Máquinas de Vetores de Suporte

Lorenzo Moreira Donatti

Introdução

- Abordagem Geral
- Vantagens e Desvantagens
- Aplicabilidade da técnica
- Fundamentação teórica
- Fundamentação matemática
- Problema de Otimização
- Soft SVM
- Kernels
- SVMs Multiclasses
- Aplicação prática (Código)

Máquinas de Vetores de Suporte

- Fundamentado por Vladimir Vapnik em 1979.
- Larga aplicabilidade na resolução de problemas.
- Matemática complexa e fundamentada por trás.
- Pode resolver problemas em domínios que outras técnicas não englobam com robustez.
- É um modelo extremamente complexo e que causa dificuldade e estranheza em aprendizes.
- A má configuração dos Hiperparâmetros pode acabar com a busca pelo melhor hiperplano.

O que são SVMs?

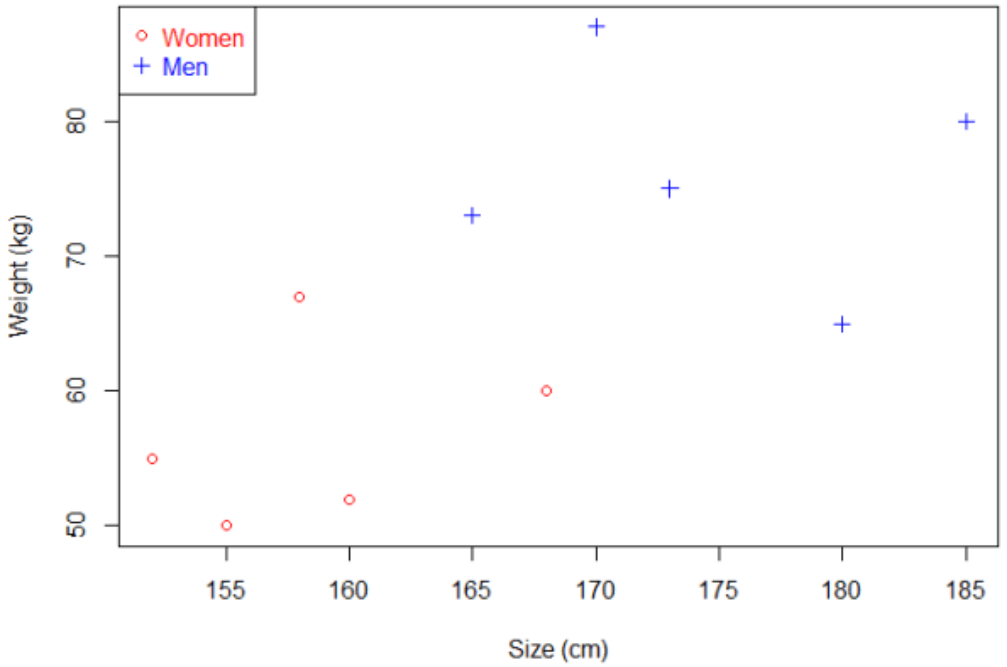
- Máquinas de Vetores de Suporte (SVM) é um algoritmo de aprendizado de máquina supervisionado, não probabilístico, utilizado tanto para problemas de classificação, tanto para regressão. Apesar de ser mais utilizado em problemas de classificação.

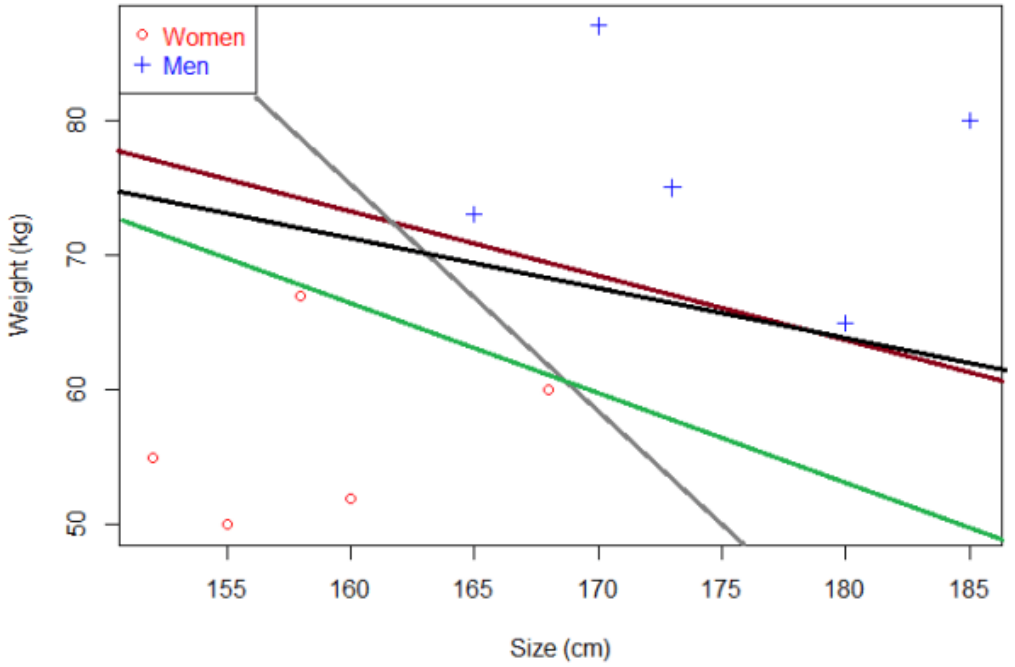
- Possui duas principais vantagens se comparado a algoritmos recentes de aprendizado de máquina: Maior velocidade e melhor performance quando trabalhado com dados limitados.

Princípios Básicos

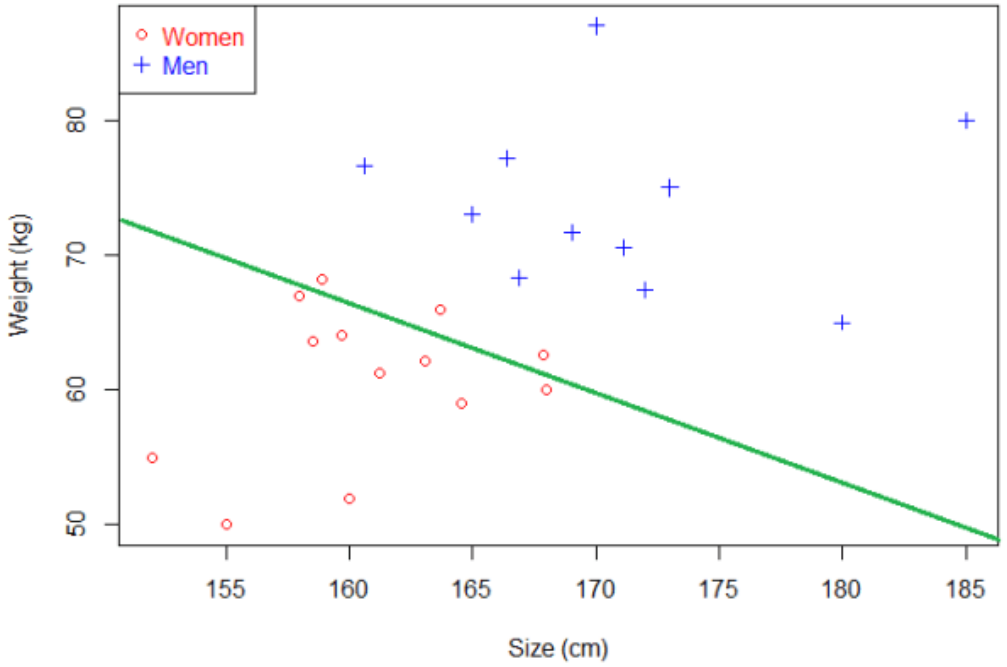
- O objetivo de um SVM é encontrar o hiperplano de separação linear ideal o qual maximiza a margem da base de treinamento.
- SVMs possuem a capacidade de trabalhar com problemas que envolvam conjunto de dados não-linearmente separáveis e com problemas multiclasse.
- Para melhor entendimento, será apresentado um exemplo de classificação binária linearmente separável.

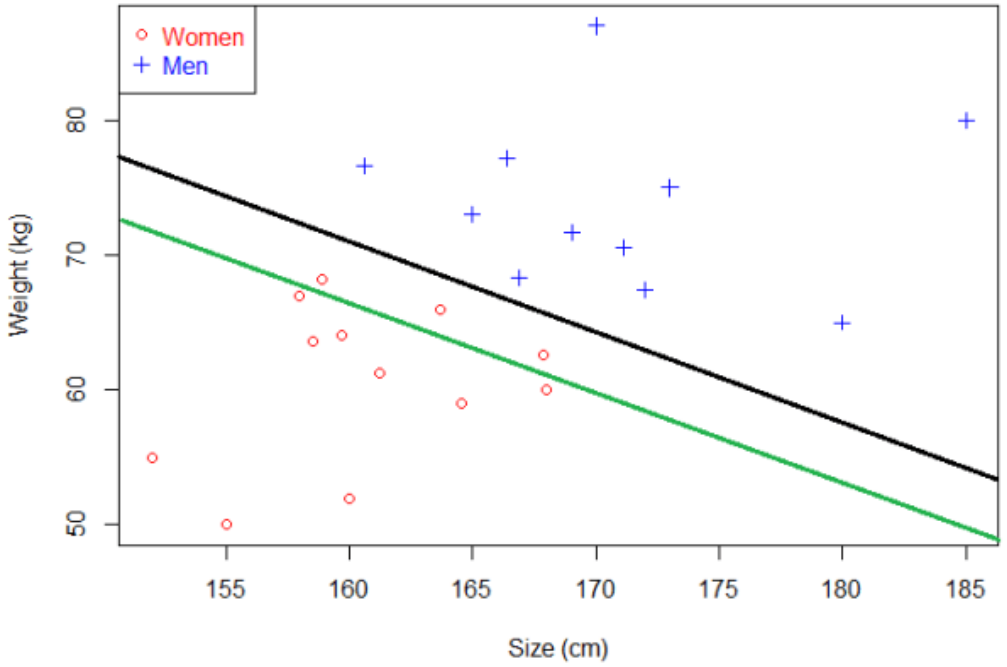
Hiperplano


Women
Men
Weight (kg)
50
60
70
80
155
160
165
170
175
180
185
Size (cm)

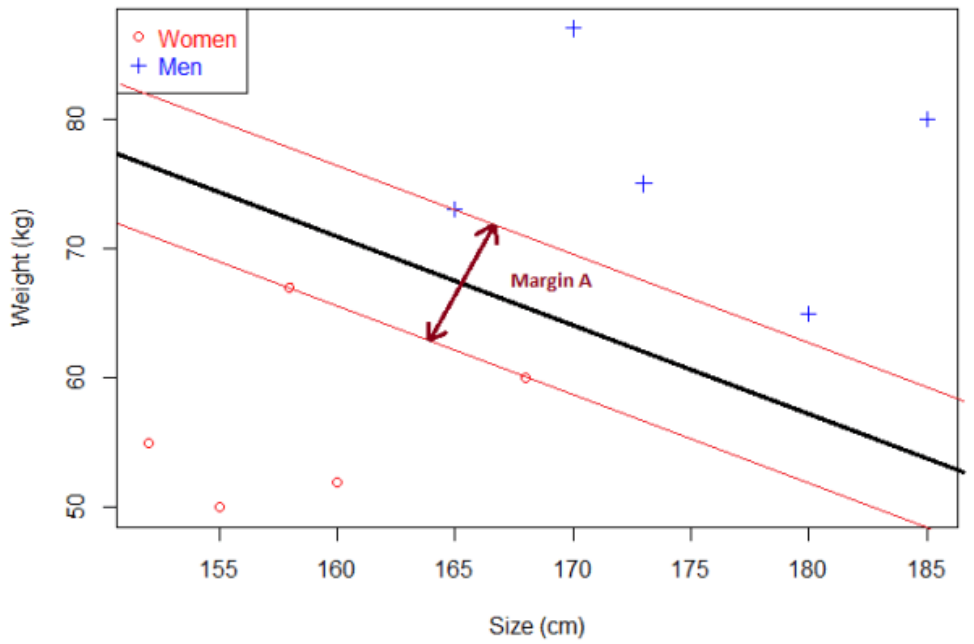

Women
Men
Weight (kg)
50
60
70
80
155
160
165
170
175
180
185
Size (cm)

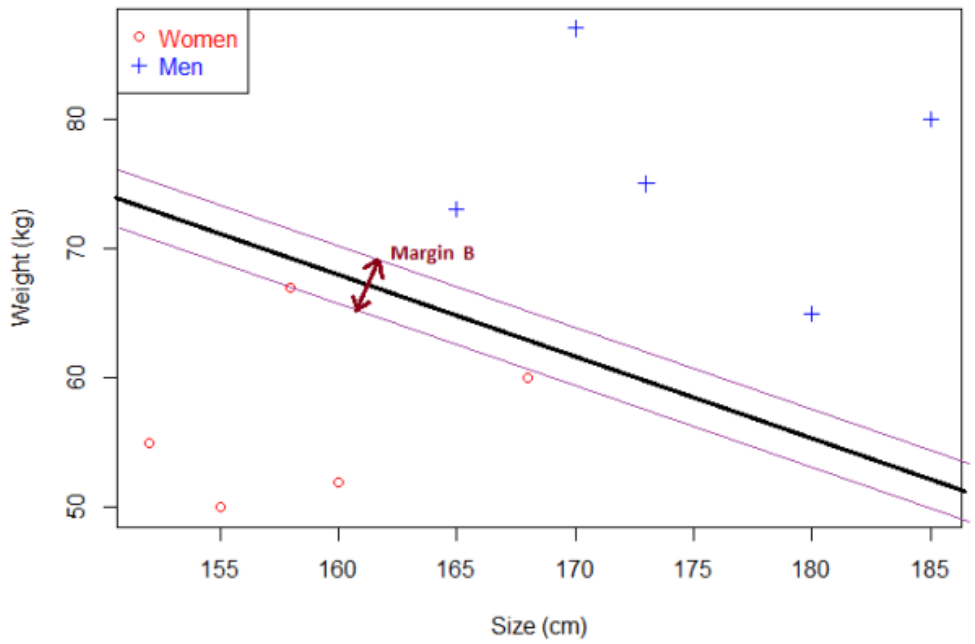
Diferenças entre diferentes Hiperplanos





Vetores de suporte e Margens



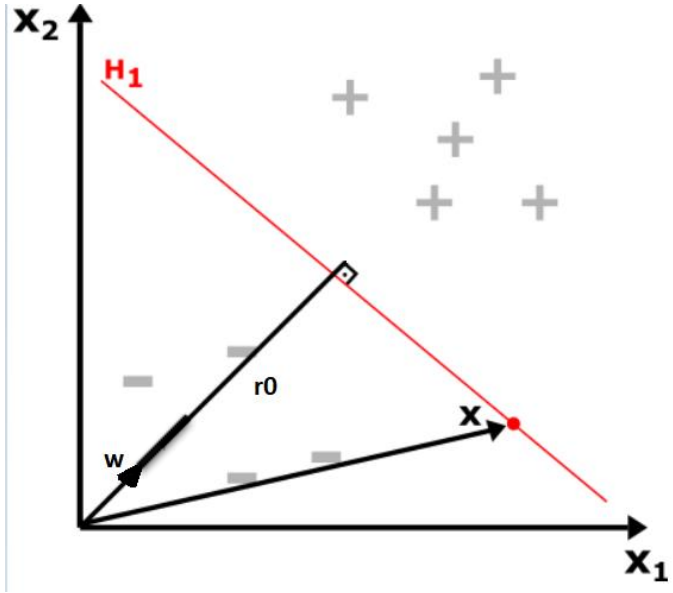


Matemática por trás de SVMs

- De onde surge a equação do hiperplano?

Pelo produto escalar: $\bar{w}^{\mathrm{T}}\,\bar{xo} = ro * \|w\|$

Por conveniência, considerar: $ro * \|w\| = -b$



Então: $\bar{w}^{\mathrm{T}}\,\bar{xo} + b = 0$

$\bar{w} \cdot \bar{xo} + b > 0 \rightarrow$ Pertence ao grupo "+"

$\bar{w} \cdot \bar{xo} + b < 0 \rightarrow$ Pertence ao grupo "-"

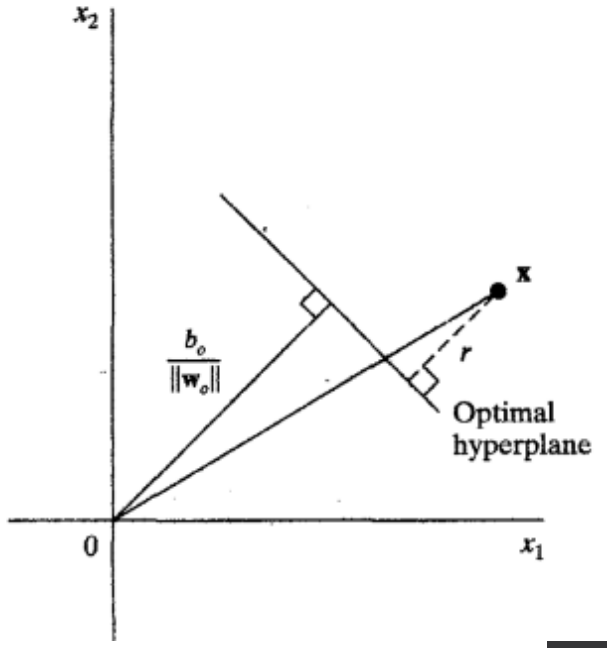
Matemática por trás de SVMs

- A partir dessa dedução, é possível afir $g(\mathbf{x}) = \mathbf{w}_o^T\mathbf{x} + b_o = r\|\mathbf{w}_o\|$

$$r = \frac{g(\mathbf{x})}{\|\mathbf{w}_o\|}$$

- Definind como:

$$\mathbf{w}_o^T\mathbf{x}_i + b_o \geq 1 \quad \text{for } d_i = +1$$
$$\mathbf{w}_o^T\mathbf{x}_i + b_o \leq -1 \quad \text{for } d_i = -1$$



- A distância algébrica se

$$r = \frac{g(\mathbf{x}^{(s)})}{\|\mathbf{w}_o\|}$$

$$= \begin{cases} \frac{1}{\|\mathbf{w}_o\|} & \text{if } d^{(s)} = +1 \\ -\frac{1}{\|\mathbf{w}_o\|} & \text{if } d^{(s)} = -1 \end{cases}$$

Problema de Otimização

- Então chegamos no valor de:

$$\rho = 2r$$

$$= \frac{2}{\|\mathbf{w}_o\|}$$

- Onde p é o valor ótimo da margem, que deve ser resultado da maximização, através da minimização do valor do módulo de **w**, chegando em um problema de otimização, onde uma das melhores soluções é pelo método de multiplicadores de Lagrange.

$$d_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 \qquad \text{for } i = 1, 2, \ldots, N$$

minimizes the cost function:

$$\Phi(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{w}$$

Problema de Otimização

$$Q(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T \mathbf{x}_j$$

$$\mathbf{w}_o = \sum_{i=1}^{N} \alpha_{o,i} d_i \mathbf{x}_i$$

subject to the constraints

(1) $\sum_{i=1}^{N} \alpha_i d_i = 0$

(2) $\alpha_i \geq 0 \qquad \text{for } i = 1, 2, \ldots, N$

$$b_o = 1 - \mathbf{w}_o^T \mathbf{x}^{(s)} \qquad \text{for } d^{(s)} = 1$$

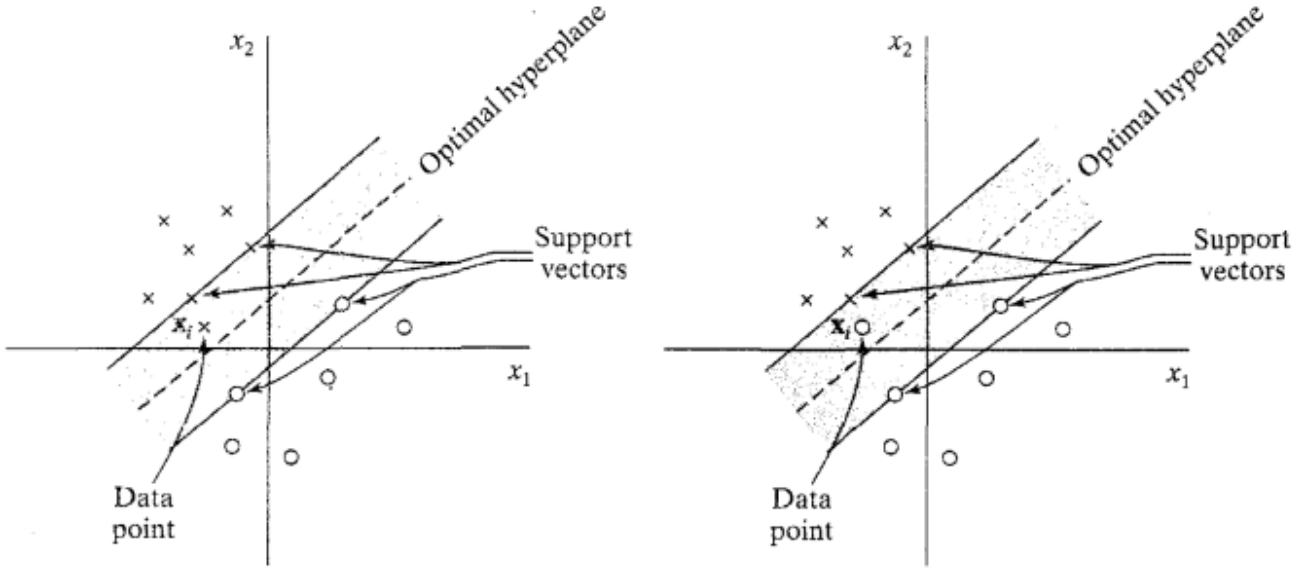
Soft SVM

- Baseado na ideia de que em dados reais, o classificador necessita ser mais flexível, admitindo algumas "invasões" na margem e no hiperplano, foi criado o Soft SVM, que não garante a divisão perfeita das classes.
- A penalização pela invasão é proporcional ao seu tipo, sendo maior a aqueles dados que invadiram o hiperplano do que naqueles que apenas invadiram a margem.

Soft SVM

- É possível notar que uma nova variável, que se refere a uma "taxa de perda", responsável por representar o quão longe o dado está de onde deveria estar.
- Se estiver entre 0 e 1, violou apenas a margem, se for maior que 1, violou o hiperplano.

$$d_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, \qquad i = 1, 2, \ldots, N$$



Soft SVM

- Analisando a função de custo, nota-se a criação de um hiperparâmetro C, que funciona como um parâmetro de regularização, evitando o overfitting, ele controla o impacto que a soma dos erros terão. Para maiores valores de C, mais vetores de suporte existirão, se assemelhando a um Hard SVM.

$$\Phi(\mathbf{w}, \xi) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{N}\xi_i$$

Soft SVM

- Além disso, algumas diferenças no problema de otimização são notadas, como:

$$Q(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T \mathbf{x}_j$$

subject to the constraints

$$(1) \quad \sum_{i=1}^{N} \alpha_i d_i = 0$$

$$(2) \quad 0 \leq \alpha_i \leq C \qquad \text{for } i = 1, 2, \ldots, N$$

where C is a user-specified positive parameter.

$$\mathbf{w}_o = \sum_{i=1}^{N_s} \alpha_{o,i} d_i \mathbf{x}_i$$

$$\alpha_i \left[d_i (\mathbf{w}^T \mathbf{x}_i + b) - 1 + \xi_i \right] = 0, \qquad i = 1, 2, \ldots, N$$

$$\mu_i \xi_i = 0, \qquad i = 1, 2, \ldots, N$$

$$\xi_i = 0 \quad \text{if} \quad \alpha_i < C$$

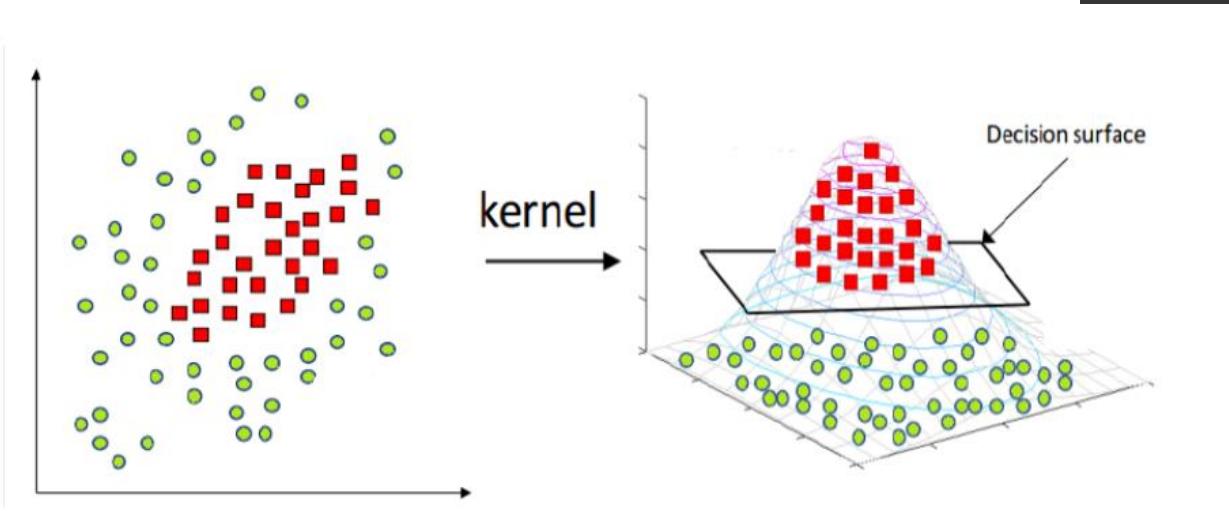
Truque do Kernel

- Em diversas aplicações reais, dados podem ser não-linearmente separáveis na dimensão trabalhada.
- Para isso, seguindo o teorema de Cover, é possível transformar os dados em linearmente separáveis em dimensões maiores.

- **Teorema de Cover**: Seguindo duas condições, é possível com alta probabilidade, transformar um conjunto de dados de treinamento que não é separável por um classificador linear, em um conjunto que é linearmente separável projetando-o em um espaço dimensional superior por meio de uma transformação não linear.
- Condição 1 – A transformação deve ser não linear.
- Condição 2 – O espaço transformado deve ter uma dimensão suficientemente alta.

Truque do Kernel

- Porém, ficar aumentando as dimensões seria inviável computacionalmente.
- O truque do Kernel consiste no uso de uma função Kernel que permite operar no espaço original, abstraindo as coordenadas dos dados em um espaço dimensional superior, computando apenas o produto interno entre os pares.



Kernels Mais utilizados

- Kernel Linear:

  K(x,y) = x.y (usado quando os dados são linearmente separáveis)

- Kernel de função de base radial (RBF)

$$K(\mathbf{x}, \mathbf{x}') = \exp(-\gamma \|\mathbf{x} - \mathbf{x}'\|^2)$$

- Kernel polinomial

$$K(x, y) = tanh(\gamma . x^T y + r)^d, \gamma > 0$$

- Kernel Sigmoide

$$K(x, y) = tanh(\gamma . x^T y + r)$$

Parâmetro Gamma

- Influencia na distância na qual as amostras serão consideradas para o cálculo da fronteira de decisão. São pesos que são introduzidos para a distância entre amostras, dando maior ou menor importância à amostras distantes ou próximas da fronteira de decisão.
- Aumentar seu valor faz com que os pontos mais distantes da região de separação entre classes sejam desconsiderados, levando a fronteiras de decisão mais complexas e *overfitting*.
- Valores menores aumentam a influência dos pontos mais distantes, permitindo alguns erros de classificação e podendo levar ao *underfitting*.

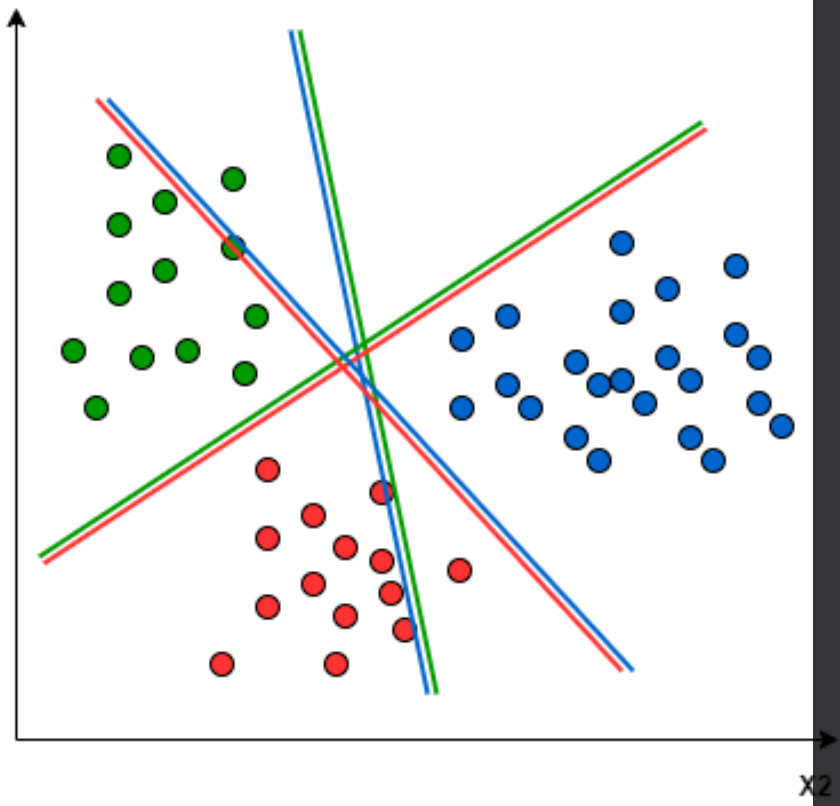
SVMs Multiclasse

- Em seu projeto original, SVMs não foram feitas para suportar problemas Multiclasse, porém, existem técnicas que conseguem reduzir um problema de várias classes em vários problemas binários.

- Serão abordadas três principais técnicas:
- One versus One
- One versus All
- Grafo acíclico dirigido

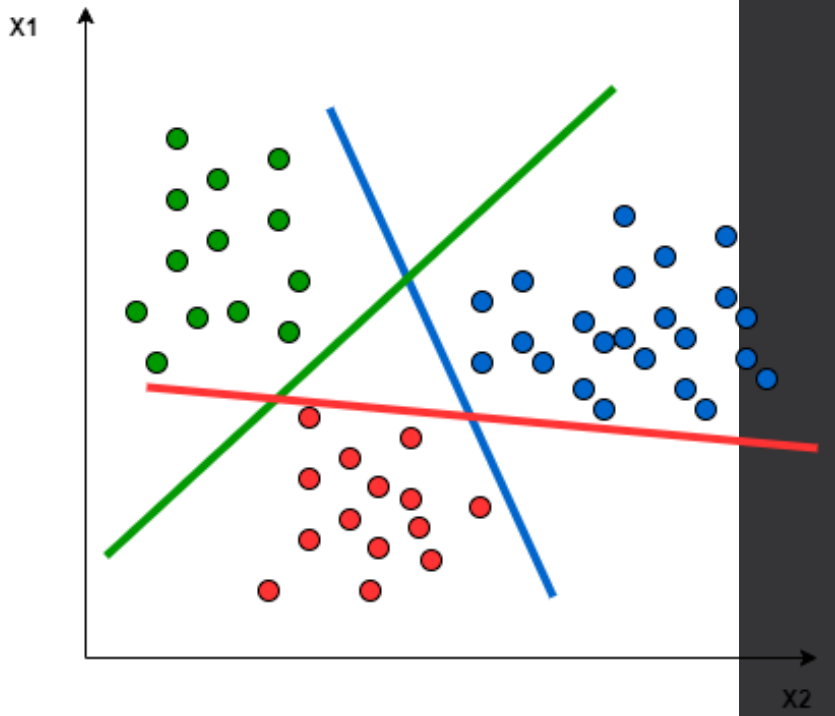
Um por Um

- Divide um problema Multiclasse em dois ou mais problemas de classificação binária, retornando um par de classificadores por classe.
- Utiliza o conceito da maioria e das distâncias de margem como principal critério.
- Objetivo é encontrar o hiperplano ideal entre duas classes, ignorando a outra.
- Pode ser necessário o treinamento de diversas SVMs



Um contra todos

- Nesta técnica, se tivermos N classes, existirão N SVMs.
- O objetivo é encontrar o hiperplano que isole a classe selecionada das demais.
- A principal desvantagem é o de muita computação em problemas maiores.
- Além disso, problemas podem sofrer com o desbalanceamento.



Grafo Acíclico Direcionado

- Foi criada para melhorar ou tentar acabar com as desvantagens presentes nas demais técnicas. Possuindo um número muito menor de SVMs que a técnica One x All, por exemplo.
- A principal desvantagem é que pelo fato de trabalhar em grupos, a depender do dataset, pode ocorrer uma demora, ou a necessidade da escolha manualmente.
- Por ser um grafo, a possibilidade a erros é menor. Porém se decompõe em diversas classificações binárias

Referências

- SVM – Entendendo Sua Matemática – Parte 3 – O Hiperplano ótimo - Laboratório iMobilis (ufop.br)
- Máquina de Vetores de Suporte — SVM | by Matheus Remigio | Medium
- GitHub - vfcarida/Live_SVM_Estatidados: Live_SVM_Estatidados
- 16. Learning: Support Vector Machines - YouTube
- SVM1 Expression of the decision function - YouTube