

Aula 8:

Máquina de Vetores de Suporte

Prof. Sérgio Montazzolli Silva
smsilva@uel.br

Introdução

- Máquinas de Vetores de Suporte, ou *Support Vector Machines* (SVM) é um método de classificação baseado em *kernels*
- Considerado um dos classificadores mais poderosos que pode ser encontrado na literatura
- As SVMs foram (e ainda são) amplamente empregadas em diversas áreas da computação e também de áreas correlacionadas
- Ao contrário das RNAs, as SVMs podem ser utilizadas com grande efetividade até mesmo por pessoas sem um conhecimento técnico aprofundado

Introdução

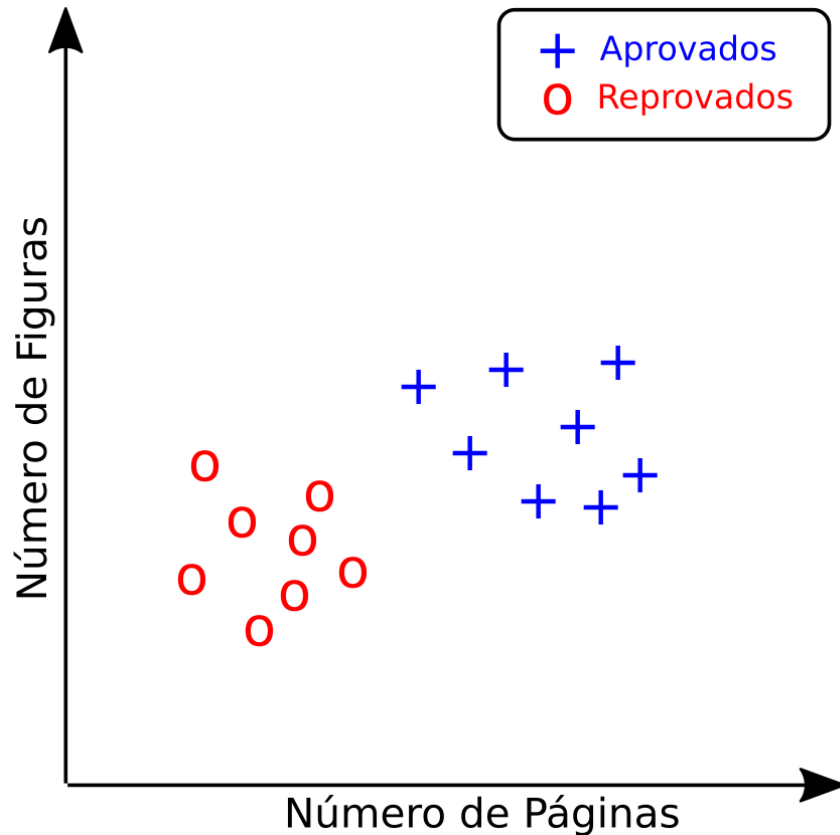
- A idéia da SVM é treinar um classificador linear através de um sub-conjunto dos dados de treinamento, chamados de **vetores de suporte**
 - Isso faz com que o *consumo de memória seja eficiente* em tempo de treinamento
- O sub-conjunto de vetores de suporte inclui as amostras de classificação mais difíceis, ou seja, aquelas que estão mais próximas da fronteira de decisão
 - Logo, são amostras mais informativas, que acabam *acelerando a convergência do treinamento*

Introdução

- Muitas vezes um conjunto não-linearmente separável pode se tornar linearmente separável através da simples aplicação de uma heurística ou função sobre eles
- A SVM utiliza desta premissa para lidar com dados não-lineares, modificando o espaço dos dados de entrada
 - Este método é conhecido com *Kernel-Trick*

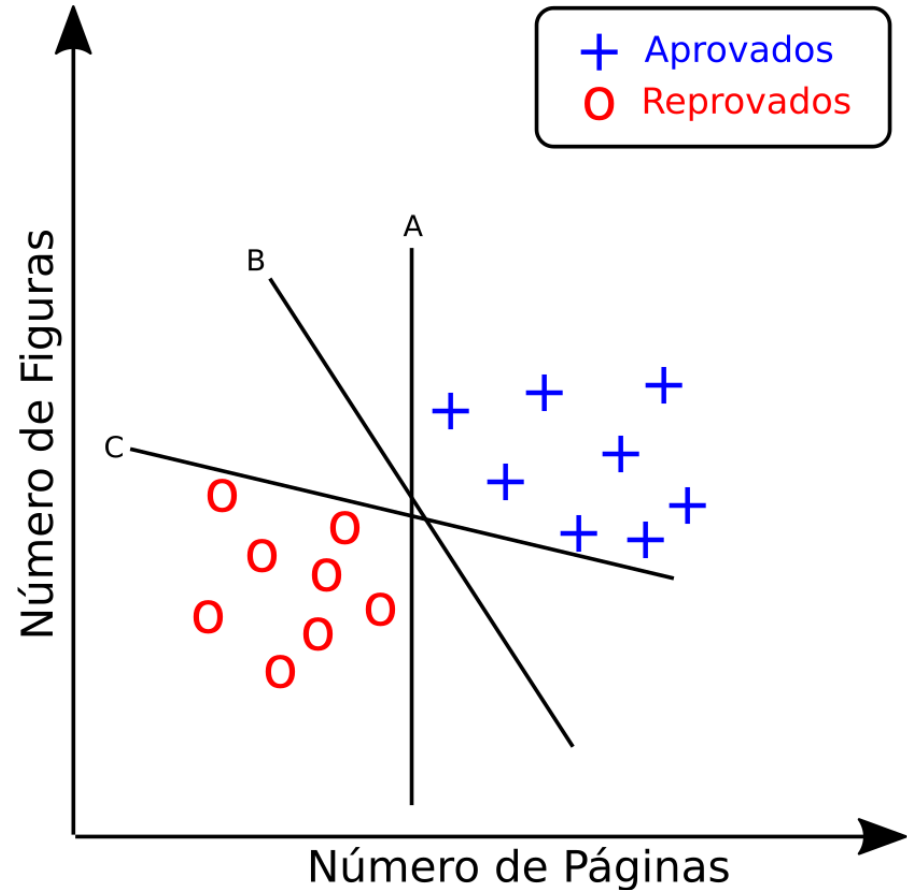
Hiperplano de Decisão

- Para um dado ano do curso de Ciência da Computação, foi feito um levantamento de aspectos textuais dos TCCs, tanto de alunos aprovados como dos reprovados, e gerou-se o seguinte conjunto de dados:



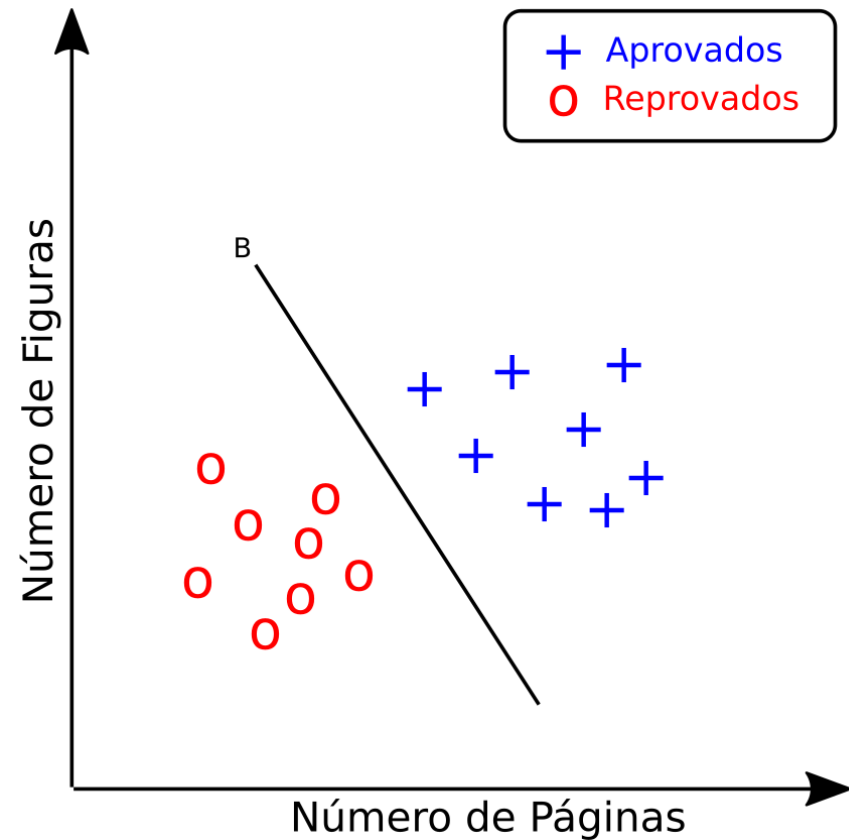
Hiperplano de Decisão

- Qual o melhor hiperplano para este problema? (A,B ou C)
 - Todos resolvem o problema de separação
 - Porém a reta B parece mais adequado
 - Será que podemos encontrar a reta B com um neurônio artificial?



Hiperplano de Decisão

- Será que podemos encontrar a reta B com um neurônio artificial?
- Na teoria sim, porém durante o treinamento de um neurônio, as soluções A, B e C tem chances praticamente iguais de serem encontradas

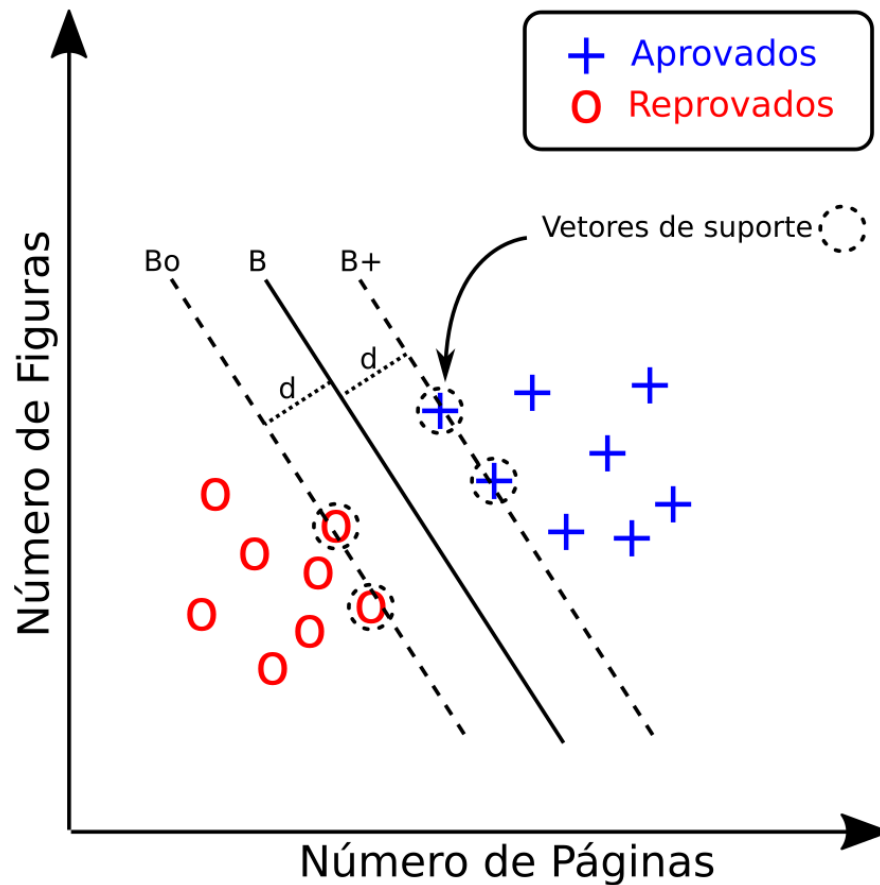


Hiperplano de Decisão

- O treinamento de uma SVM ocorre de maneira um pouco diferente de uma RNA
- Ela seleciona um subconjunto de pontos que estão à margem da fronteira de decisão
 - Estes pontos são conhecidos como **vetores de suporte**
- A partir destes pontos, cria-se margens paralelas, para cada classe
- Estas margens estão separadas por uma distância $2d$ e, exatamente entre elas, existe um hiperplano de separação

Margens

- Exemplo:



Hard Margin

- O treinamento da SVM visa, além da minimização do erro, a maximização da distância d entre as margens
- Na formulação original, chamada de *Hard Margin*, essa maximização não é robusta a ruídos nos dados

Soft Margin

- Para contornar este problema, foi adicionado um novo termo de penalidade na função de custo da SVM
- Este novo termo permite que dados ruidosos sejam desconsiderados ao gerar-se o plano e suas margens
- A permissividade deste novo termo é controlada por um parâmetro C

Parâmetro C

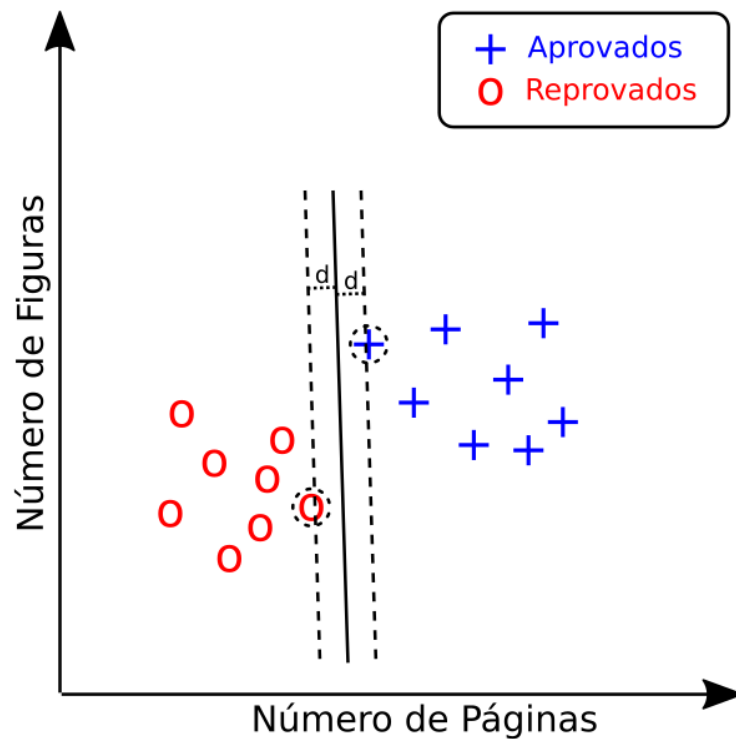
- A distância das margens pode ser configurada através deste parâmetro
- A grosso modo, C é inversamente proporcional a distância d , ou seja: quanto maior C , menos permissivo, e portanto menor é a margem. E quanto menor C , maior é a margem

$$C \propto \frac{1}{d}$$

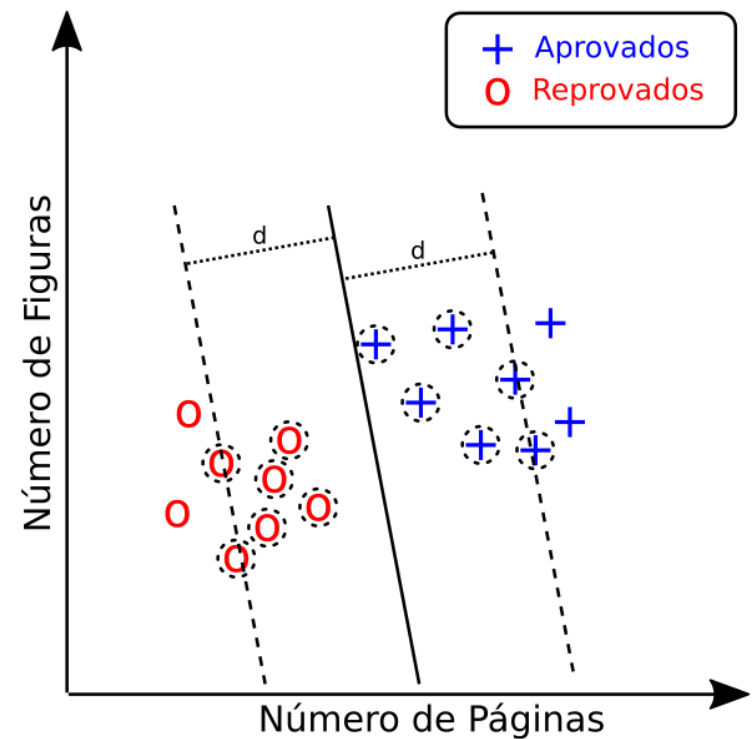
- Em geral, valores C muito grandes podem causar um excecivo sobre-ajuste dos dados (*overfitting*)
- E valores pequenos podem gerar sub-ajuste (*underfitting*)

Parâmetro C

Valor C grande
(d pequeno)



Valor C pequeno
(d grande)



Kernel Trick

- A utilização de kernels é o que faz a SVM ser um algoritmo de classificação poderosíssimo
- A idéia principal é utilizar os vetores suporte para gerar informações novas, que são acopladas como novas dimensões nos dados de entrada, sem alterá-los diretamente
- O tipo de Kernel mais comum é o de base radial, que iremos tratar um pouco mais adianta nesta aula

Kernel Trick

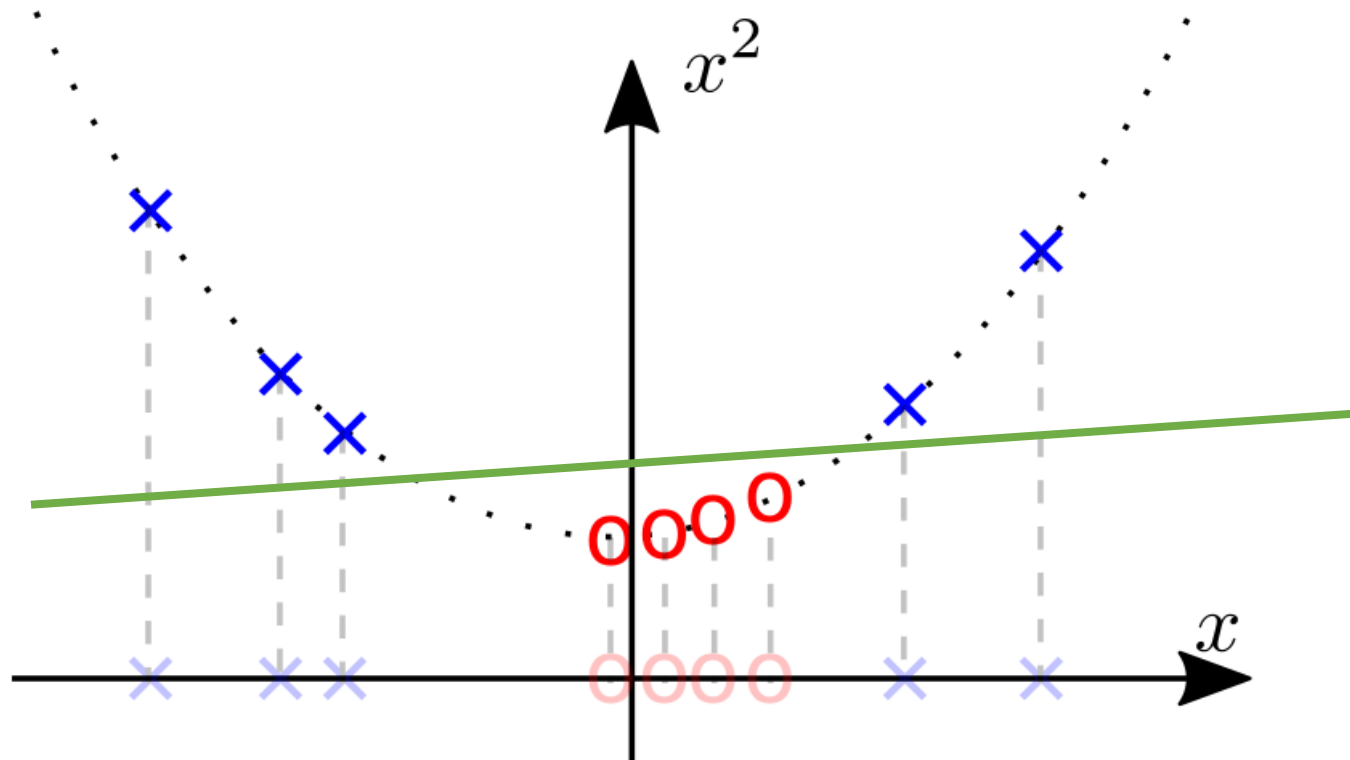
- Imagine que podemos transformar um dado não-linearmente separável em linearmente separável
- Por exemplo:



- Existe algum hiperplano que separe estes pontos corretamente?

Kernel Trick

- Agora, se adicionarmos uma nova dimensão, digamos x^2



Kernel RBF

- O Kernel mais utilizado junto a SVM é o de base radial, ou RBF (*Radial Basis Function*)
- Supondo um ponto \vec{x} e um conjunto de n vetores suporte $X' = \{\vec{x}'_1, \dots, \vec{x}'_n\}$, temos:

$$f(\vec{x}) = \sum_{i=1}^n \alpha_i \exp\left(-\frac{\|\vec{x} - \vec{x}'_i\|^2}{2\sigma}\right)$$

- onde que $f(\vec{x})$ é o valor da nova dimensão, $\alpha_i \in \{-1, 1\}$ o parâmetro que informa a classe ao qual \vec{x}'_i pertence, e σ (também chamado de γ) é um hiperparâmetro correspondente ao desvio padrão do kernel radial

Kernel RBF

- Exemplo

$$f(\vec{x}) = \sum_{i=1}^n \alpha_i \exp\left(-\frac{\|\vec{x} - \vec{x}'_i\|^2}{2\sigma}\right)$$

Kernel RBF

- Sobre o parâmetro σ (ou γ):
 - Valores pequenos geram regiões radiais em volta de cada vetor suporte, podendo causar sobre-ajuste
 - Valores grandes fazem com que o Kernel RBF se aproxime do kernel linear, causando sub-ajuste
 - Ver implementação em <https://cs.stanford.edu/people/karpathy/svmjs/demo/>

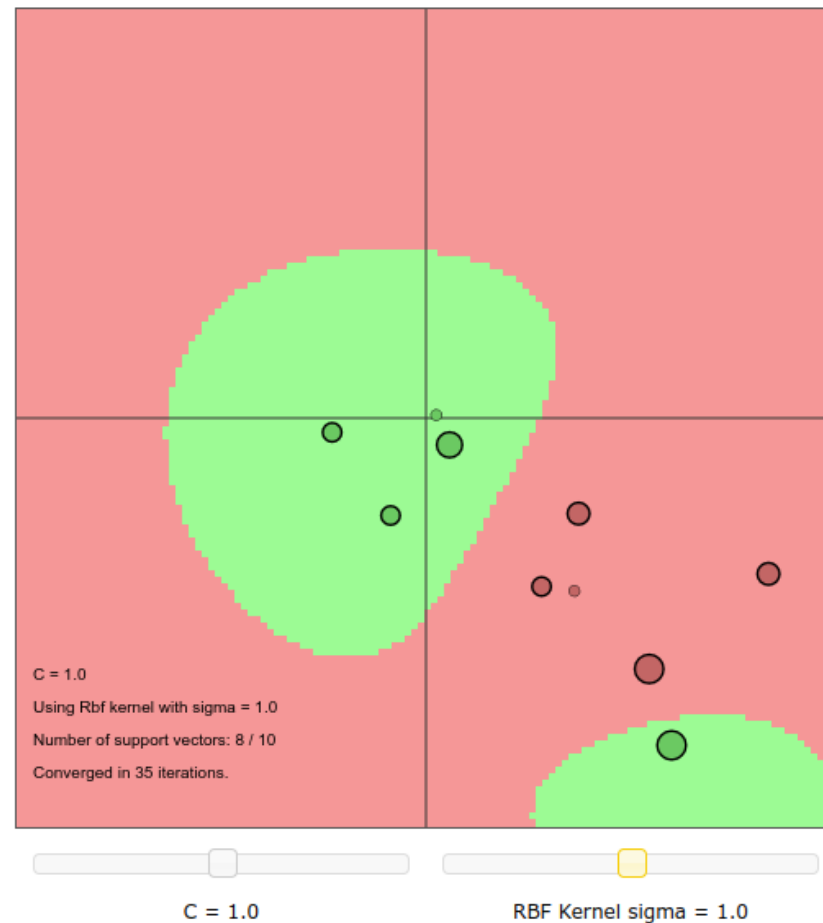
Kernel RBF

- Valor σ pequeno ($\sigma = 0.1$)



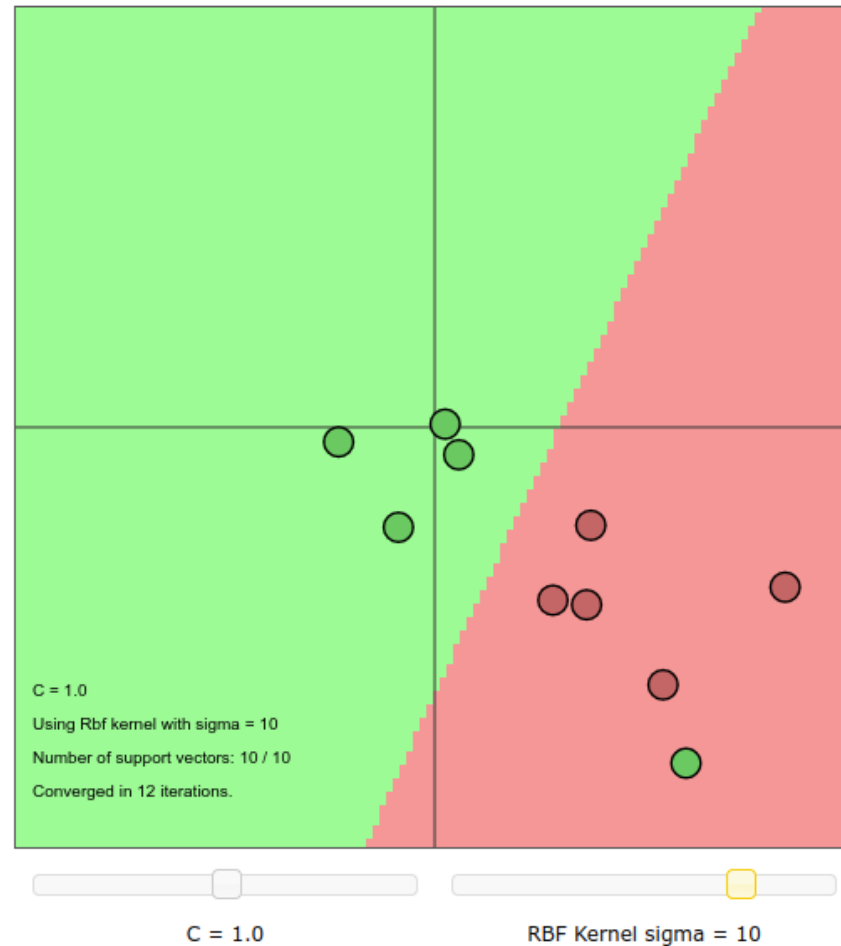
Kernel RBF

- Valor σ padrão ($\sigma = 1$)



Kernel RBF

- Valor σ alto ($\sigma = 10$)



Problemas multi-classe

- SVMs são por natureza classificadores binários
- Em problemas multi-classe, a estratégia mais comum é a utilização de uma SVM para cada classe
- Ou seja, se existem n classes, treina-se n -SVMs
- Cada SVM é treinada com os dados da classe correspondente como sendo os dados positivos, e todos os outros como sendo os negativos

Problemas multi-classe

- Outra estratégia é criar uma SVM para todas as combinações possíveis de classe-vs-classe
 - Ou seja, para um conjunto de n classes, temos:

$$n \cdot (n - 1) / 2$$

classificadores

- A rotulação final para uma certa entrada se dá pela classe mais votada, ou seja, aquela que foi selecionada por mais classificadores