

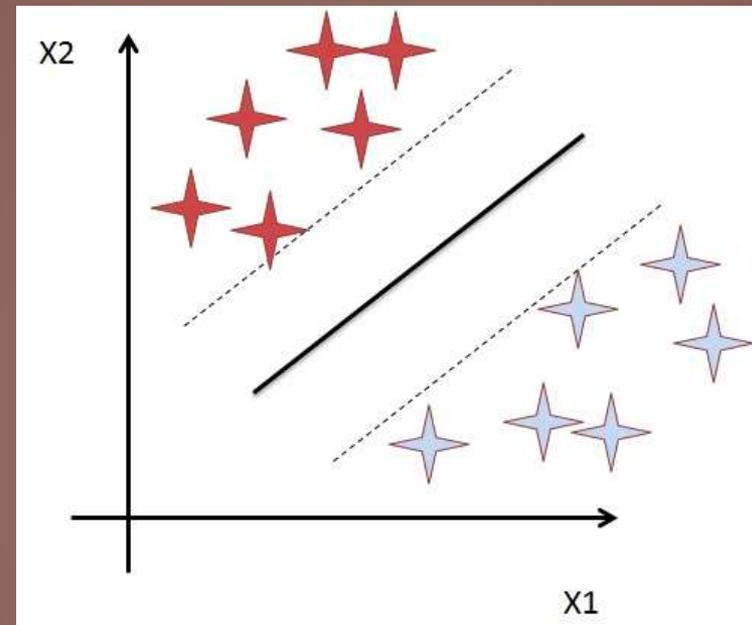
SVM – Support Vector Machines

Prof. Dr. Vladimir Costa de Alencar
LANA/UEPB

www.valencar.com

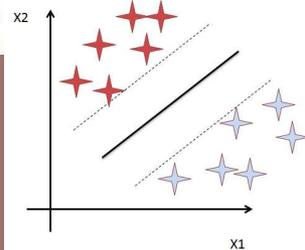
SVM – Support Vector Machines

São modelos de aprendizagem supervisionada, que possuem algoritmos de aprendizagem que analisam dados e reconhecem padrões, utilizados para classificação e análise de regressão

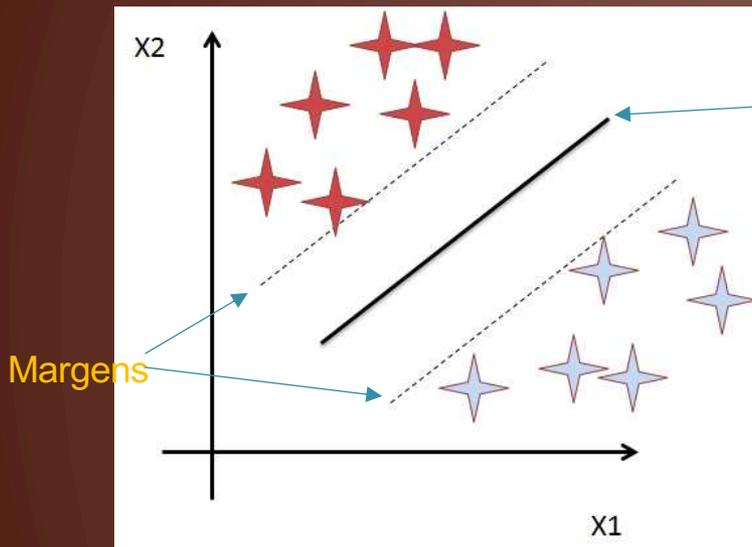


SVM – Support Vector Machines

- Proposto em 1995 pelo russo **Vladimir Vapnik**.
- Consiste em um método de aprendizado que tenta encontrar a **maior margem** para separar diferentes classes de dados.
- Pertence à classe de algoritmos de **aprendizado supervisionado**.
- A essência do SVM é a construção de um **hiperplano ótimo**, de modo que ele possa separar diferentes classes de dados com a maior margem possível.



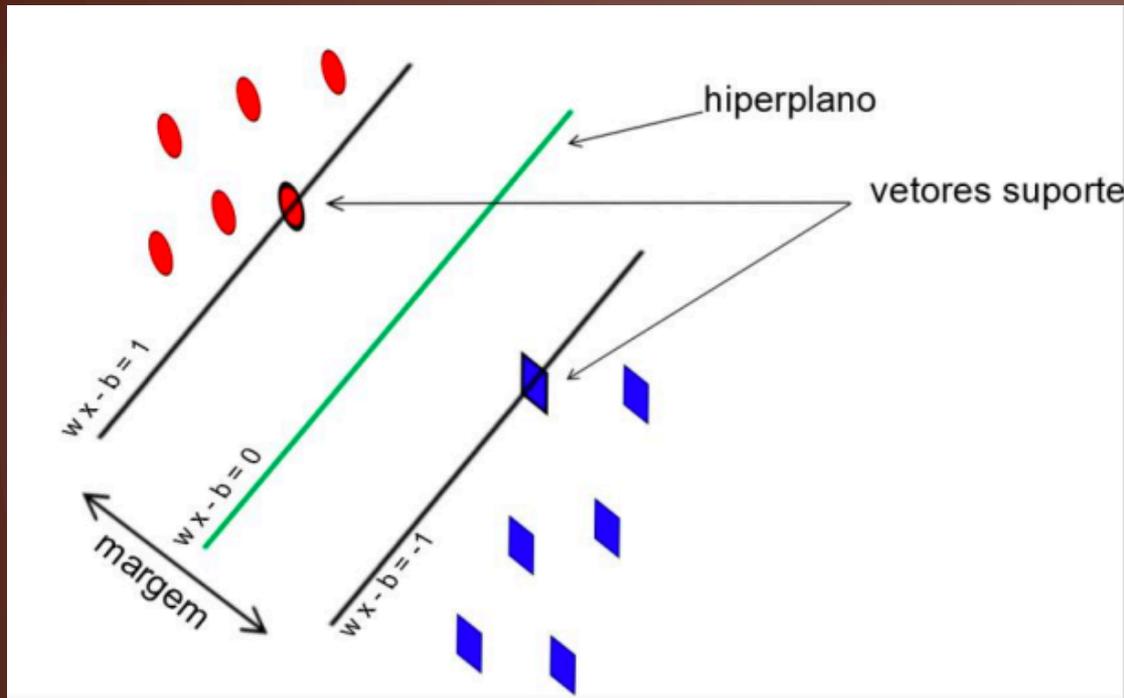
SVM – Support Vector Machines



Hiperplano
(separador das classes)

SVM é um classificador linear
binário não probabilístico

SVM – Support Vector Machines



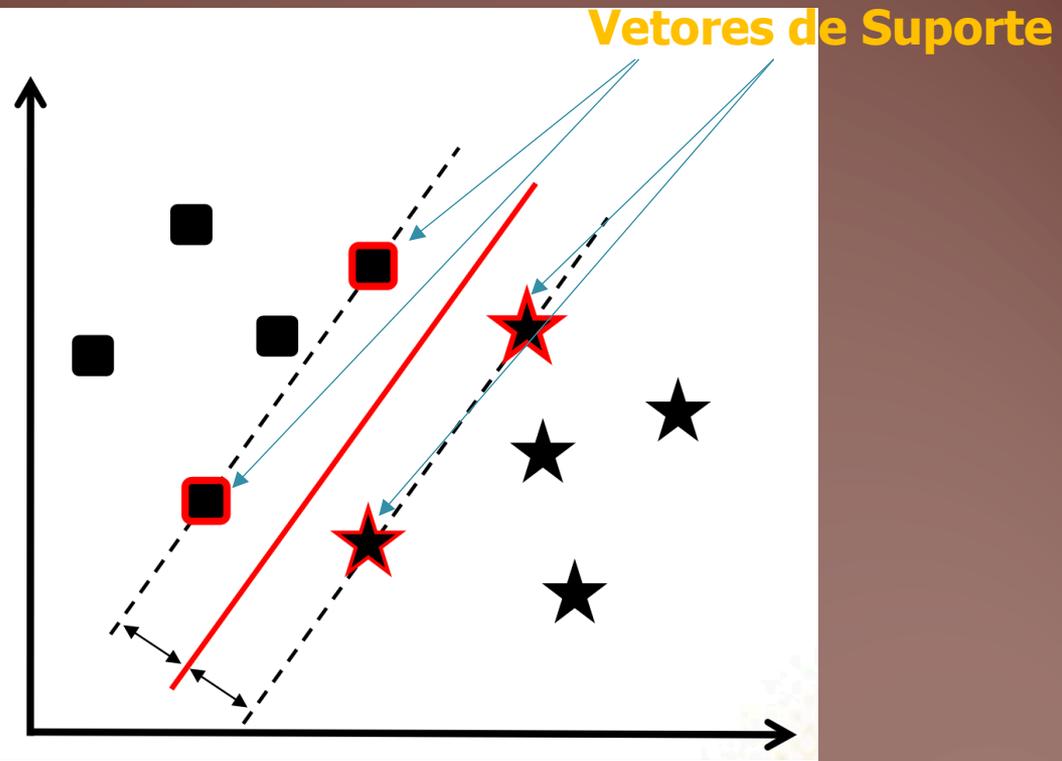
Dados Linearmente Separáveis

SVM Linear

SVM – Support Vector Machines

Vetores de Suporte

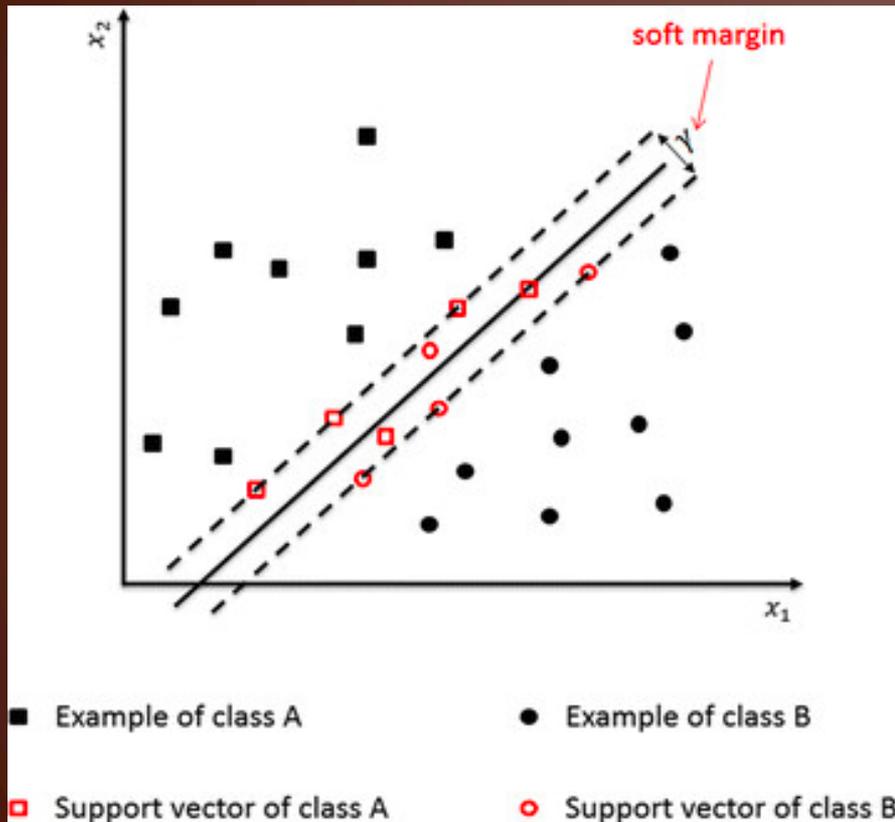
- Servem para definir qual será o hiperplano.
- São encontrados durante a fase de treinamento.
- Os vetores de suporte são os exemplos de treinamento realmente importantes. Os outros exemplos podem ser ignorados.



SVM – Support Vector Machines

- A aplicação de um método puramente linear para classificar um conjunto de dados pode sofrer com **dois problemas** bastante comuns:
 - Outliers
 - Exemplos rotulados erroneamente
- Mesmo assim o SVM ainda assim pode ser aplicado através do uso do parâmetro C (**soft margin** - variáveis de folga)

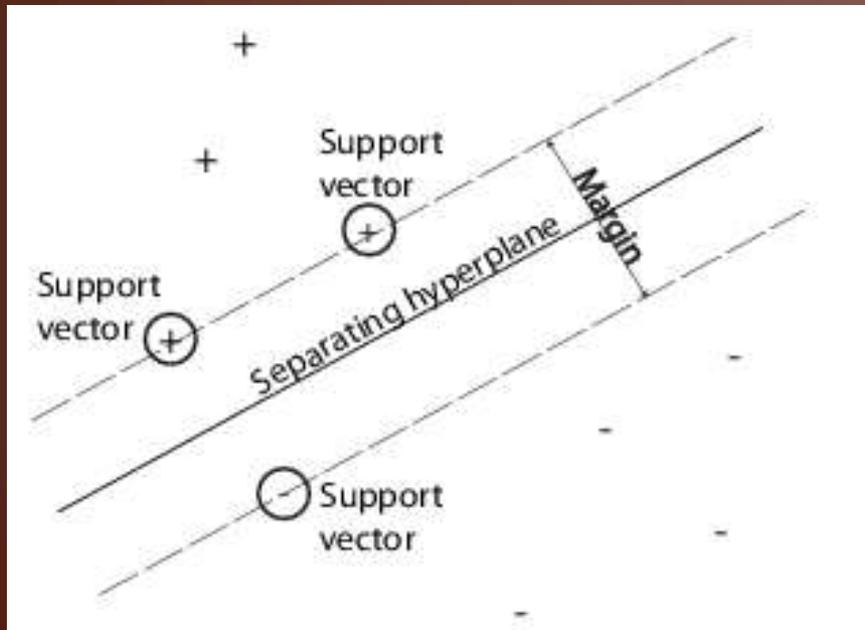
SVM – Support Vector Machines



Os SVM de Soft-Margin (margens suaves) permitem algumas classificações de dados com erro (o custo é definido pelo parâmetro C)

SVM – Support Vector Machines

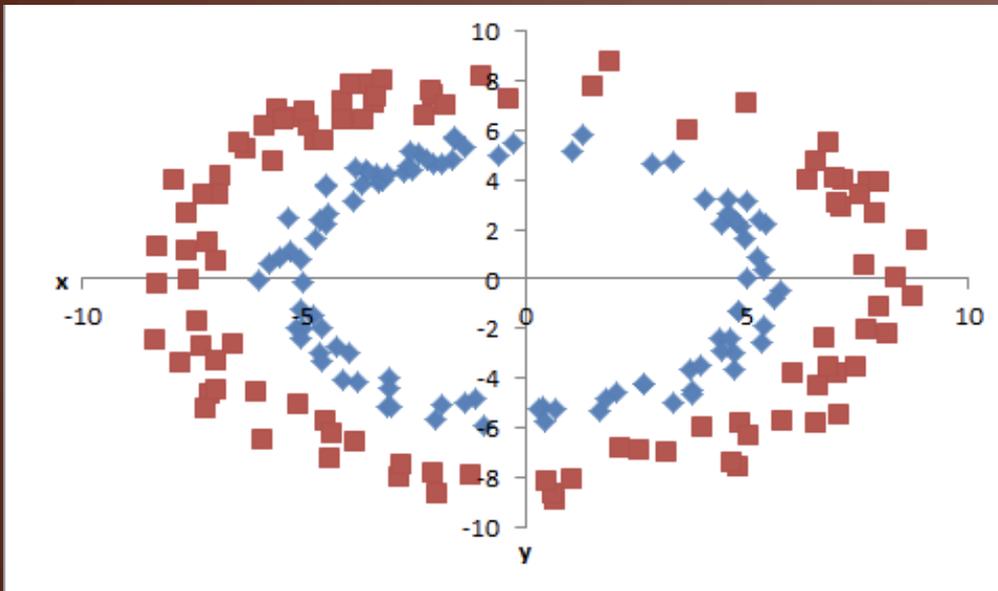
Dados Linearmente Separáveis SVM Linear



O melhor hiperplano para uma SVM significa aquele com a maior margem entre as duas classes

SVM – Support Vector Machines

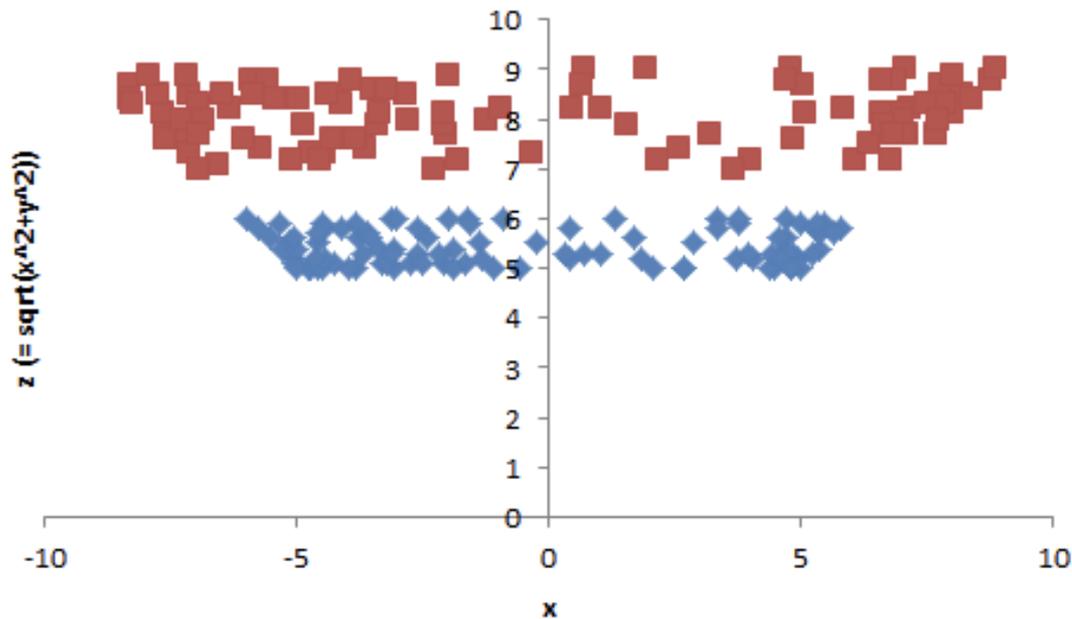
Dados Não Linearmente Separáveis SVM Não Linear



Alguns problemas de classificação binária não têm um hiperplano simples como um critério de separação útil

SVM – Support Vector Machines

Dados Não Linearmente Separáveis SVM Não Linear

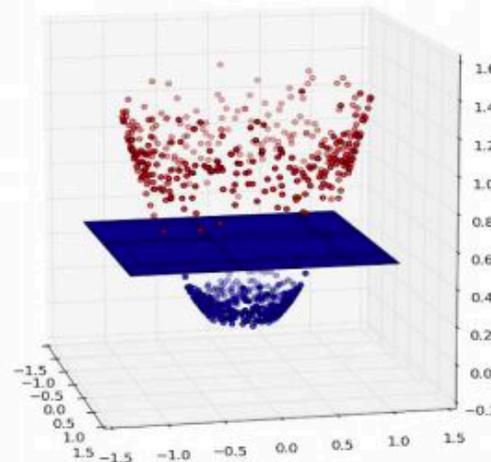
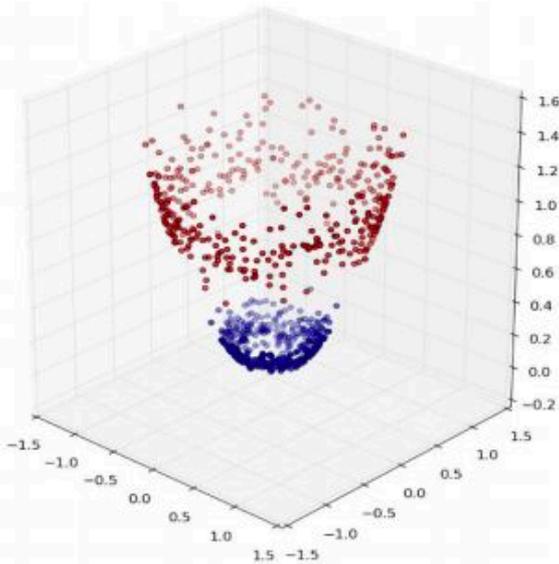


Um truque simples seria transformar as duas variáveis x e y em um novo espaço de característica envolvendo x (ou y) e uma nova variável z definida como $z = \sqrt{x^2 + y^2}$

→ Truque do kernel (Kernel Trick)

SVM – Support Vector Machines

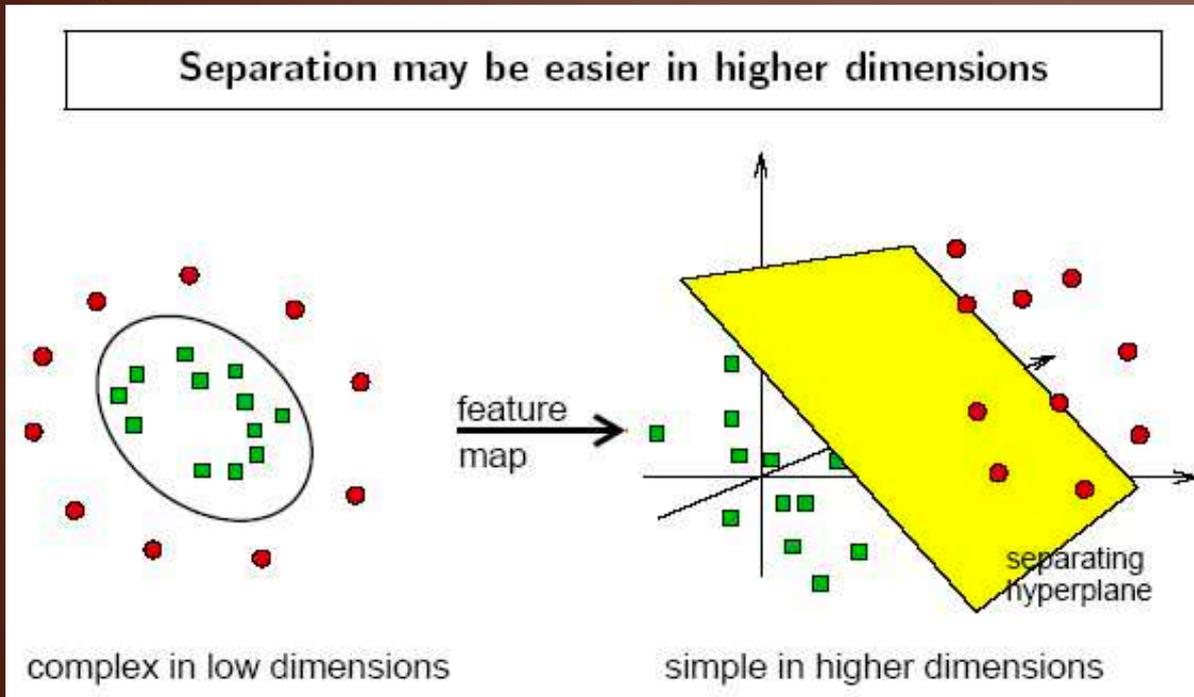
Dados Não Linearmente Separáveis SVM Não Linear



Novo espaço de característica envolvendo x (ou y) e uma nova variável z definida como $z = \sqrt{x^2 + y^2}$

SVM – Support Vector Machines

Função Kernel



SVM implementa um mapeamento não-linear (executado por um kernel escolhido) dos dados de entrada para um espaço de características (feature space) de alta-dimensão, em que um hiperplano ótimo é construído para separar os dados linearmente em duas classes.

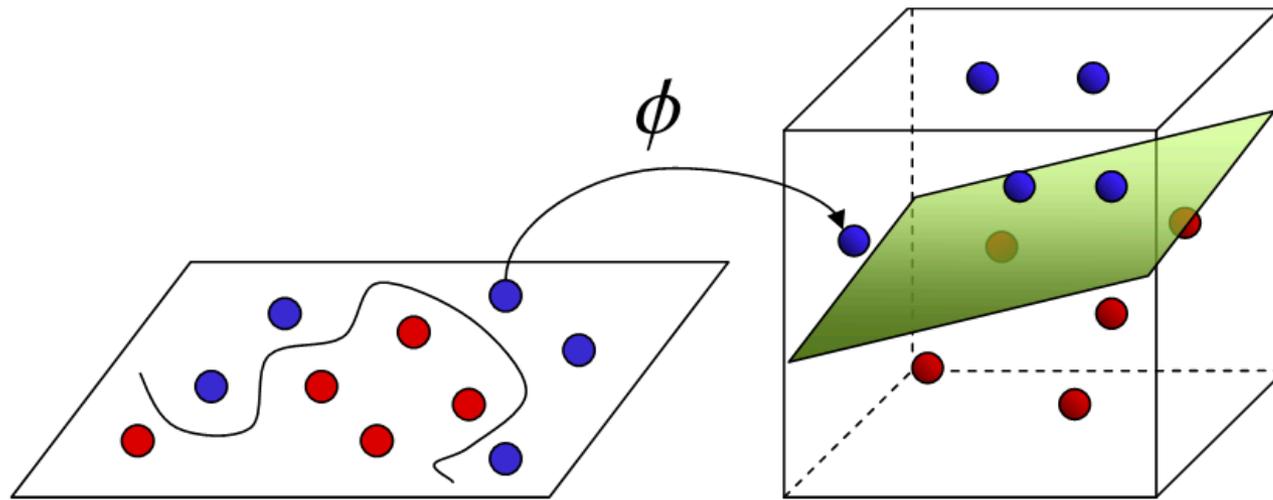
SVM – Support Vector Machines

Como qualquer modelo de aprendizagem supervisionado, você primeiro treina uma máquina de vetores de suporte e, em seguida, valida o classificador.

Para conseguir um nível de acurácia satisfatório, precisamos fazer o **tuning** dos parâmetros das funções de kernel!

SVM – Support Vector Machines

Métodos Kernel

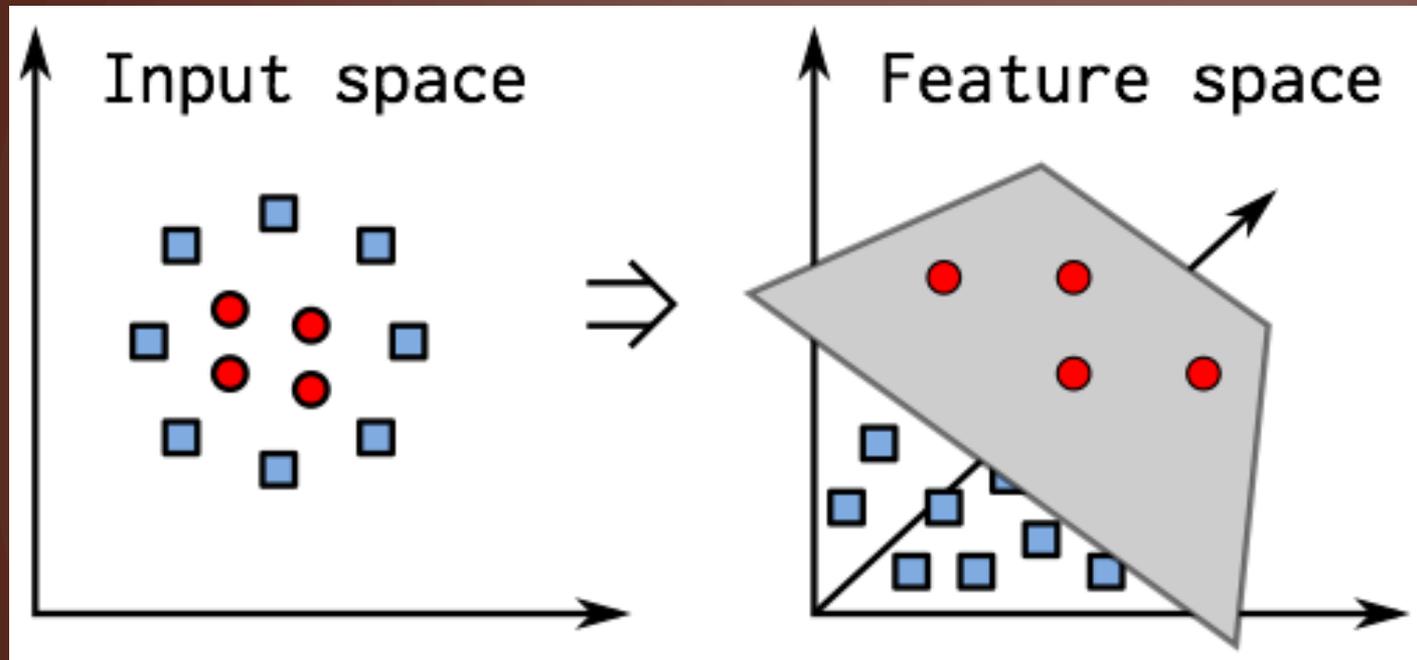


Input Space

Feature Space

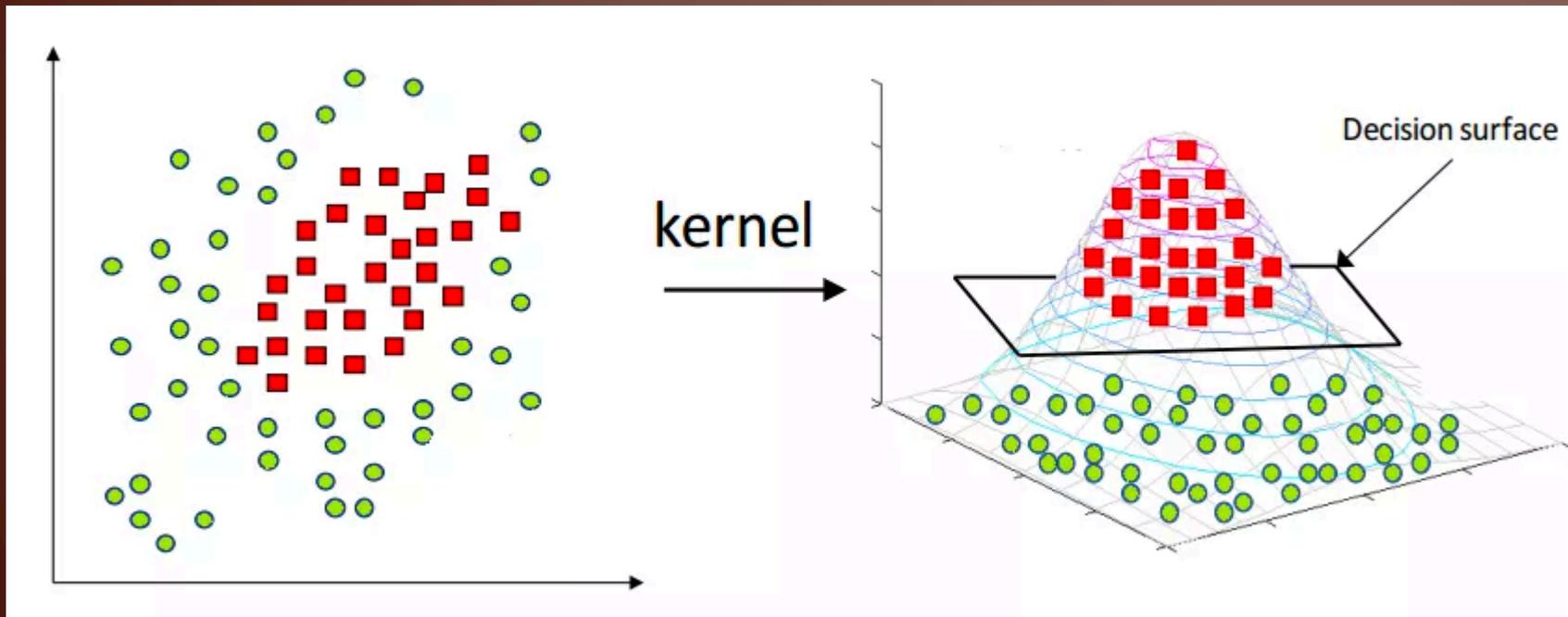
SVM – Support Vector Machines

Métodos Kernel

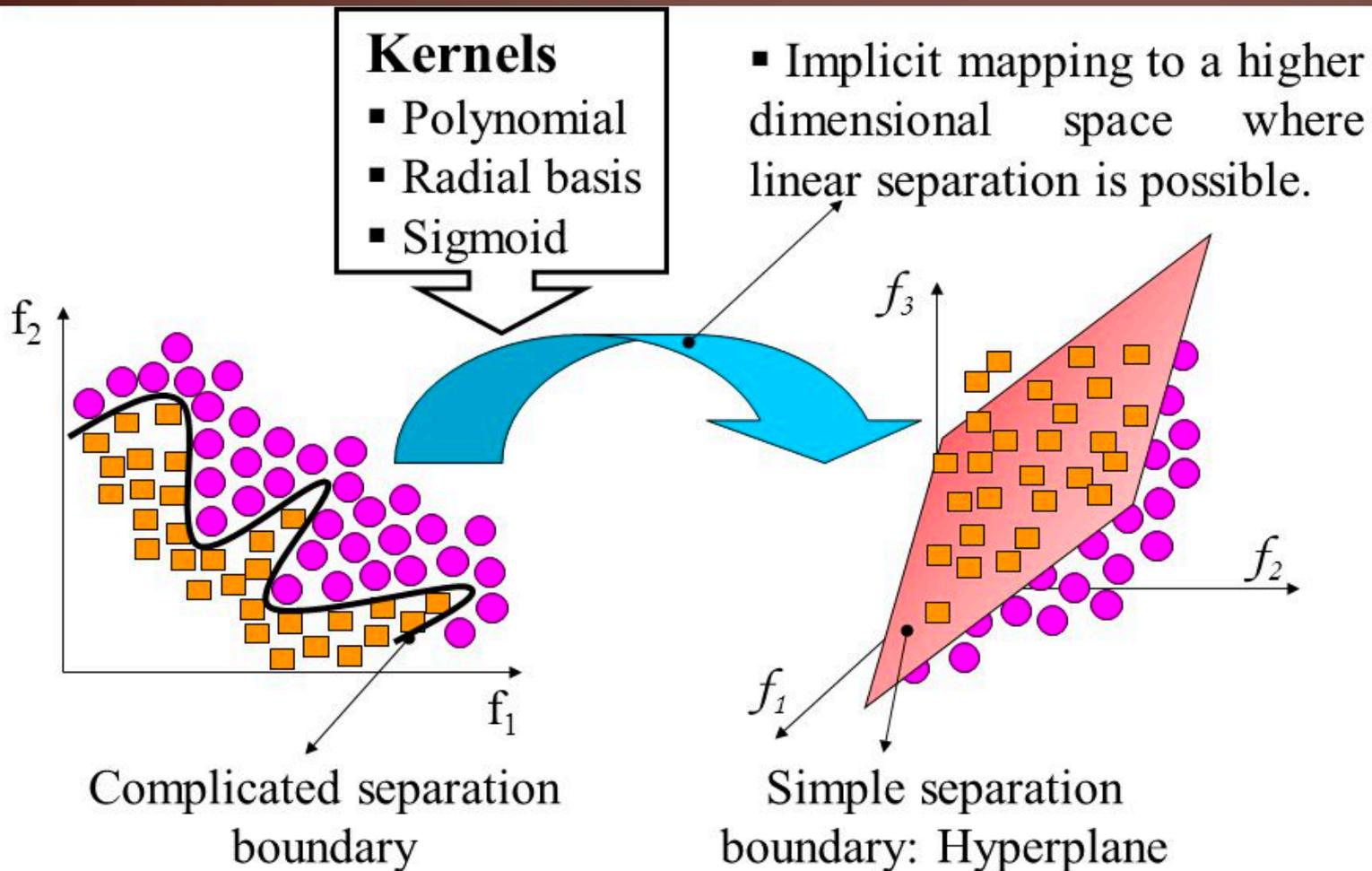


SVM – Support Vector Machines

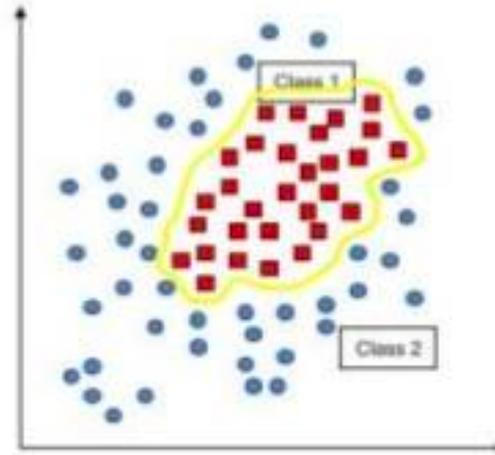
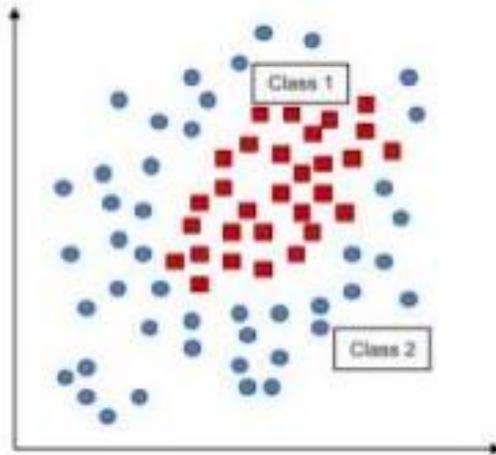
Truque do Kernel → substitui os atributos (preditores) por uma função kernel



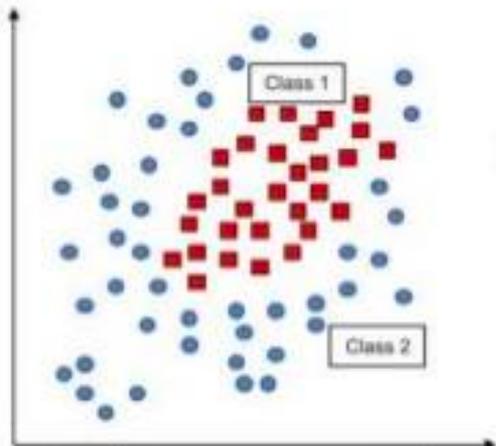
SVM – Support Vector Machines



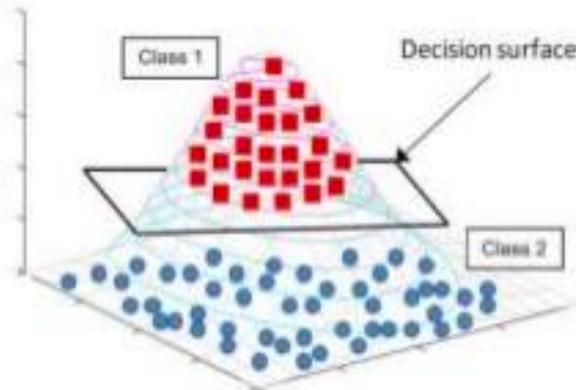
SVM – Support Vector Machines



Non Linear
Decision
Boundary



kernel
→

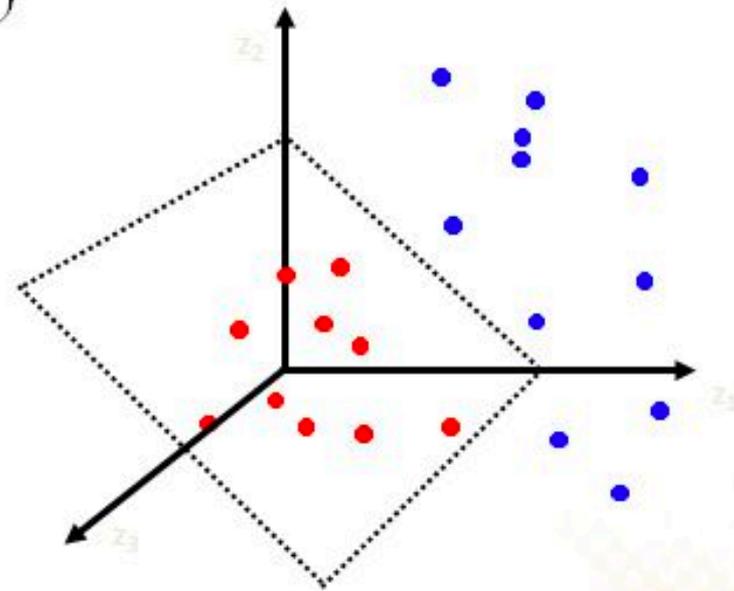
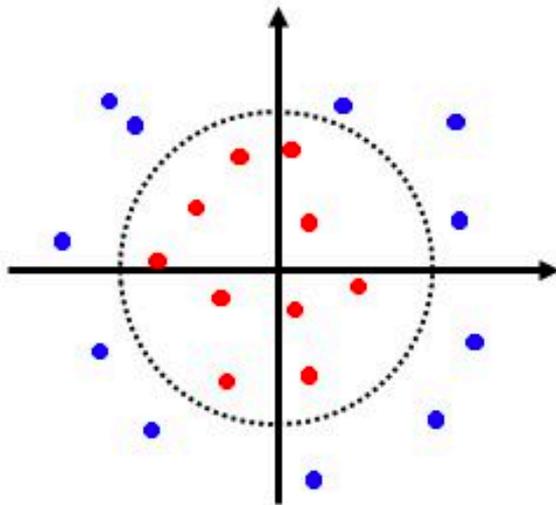


Kernel
method

SVM – Support Vector Machines

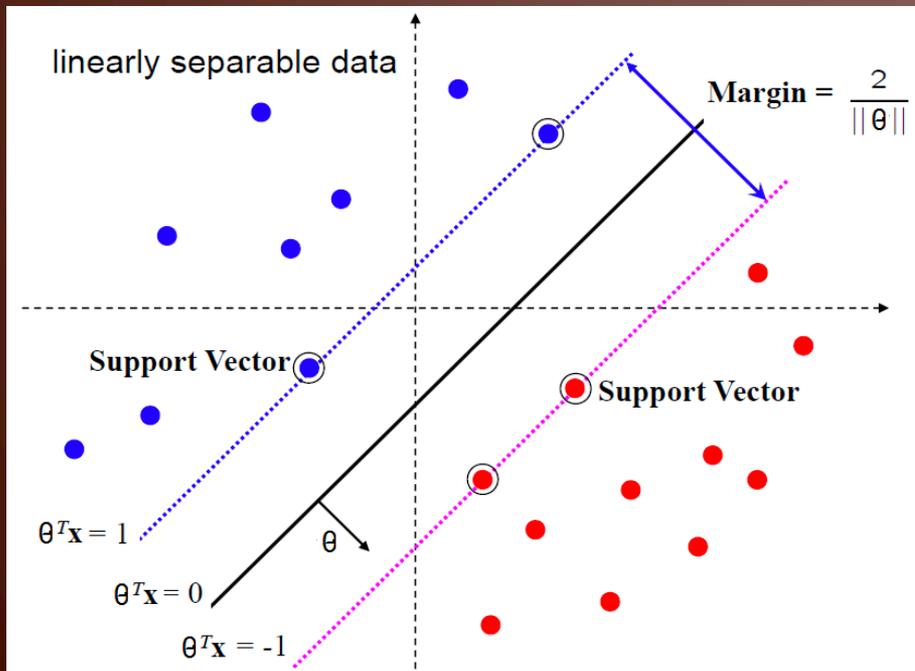
Funções de Kernel

$$\phi(x_1, x_2) \rightarrow (z_1, z_2, z_3) = (x_1^2, \sqrt{2x_1x_2}, x_2^2)$$



SVM – Support Vector Machines

Em SVM a complexidade da construção do modelo depende do número de vetores de suporte e não da dimensão do espaço de características

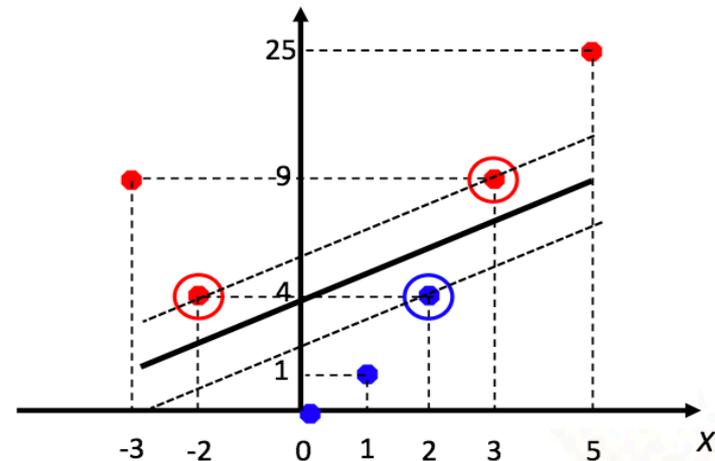


SVM – Support Vector Machines

- Considerando o seguinte conjunto de exemplos de treinamento que não são linearmente separáveis:



- Elevando para uma dimensão linearmente separável ($\mathbb{R}^1 \rightarrow \mathbb{R}^2$):

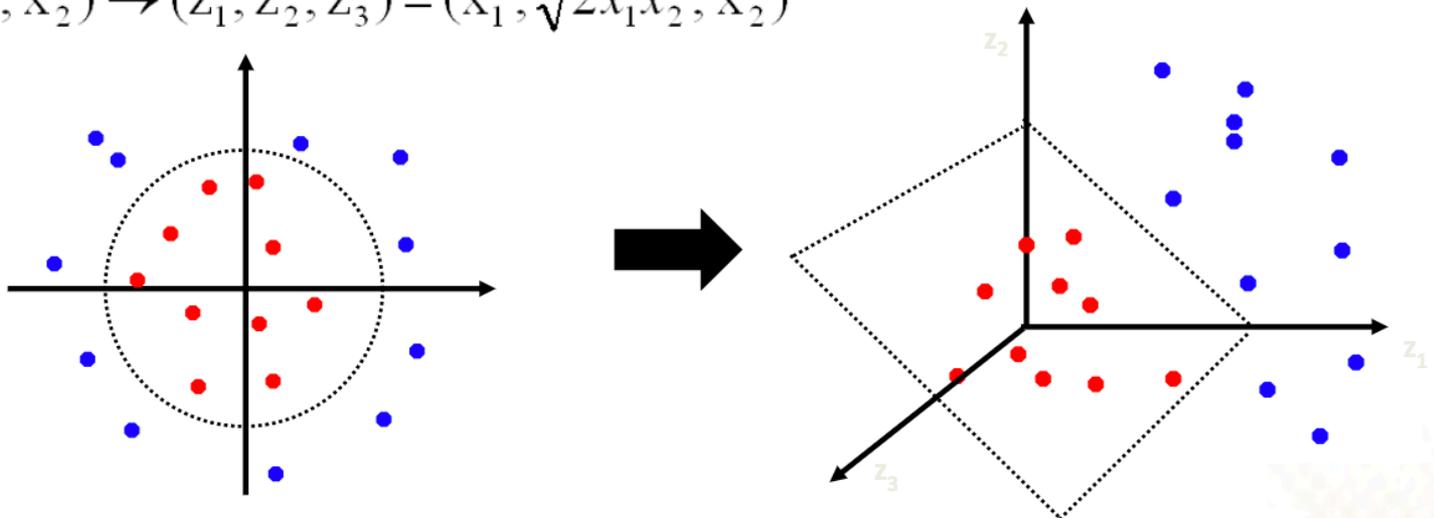


- **Kernel:** $\phi(x) = (x, x^2)$

SVM – Support Vector Machines

- A mesma metodologia pode ser aplicada em um espaço 2D de características ($\mathbb{R}^2 \rightarrow \mathbb{R}^3$).
- A única diferença é a necessidade de uma nova função de kernel. Um exemplo de função de kernel aplicável nesse caso seria:

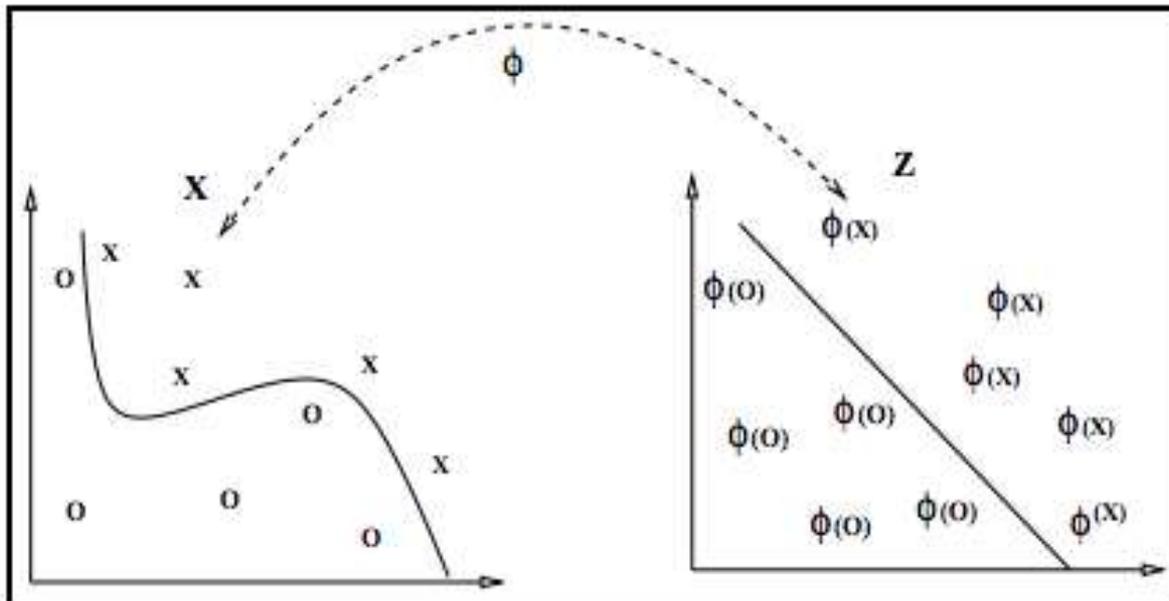
$$\phi(x_1, x_2) \rightarrow (z_1, z_2, z_3) = (x_1^2, \sqrt{2x_1x_2}, x_2^2)$$



SVM – Support Vector Machines

Funções de Kernel

Em outras palavras, embora a dimensão do espaço aumente em Z , a complexidade diminui, porque a classificação, que no espaço de entrada só era possível utilizando superfícies de decisão não lineares, no espaço de características, pode ser feita apenas com um simples hiperplano (superfície de decisão linear).



SVM – Support Vector Machines

Kernel:

RBF ou Gaussiana – Bom para problemas não lineares

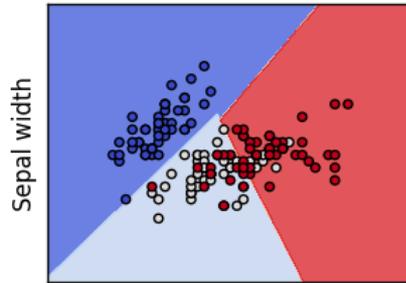
Sigmoid – Tem uma semelhança com Redes Neurais

- linear: $K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$.
- polynomial: $K(\mathbf{x}_i, \mathbf{x}_j) = (\gamma \mathbf{x}_i^T \mathbf{x}_j + r)^d, \gamma > 0$.
- radial basis function (RBF): $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2), \gamma > 0$.
- sigmoid: $K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\gamma \mathbf{x}_i^T \mathbf{x}_j + r)$.

SVM – Support Vector Machines

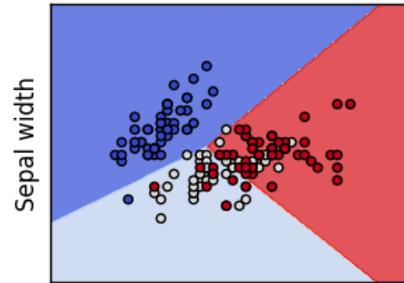
Funções de Kernel

SVC with linear kernel



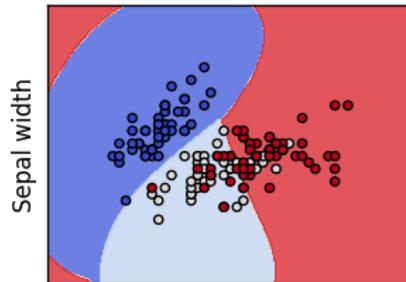
Sepal length

LinearSVC (linear kernel)



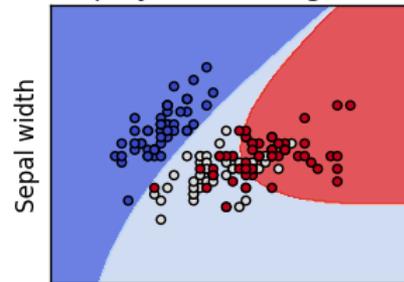
Sepal length

SVC with RBF kernel



Sepal length

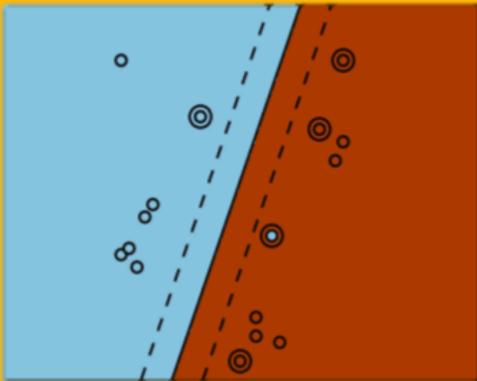
SVC with polynomial (degree 3) kernel



Sepal length

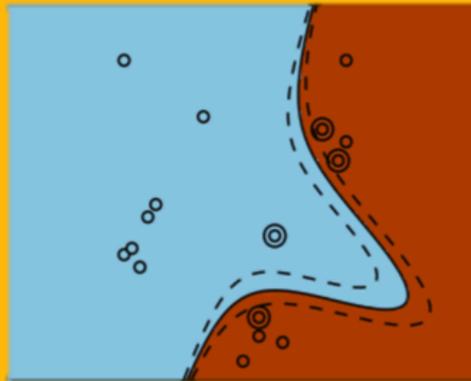
SVM – Support Vector Machines

Linear Kernel



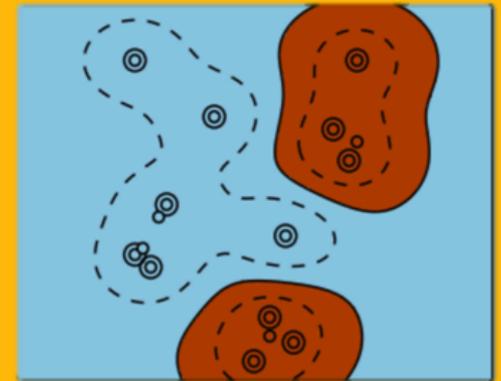
C hyperparameter

Polynomial Kernel



C plus gamma, degree and coefficient hyperparameters

RBF Kernel

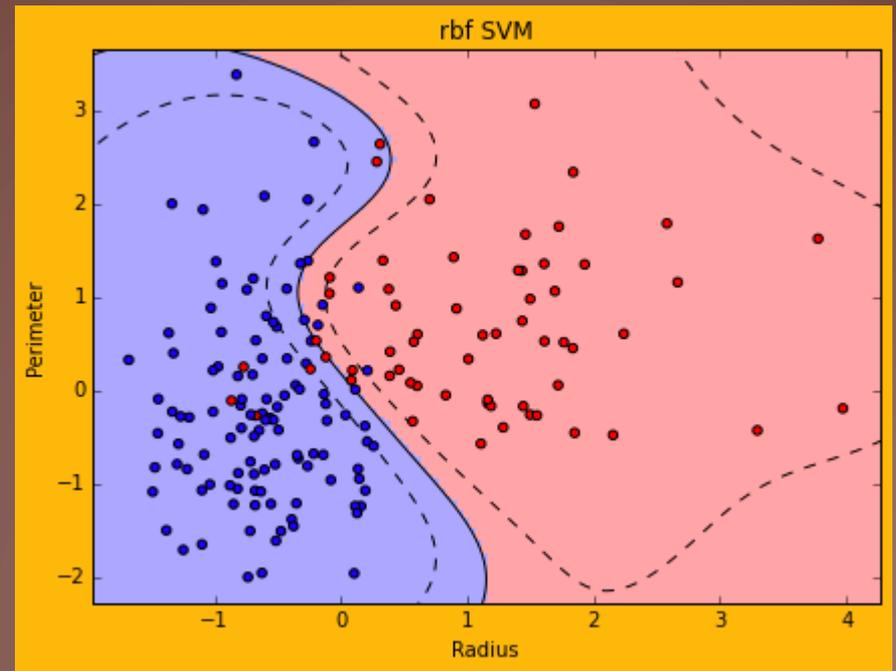
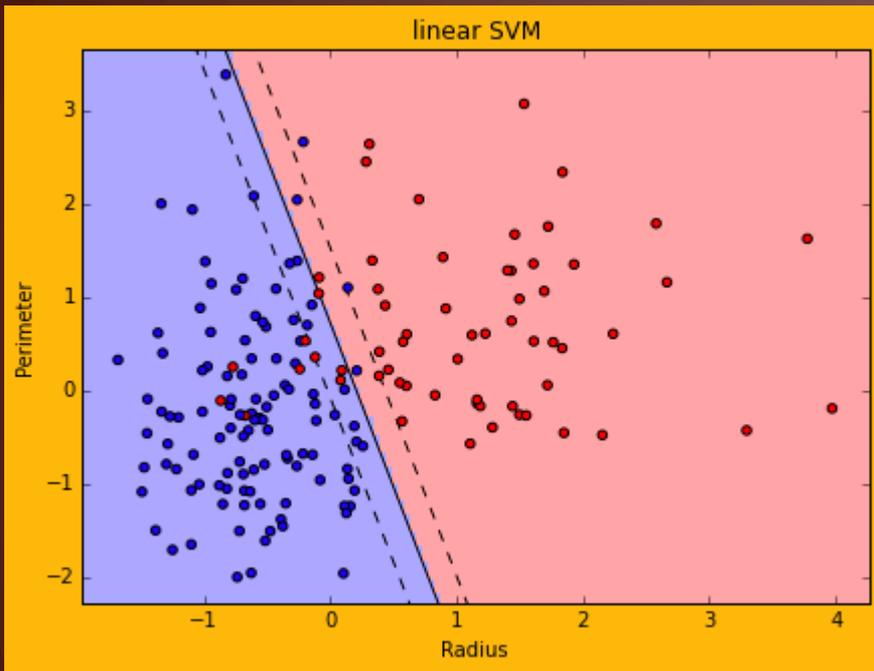


C plus gamma hyperparameter

RBF ou Gaussiana – Bom para problemas não lineares

Sigmoid – Tem uma semelhança com Redes Neurais

SVM – Support Vector Machines



SVM – Support Vector Machines

Antes de aplicar uma SVM para classificar um conjunto de dados é necessário responder algumas questões:

- Quais funções de kernel utilizar?
- Qual o valor do parâmetro C (Soft Margin)?

Validações cruzadas (cross-validations) devem ser utilizadas para evitar overfitting.

SVM – Support Vector Machines

– Quais funções de kernel utilizar?

Linear – Classificações binárias (dados lineares)

RBF (Gaussiana)/Sigmoidal – Bom para problemas não lineares

– Qual o valor do parâmetro C (Soft Margin)?

O valor de C é um trade-off entre Erros e a Simplicidade de uma Superfície de Decisão.

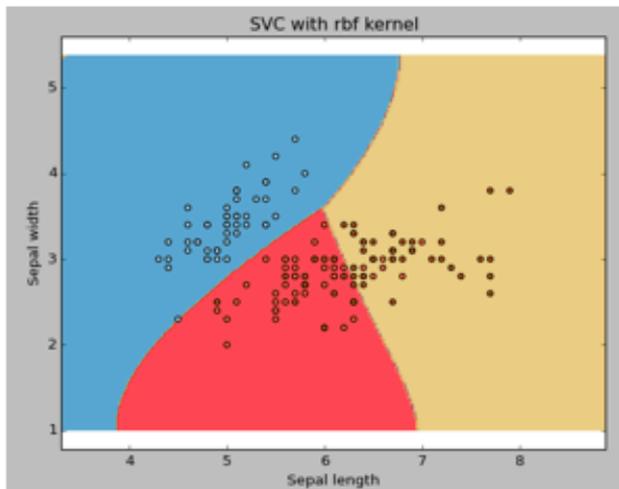
.Um baixo C tem uma tolerância de erros de classificação e permite margens mais suaves (simples)

.Um C alto podem levar a casos de overfitting (aprende o erro também)

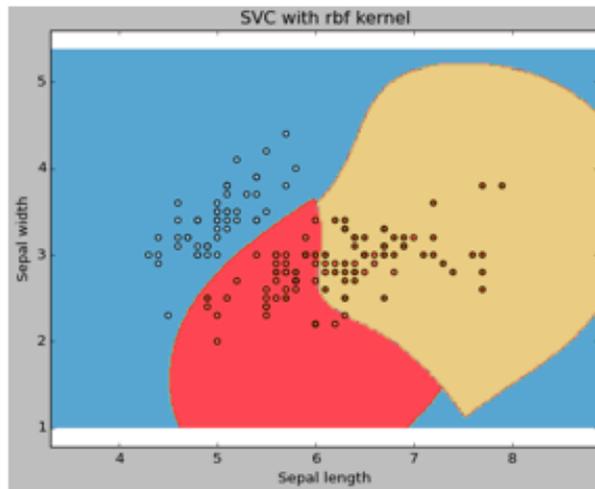
SVM – Support Vector Machines

- .Um baixo C tem uma tolerância de erros de classificação e permite margens mais suaves (simples)
- .Um C alto podem levar a casos de overfitting (aprende o erro também)

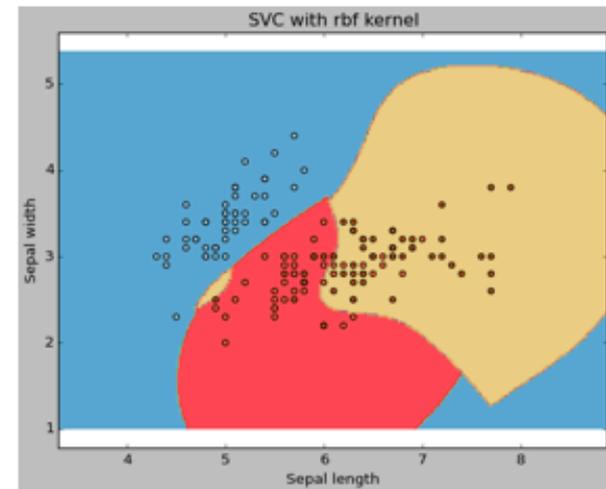
c = 1



C = 100



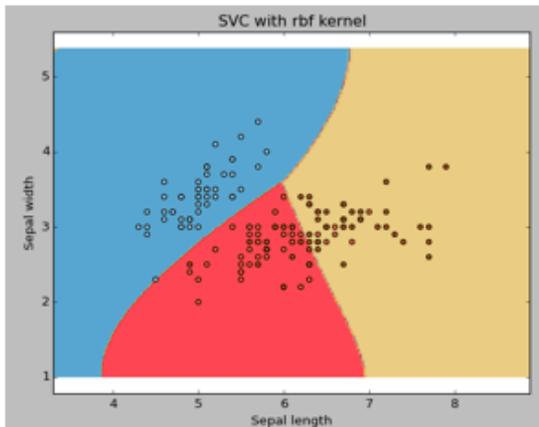
c = 1000



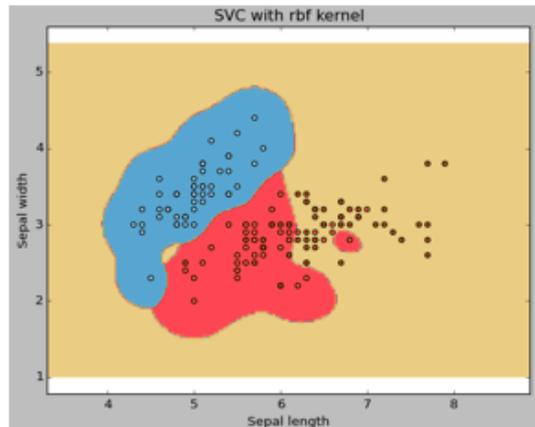
SVM – Support Vector Machines

- Em kernels não lineares temos o parâmetro **gamma**
O **gamma** controla a não-linearidades das fronteiras.
- Um baixo **gamma**, tende a ter fronteiras mais amplas e mais lineares.
- Um alto **gamma**, terá fronteiras envolvendo amostras

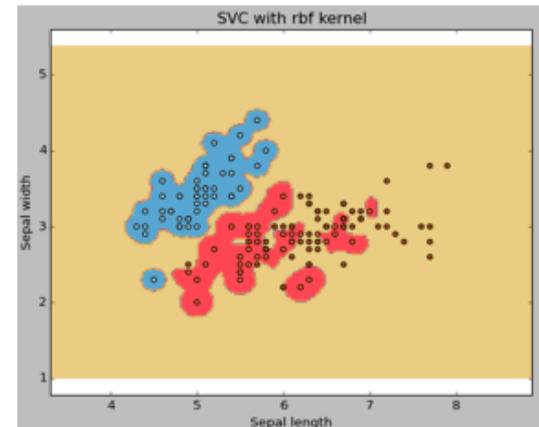
gamma =0



gamma =10



gamma =100



SVM – Support Vector Machines

- **Vantagens:**

- Consegue lidar bem com grandes conjuntos de exemplos.
- Trata bem dados de alta dimensão.
- O processo de classificação é rápido.

- **Desvantagens:**

- É necessário definir um bom Kernel.
- O tempo de treinamento pode ser bem longo dependendo do número de exemplos e dimensionalidade dos dados.