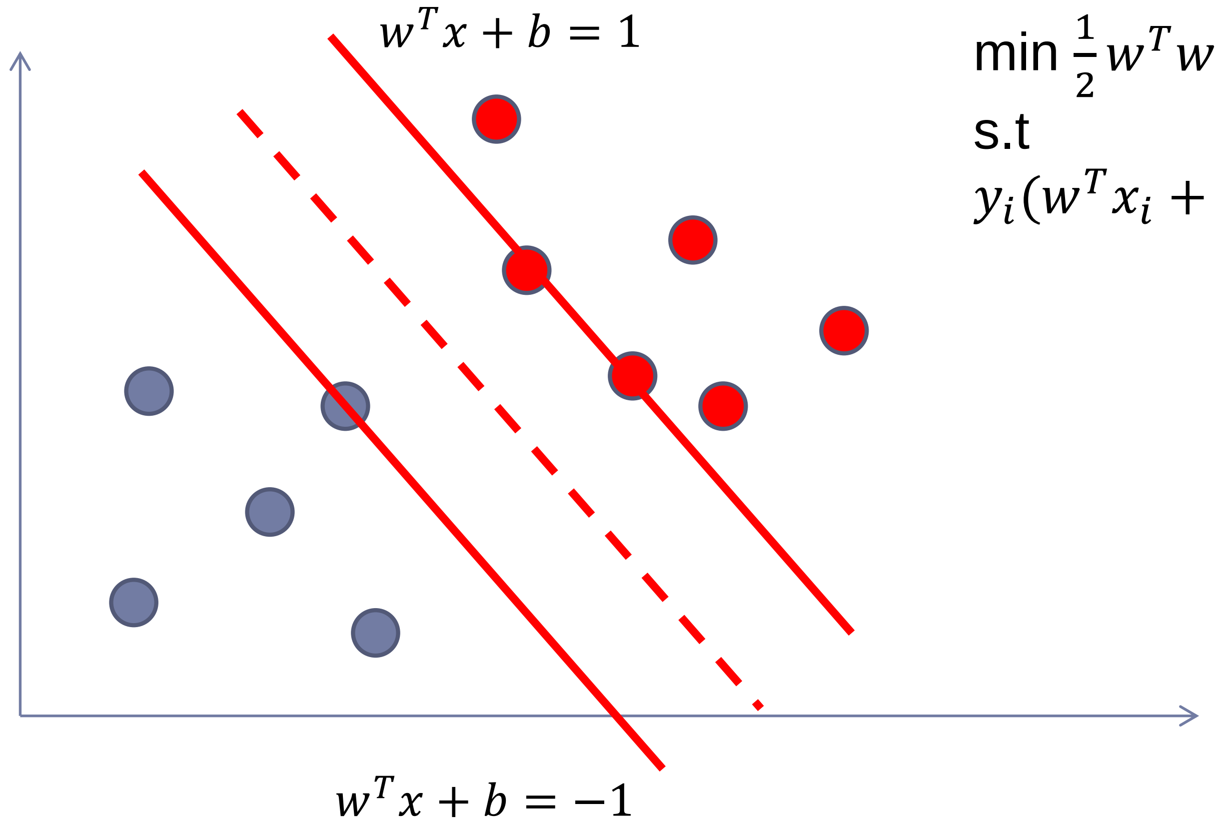
Aprendizagem Automática

João Paulo Pordeus Gomes

Support Vector Machines

SVM

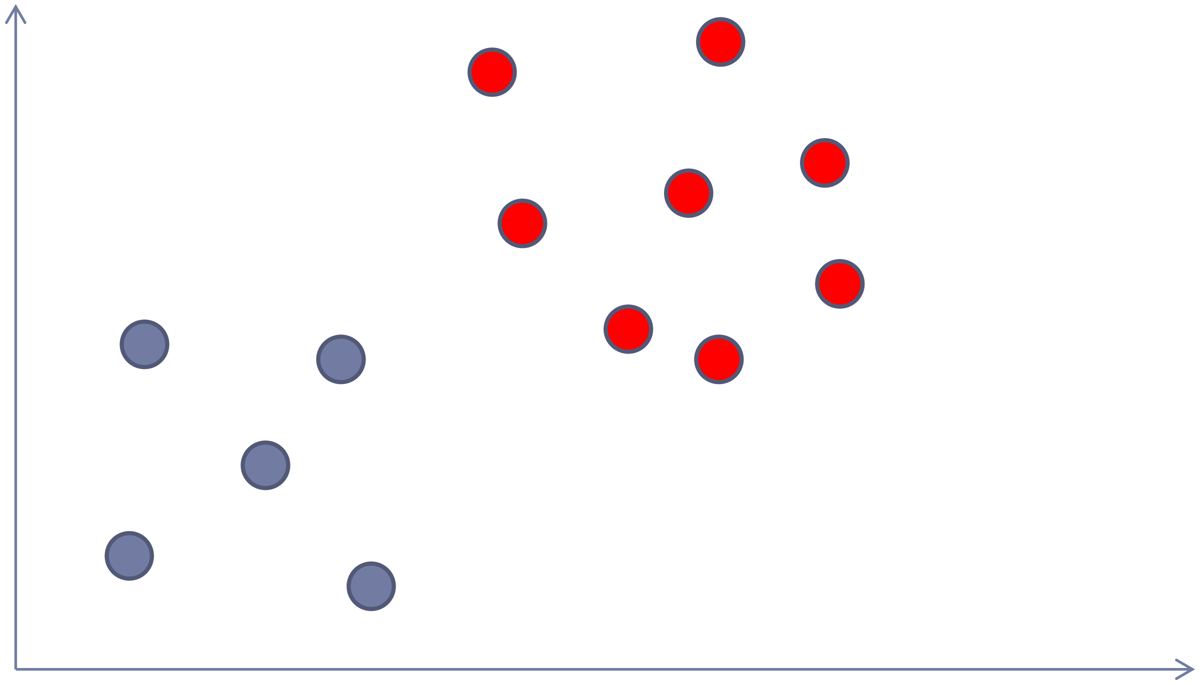


$\min \frac{1}{2} w^T w$

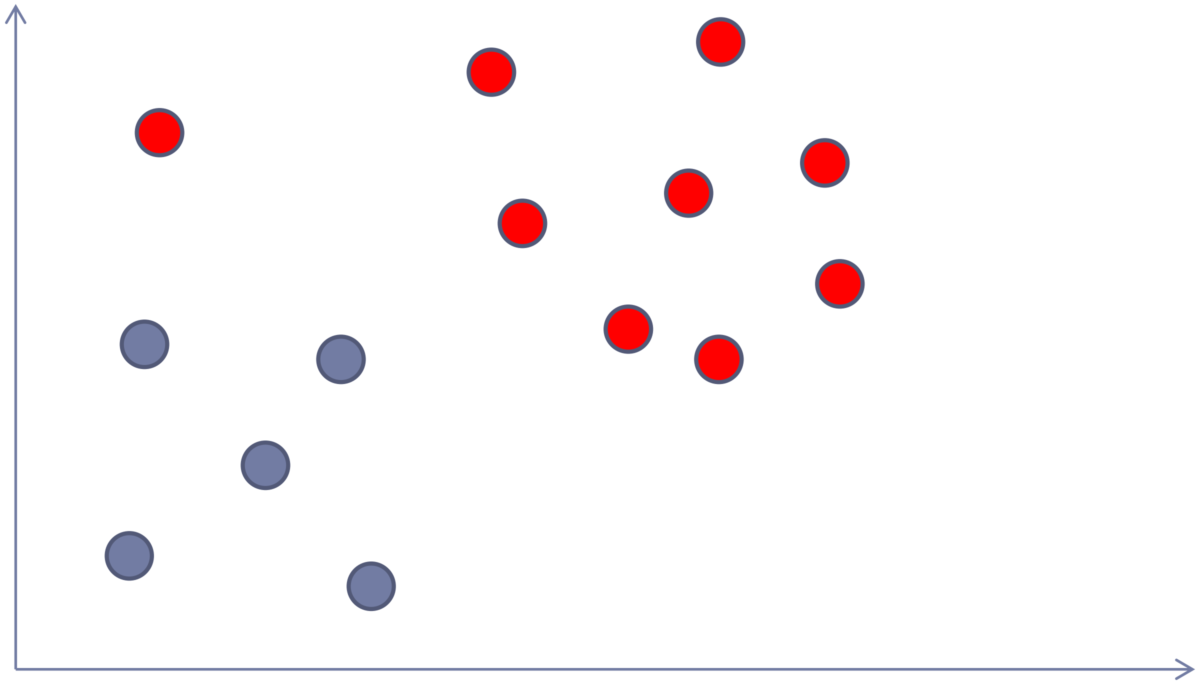
s.t

$y_i(w^T x_i + b) \geq 1$

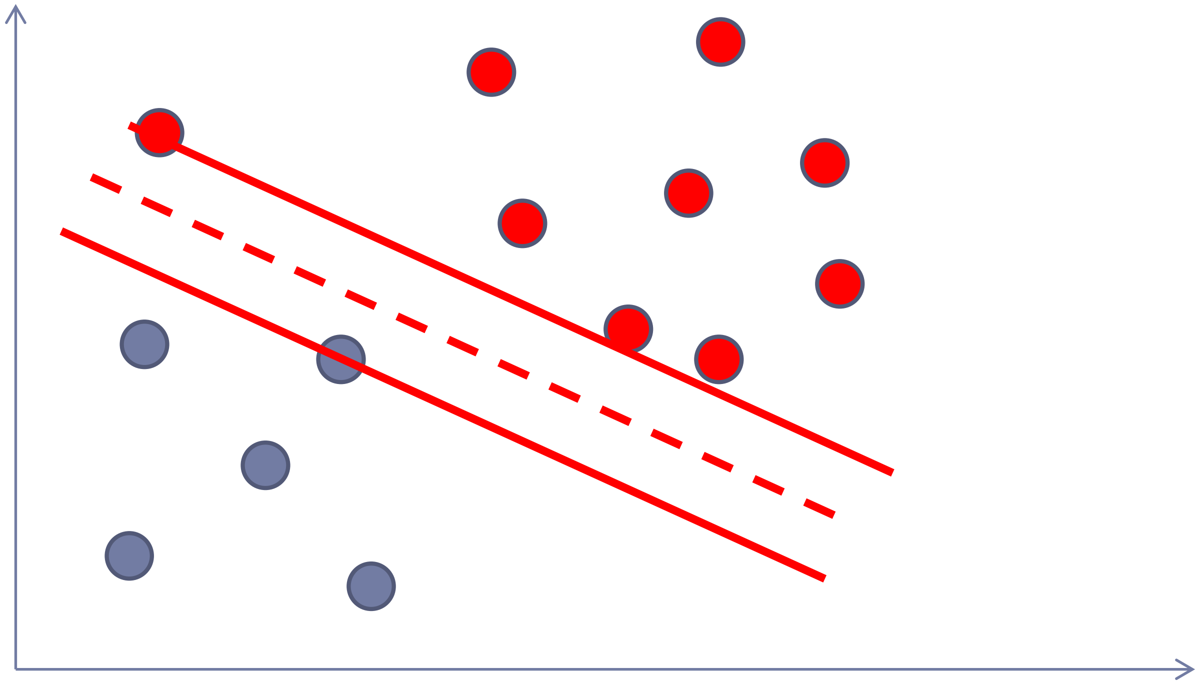
SVM



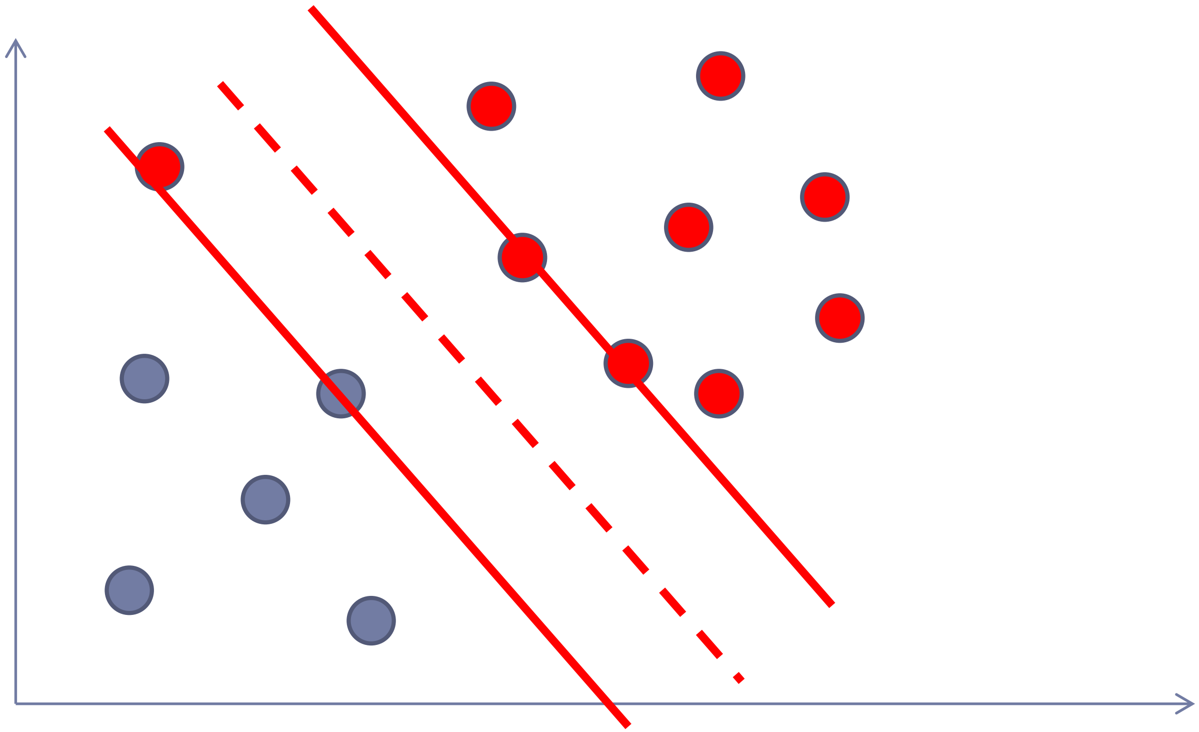
SVM



SVM

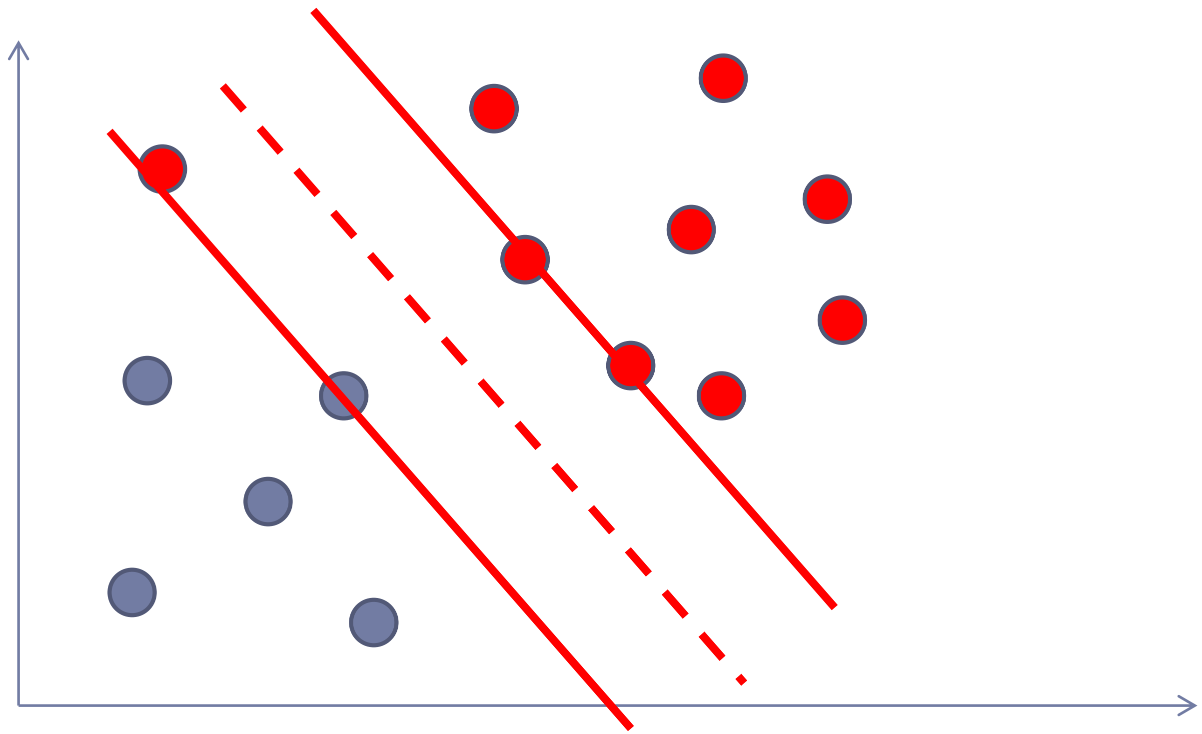


SVM



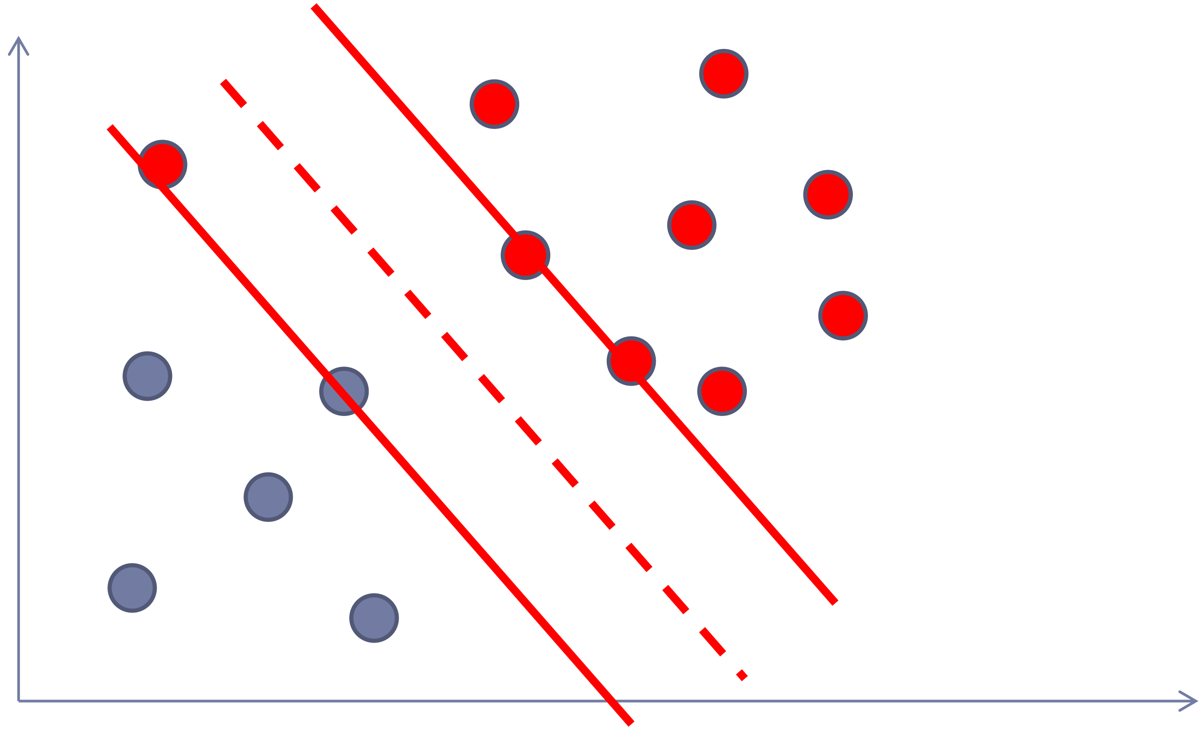
SVM

- Problema
- $\min \frac{1}{2} w^T w$
  s.t
  $y_i(w^T x_i + b) \geq 1$



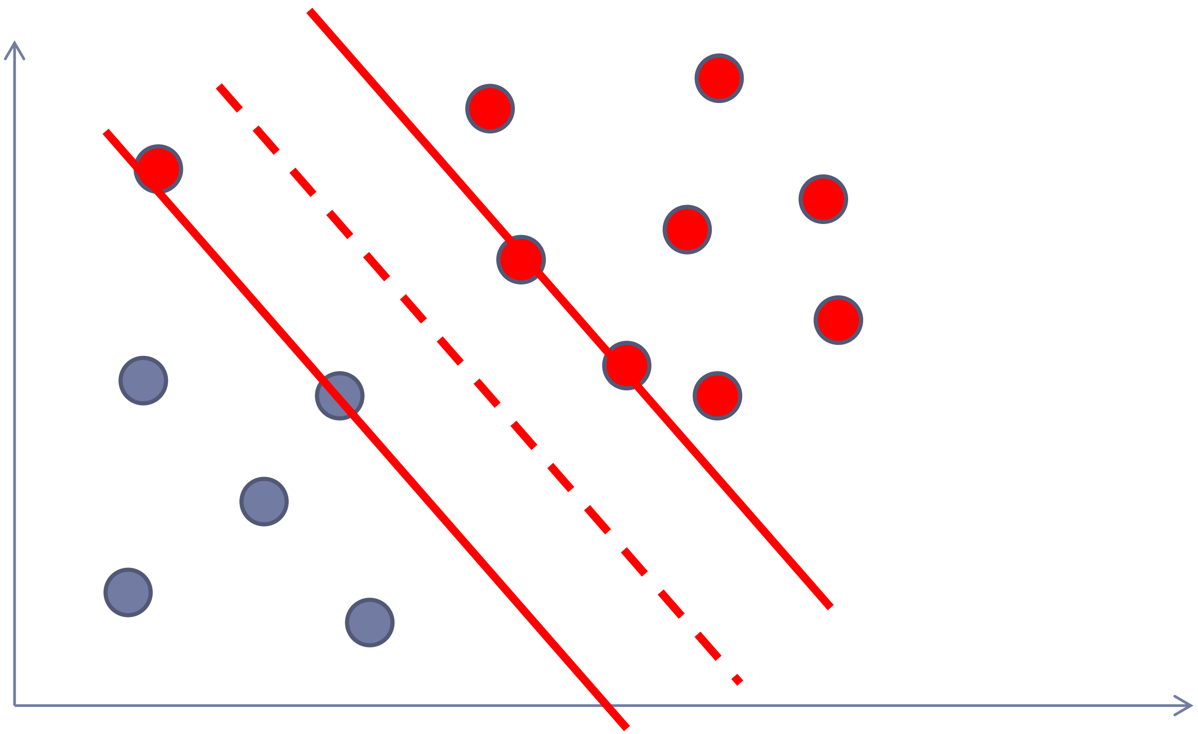
SVM

- Problema
- $\min \frac{1}{2} w^T w$
  s.t
  $y_i(w^T x_i + b) \geq 1 - \xi_i$



SVM (Soft Margin)

- Problema
- $\min \frac{1}{2} w^T w + C \sum_{i=1}^{n} \xi_i$
  s.t
  $y_i(w^T x_i + b) \geq 1 - \xi_i$



Support Vector Machines

- Parâmetro C
  - Seleciona o grau de ajuste do modelo aos dados de treinamento.
    - Overfitting

Support Vector Machines

- Até agora
  - SVM linear
  - Dados linearmente separáveis

- Versão não linear ?

Support Vector Machines

- Até agora
  - SVM linear
  - Dados linearmente separáveis

- Versão não linear ?
  - Kernel trick

Kernel Trick

- Como transformar um classificador linear em não linear ?

Kernel Trick

- Como transformar um classificador linear em não linear ?
- Criar novos atributos de forma não linear
  - $x = [x_1 \; x_2]$
    - $x_3 = {x_1}^2 + {x_2}^2$
  - $x = [x_1 \; x_2 \; x_3]$

Kernel Trick

- Usando SVM
  - $w = \sum_i \alpha_i y_i x_i$
  - $y_i(w^T x_i + b) \geq 1$
- $\overline{y_j} = g(w^T x_j + b)$
  - $\overline{y_i} = +1 \; ou \; -1$

Kernel Trick

- Usando SVM
  - $w = \sum_i \alpha_i y_i x_i$
  - $y_i(w^T x_i + b) \geq 1$
- $\overline{y_j} = (\sum_i \alpha_i y_i {x_i}^T x_j + b)$
  - $\overline{y_i} = +1 \; ou \; -1$

Aumentando a dimensão

- Exemplo
  - $x = [x_1 \; x_2]$

Aumentando a dimensão

- Exemplo
  - $x = [x_1 \; x_2]$
  - $x^a = \left[{x_1}^2 \quad \sqrt{2}x_1x_2 \quad {x_2}^2\right]$

Aumentando a dimensão

- Exemplo
  - $x = [x_1 \; x_2]$
  - $x^a = \left[{x_1}^2 \;\; \sqrt{2}x_1x_2 \;\; {x_2}^2 \right]$
- Do SVM
  - $\bar{y}_j = (\sum_i \; \alpha_i y_i {x_i}^T x_j + b)$

Aumentando a dimensão

- Exemplo
  - $x = [x_1 \; x_2]$
  - $x^a = \left[{x_1}^2 \;\; \sqrt{2} x_1 x_2 \;\; {x_2}^2 \right]$
- O produto interno entre dois vetores aumentados:
  - ${x^a}^T z^a = \left[{x_1}^2 \;\; \sqrt{2} x_1 x_2 \;\; {x_2}^2 \right] \left[{z_1}^2 \;\; \sqrt{2} z_1 z_2 \;\; {z_2}^2 \right]^T$

Aumentando a dimensão

- Exemplo
  - $x = [x_1 \; x_2]$
  - $x^a = \left[x_1{}^2 \;\; \sqrt{2}x_1 x_2 \;\; x_2{}^2\right]$
- O produto interno entre dois vetores aumentados:
  - $x^{a^T} z^a = \left[x_1{}^2 \;\; \sqrt{2}x_1 x_2 \;\; x_2{}^2\right]\left[z_1{}^2 \;\; \sqrt{2}z_1 z_2 \;\; z_2{}^2\right]^T$
  - $x^{a^T} z^a = x_1{}^2 z_1{}^2 + 2x_1 x_2 z_1 z_2 + x_2{}^2 z_2{}^2$

Aumentando a dimensão

- Exemplo
  - $x = [x_1 \; x_2]$
  - $x^a = \left[{x_1}^2 \;\; \sqrt{2}x_1x_2 \;\; {x_2}^2\right]$
- O produto interno entre dois vetores aumentados:
  - ${x^a}^T z^a = \left[{x_1}^2 \;\; \sqrt{2}x_1x_2 \;\; {x_2}^2\right]\left[{z_1}^2 \;\; \sqrt{2}z_1z_2 \;\; {z_2}^2\right]^T$
  - ${x^a}^T z^a = {x_1}^2{z_1}^2 + 2x_1x_2z_1z_2 + {x_2}^2{z_2}^2$
  - ${x^a}^T z^a = \{[x_1 \; x_2]^T[z_1 \; z_2]\}^2$

Aumentando a dimensão

- $x^{a^T} z^a = \{[x_1\ x_2]^T [z_1\ z_2]\}^2$
- $x^{a^T} z^a = f(x, z)$

Aumentando a dimensão

- ${x^a}^T z^a = \{[x_1\ x_2]^T [z_1\ z_2]\}^2$
- ${x^a}^T z^a = f(x, z)$

- *f* é uma função de Kernel
  - ${x^a}^T z^a = K(x, z)$

Aumentando a dimensão

- Estratégia convencional

Vetores originais → Vetores aumentados → Produto Interno → Resultado

Aumentando a dimensão

- Estratégia convencional

Vetores originais → Vetores aumentados → Produto Interno → Resultado

- Com Kernel

Vetores originais → Função Kernel dos vetores originais → Resultado

SVM com Kernel

- Usando SVM
- $\overline{y_j} = \sum_i \alpha_i y_i {x_i}^T x_j + b$
  - $\overline{y_i} = +1 \; ou \; -1$

SVM com Kernel

- Usando SVM
- $\overline{y_j} = \sum_i \alpha_i y_i K({x_i}^T x_j) + b$
  - $\overline{y_i} = +1 \, ou \, -1$

Funções Kernel

$K(\vec{x}_i, \vec{x}_j) = \vec{x}_i \cdot \vec{x}_j$ Linear kernel

$K(\vec{x}_i, \vec{x}_j) = \exp(-\gamma \|\vec{x}_i - \vec{x}_j\|^2)$ Gaussian kernel

$K(\vec{x}_i, \vec{x}_j) = \exp(-\gamma \|\vec{x}_i - \vec{x}_j\|)$ Exponential kernel

$K(\vec{x}_i, \vec{x}_j) = (p + \vec{x}_i \cdot \vec{x}_j)^q$ Polynomial kernel

$K(\vec{x}_i, \vec{x}_j) = (p + \vec{x}_i \cdot \vec{x}_j)^q \exp(-\gamma \|\vec{x}_i - \vec{x}_j\|^2)$ Hybrid kernel

$K(\vec{x}_i, \vec{x}_j) = \tanh(k\vec{x}_i \cdot \vec{x}_j - \delta)$ Sigmoidal

Funções Kernel mais Comuns

- $K(x,y) = \exp\left(\frac{(x-y)^T(x-y)}{\sigma^2}\right)$
- $K(x,y) = (p + x^T y)^q$

Support Vector Machines

- Parâmetro C
  - Seleciona o grau de ajuste do modelo aos dados de treinamento.
- Parâmetro do Kernel (σ^2, p,q ...)
  - Seleciona a complexidade da superfície de decisão

Support Vector Machines

- Seleção dos Hiper-parâmetros
  - Validação Cruzada

Support Vector Machines

- Seleção dos Hiper-parâmetros
  - Validação Cruzada
- Classificação Multiclasse
  - 1 contra todos

Dúvidas ?