemplos dos dados de treino possam ficar dentro dessa margem ou mesmo ficarem do lado errado do hiperplano de separação.*
    - Isso é conseguido com a introdução de variáveis de tolerância ξ_i:

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

  - *Ainda que as SVM de Margem Suave tolerem que alguns dos exemplos de treino violem a restrição da margem de exclusão, tentarão naturalmente minimizar a sua ocorrência.*

SVM de margem suave

- Cada variável ξ_i representa a **distância do exemplo "mal comportado"** ao hiperplano de fronteira da sua classe



Posicionamento de exemplos de treino numa SVM de margem suave

- *se o seu valor se situar entre 0 e 1, significa que o exemplo se encontra posicionado dentro da margem de "exclusão",*
- *mas caso supere a unidade, tratar-se-á de um erro de classificação, uma vez que o exemplo de treino estará já do lado errado do hiperplano de separação.*

SVM de margem suave

- De forma a acomodar a possibilidade de existirem alguns erros de classificação nos dados de treino, o problema de otimização quadrática leva em consideração, como novo termo a minimizar, a soma de todos os desvios ξ_i, passando a ser formulado da seguinte forma[1]:

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

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

onde o parâmetro C é um termo de regularização que atribui um peso à minimização dos erros de classificação em relação ao outro objetivo da otimização: o da maximização da margem de separação.

Ou seja, este parâmetro C (também designado 'parâmetro de penalização do erro') permite ao analista controlar a importância relativa que cada um desses dois objetivos

- minimização dos erros
- maximização da margem

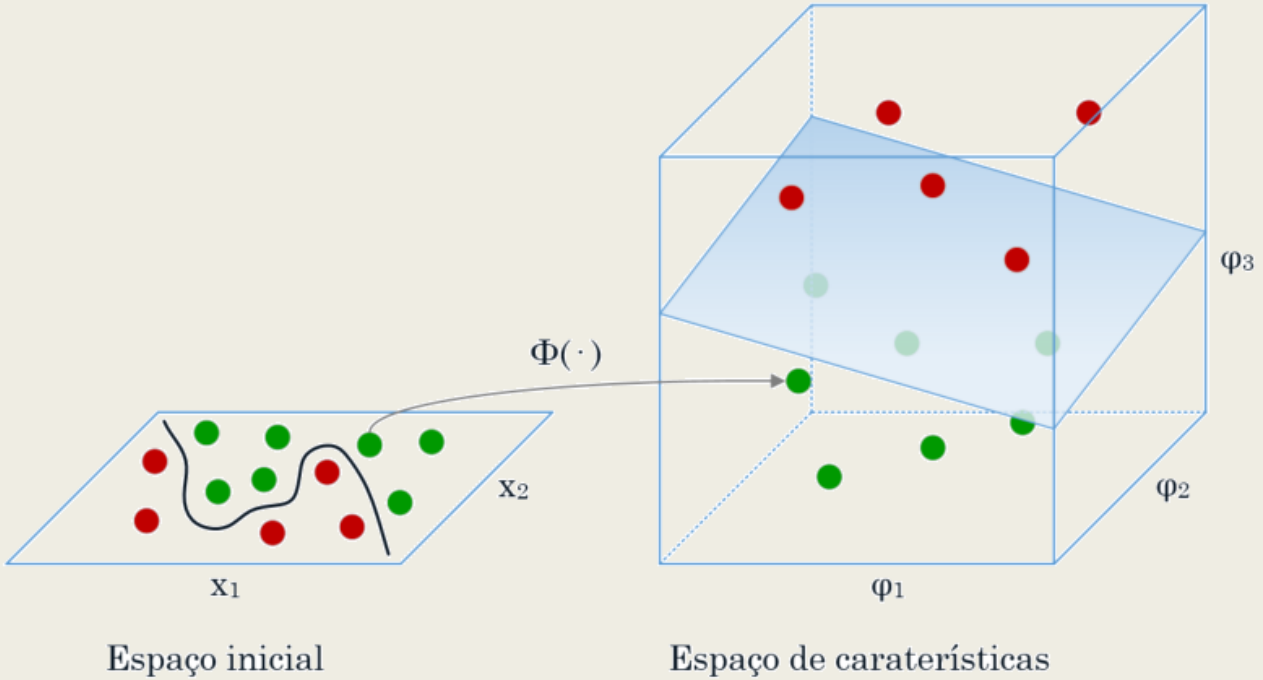
Com isso, controla também, de alguma forma, a capacidade de generalização do classificador

Como para valores pequenos de *C* as margens são tendencialmente maiores, será de esperar uma maior capacidade de generalização por parte do classificador, ainda que na fase de treino permita um maior número de erros de classificação.

[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.*

SVM não lineares

- Uma grande parte dos problemas do mundo real envolve dados para os quais não existe um hiperplano separador, por apresentarem estruturas inerentemente não lineares.
  - *Ainda que as SVM de margem suave consigam mitigar em parte essa dificuldade, não conseguem lidar bem com conjuntos de dados que tenham uma distribuição claramente não linear.*
  - *Felizmente, uma característica atrativa das SVM é que podem facilmente ser transformadas em mecanismos de aprendizagem não linear.*
    - Para o efeito, as instâncias de entrada são normalmente mapeadas para um espaço de maior dimensão, designado espaço de características, onde já será possível definir hiperplanos que as separe linearmente.



Funções de Kernel

- Em geral, as transformações responsáveis pelo mapeamento dos conjuntos de treino, do seu espaço inicial para um novo espaço de dimensão em geral mais elevada, que torne as observações linearmente separáveis em duas classes, podem ser de grande complexidade ou até mesmo inviáveis.
- As SVM contornam esta dificuldade ao perceberem 2 coisas:
  - *que a única operação que é necessário realizar no espaço de características é o cálculo de produtos internos*
  - *e que para determinados mapeamentos, esses produtos internos podem ser facilmente realizados através de funções conhecidas, designadas funções de Kernel*
    - Representando por $\Phi(\cdot)$ a transformação responsável por um desses mapeamentos, o produto interno entre quaisquer dois vetores x_i e x_j, depois de mapeados, pode ser obtido aplicando a função Kernel $K(\cdot)$ diretamente a esses dois vetores do espaço inicial

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

Funções de Kernel Típicas

Na implementação das SVM não lineares opta-se, normalmente, por um mapeamento a que esteja associada uma função de Kernel conhecida

- Entre as funções kernel mais usadas, encontram-se
  - *a Polinomial*

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

  - *a de base radial (RBF – radial basis function)*

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

  - *e a Sigmoidal*

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

- O gamma (γ) é mais um importante parâmetro de afinação das SVM,
  - *para além do já mencionado termo de regularização C e da própria função de Kernel a usar.*
  - *Mas só se considera no caso de não se usar o Kernel linear.*
  - *Define o alcance da influência de cada exemplo de treino. Quanto maior for o γ, menor será o alcance (e maior o overfitting).*
  - *O γ, dependendo do dataset e natureza do problema, pode variar, por exemplo, entre 0.001 e 10.*

Vantagens e Desvantagens das SVM

- Vantagens
  - *lidam bem com espaços de alta dimensionalidade (muitas features);*
  - *são eficazes mesmo em datasets de pequena dimensão (poucas observações);*
  - *sabem lidar com problemas não-lineares (com recurso às funções de kernel);*
  - *regularização embutida, através dum parâmetro de penalização incorporado que ajuda a evitar o overfitting (o termo de regularização).*
- Desvantagens
  - *nem sempre fáceis de configurar;*
  - *modelos criados de difícil interpretação (vistas como uma técnica de caixa-preta);*
  - *não conseguem lidar com atributos categóricos (requerem a aplicação da técnica "one hot encoding");*
  - *preditores com diferentes escalas compromete o seu desempenho (requerem normalização);*
  - *dificuldade em lidar com datasets de grande dimensão, devido à complexidade quadrática do seu algoritmo de otimização;*
  - *desempenho dependente do balanceamento dos datasets.*