Introdução às *Support Vector Machines*

Ana Carolina Lorena

Tópicos

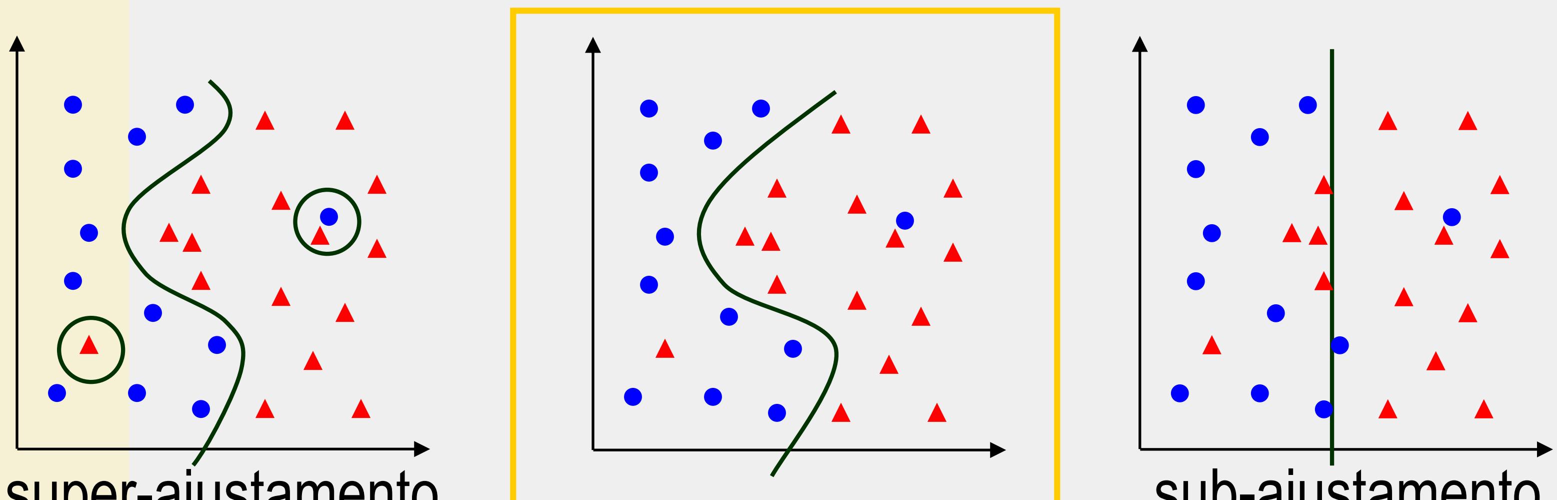
- Introdução
- Teoria de Aprendizado Estatístico
- SVMs lineares
- SVMs não lineares
- SVMs em problemas multiclasses
- Conclusões

Introdução

- Support Vector Machines (SVMs)
  - Máquinas de Vetores de Suporte
- Algoritmo de Aprendizado de Máquina
- Base na Teoria de Aprendizado Estatístico
- Bons resultados em diversos domínios

Aprendizado de Máquina

- Generalização de classificador
  - Capacidade de prever a classe de novos dados



super-ajustamento (*overfitting*)

sub-ajustamento (*underfitting*)

Complexidade intermediária e classifica corretamente grande parte dos dados de treinamento

Teoria de Aprendizado Estatístico

- Condições para a escolha de um classificador
  - Dados gerados de acordo com probabilidade $P(\mathbf{x},y)$
  - Risco (erro) esperado de classificador f

$$R(f) = \int c(f(\mathbf{x}), y)\, dP(\mathbf{x}, y)$$

    - Mede capacidade de generalização de f
    - Não pode ser minimizado diretamente

P(**x**,y) é desconhecida

Teoria de Aprendizado Estatístico

- Condições para a escolha de um classificador
  - Dados de treinamento também amostrados de $P(\mathbf{x},y)$
  - Risco (erro) empírico de classificador f

$$R_{emp}(f) = \frac{1}{n}\sum_{i=1}^{n} c(f(\mathbf{x}_i), y_i)$$

    - Mede desempenho no conjunto de treinamento

Procura-se *f* que minimize risco empírico

Teoria de Aprendizado Estatístico

- Condições para a escolha de um classificador
  - Nem sempre *f* de R_{emp} mínimo possui R mínimo
    - Classificador *f'* que memoriza dados de treinamento ...
    - ... e gera classificações aleatórias para outros dados

$R_{emp}(f') = 0$ . . . $R(f') = 0{,}5$

É desejável que *f'* possuia baixo R_{emp} e baixo R

Teoria de Aprendizado Estatístico

- Condições para a escolha de um classificador
  - *F* = conjunto de funções *f*

amplo → Maior possibilidade de achar *f'* com baixo R_{emp}
amplo → Maior possibilidade de super-ajustamentos

Restringir *F* ⇨ TAE controla a complexidade de *F*

reduzido → Menor possibilidade de achar *f'* com baixo R_{emp}
reduzido → Maior possibilidade de sub-ajustamentos

Teoria de Aprendizado Estatístico

- Limites em R

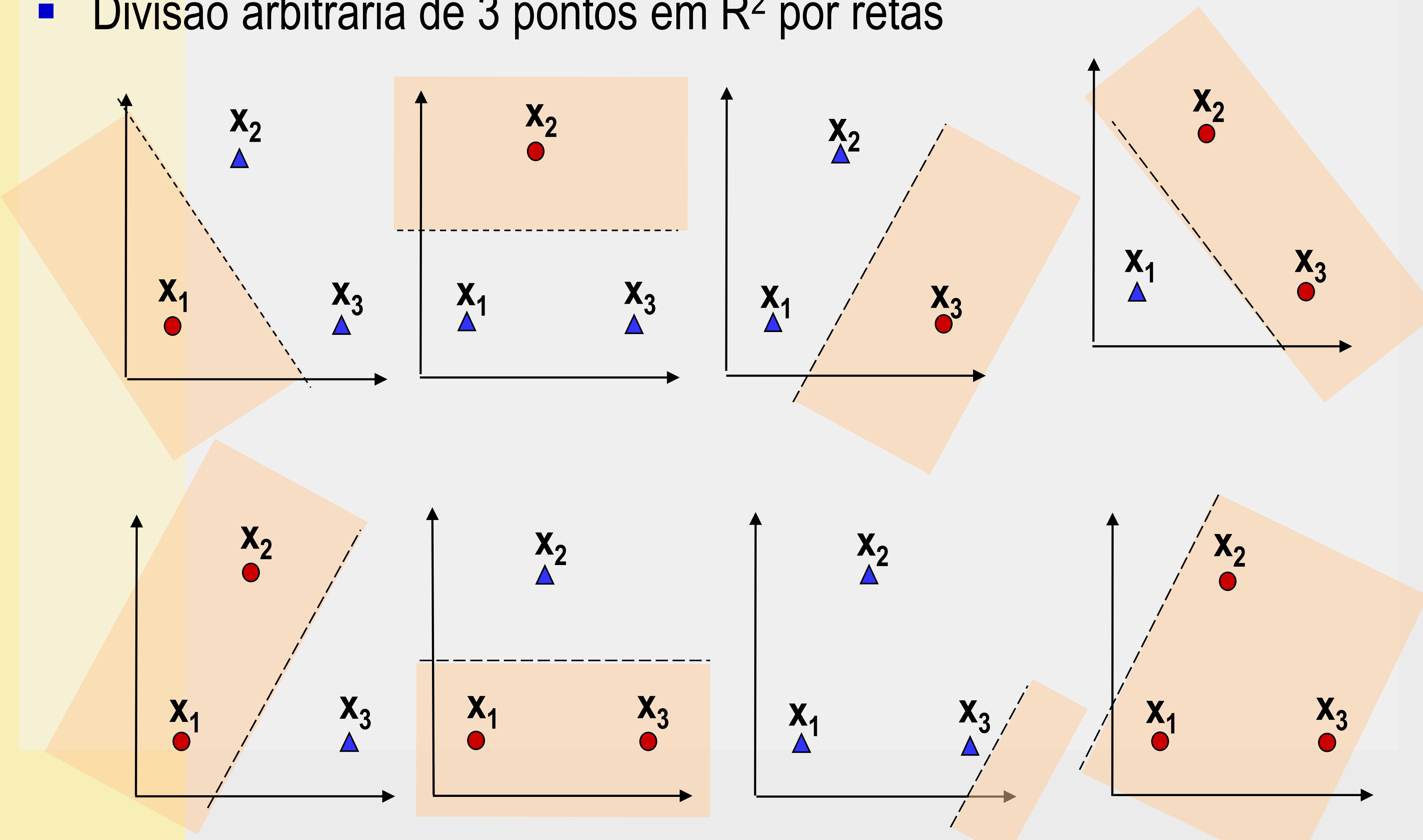
$$R(f) \leq R_{emp}(f) + \sqrt{\frac{h\left(\ln\left(\frac{2n}{h}\right)+1\right)-\ln\left(\frac{\theta}{4}\right)}{n}}$$

h = dimensão VC (Vapnik-Chernonenkis)

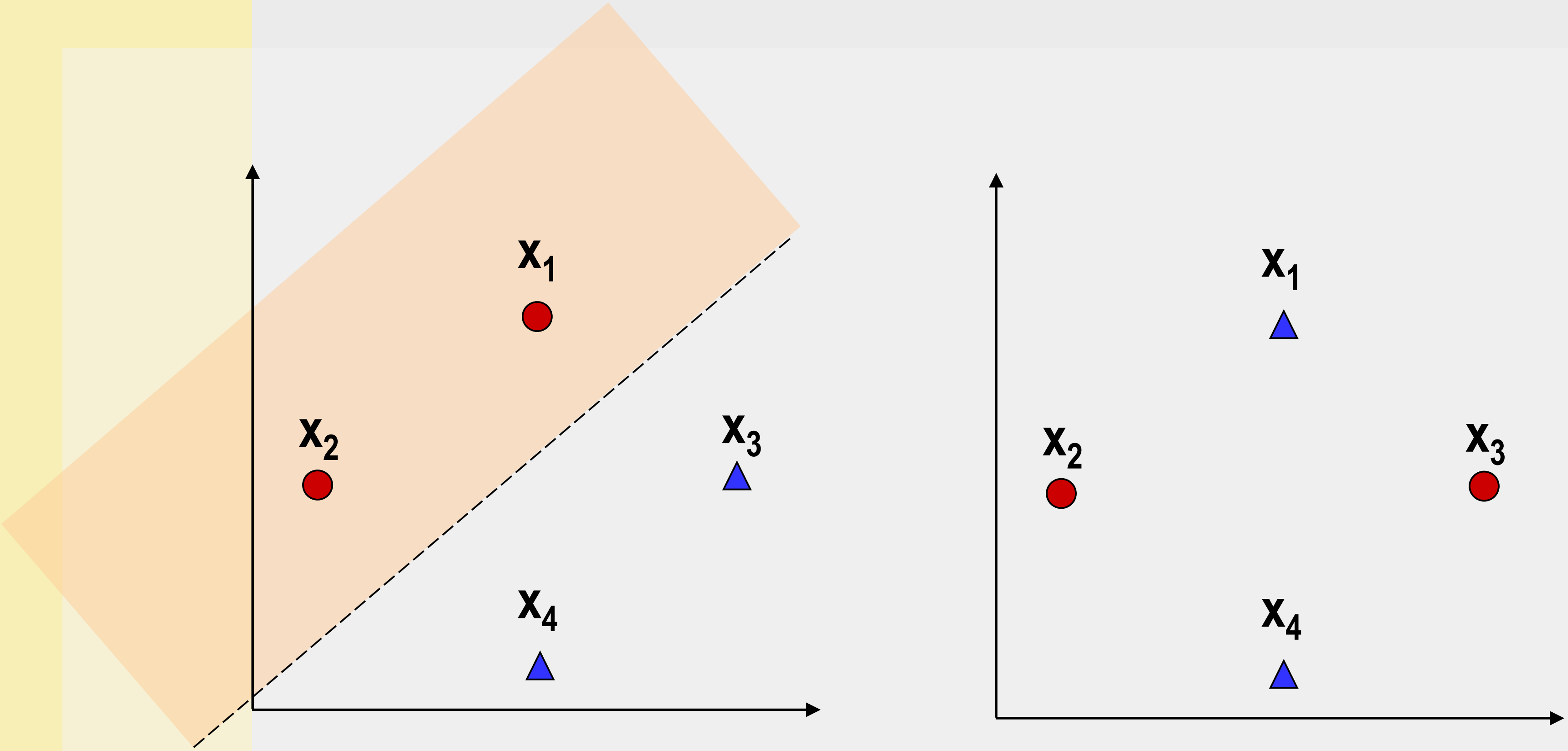
Maior $h \Rightarrow$ mais complexas as possíveis funções em F

Teoria de Aprendizado Estatístico

- Divisão arbitrária de 3 pontos em R^2 por retas



Teoria de Aprendizado Estatístico



- Divisão arbitrária de 4 pontos exige funções de maior complexidade

Dimensão VC de retas em $R^2 = 3$

Teoria de Aprendizado Estatístico

- Limites em *R*

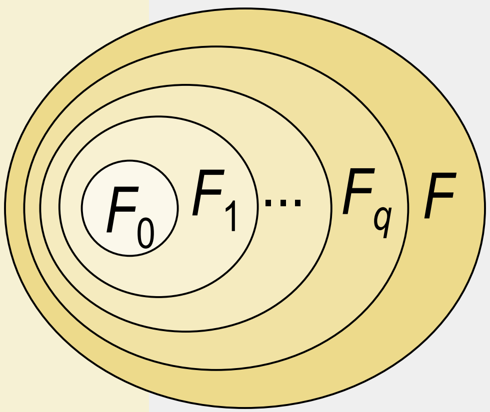
$$R(f) \leq R_{emp}(f) + \sqrt{\frac{h\left(\ln\left(\frac{2n}{h}\right)+1\right)-\ln\left(\frac{\theta}{4}\right)}{n}}$$

  - *f'* deve minimizar R_{emp} e pertencer a *F* com baixa dimensão VC

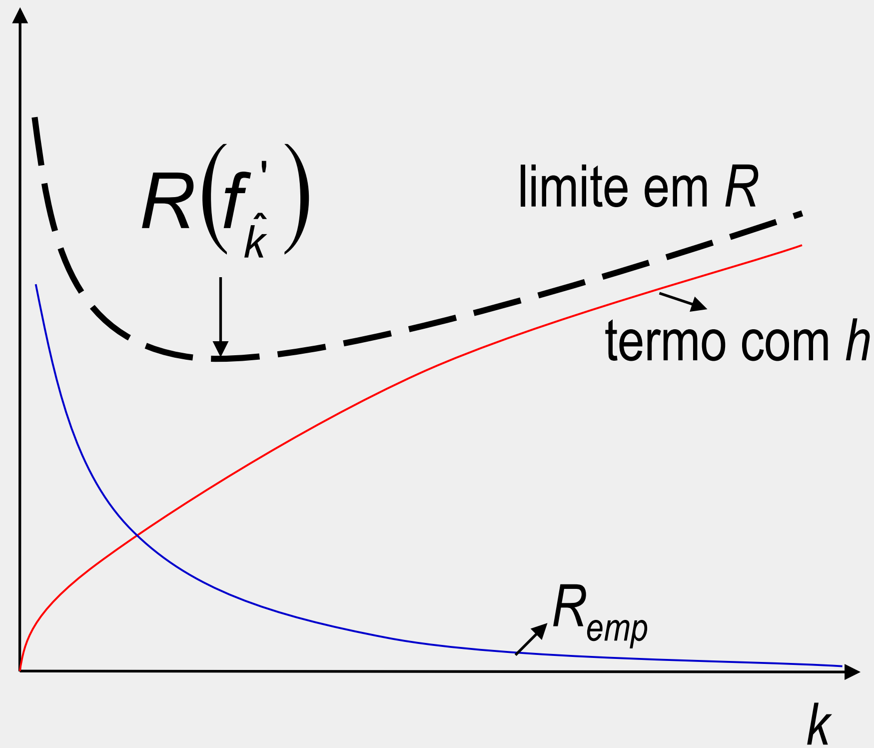
$R_{emp} \Rightarrow f$

$h \Rightarrow F$

Teoria de Aprendizado Estatístico



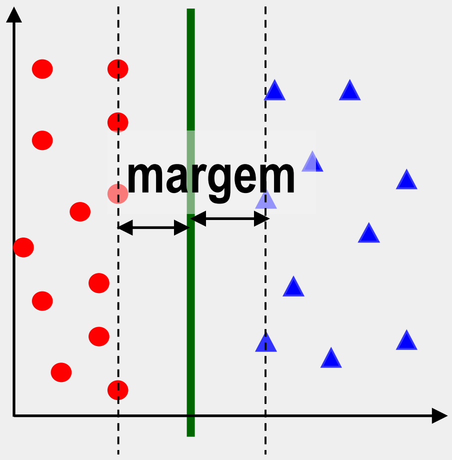
$h_0 < h_1 < \ldots < h_q < h$

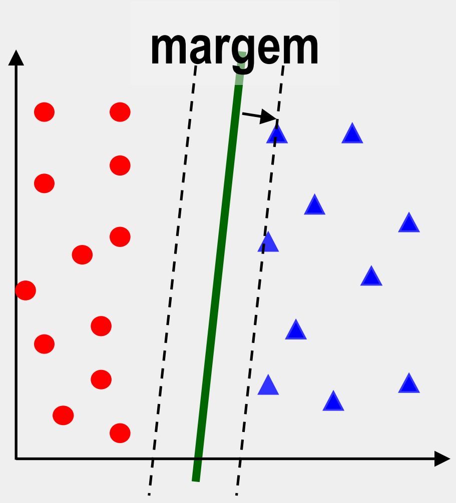


$\hat{k}$ em que se obtém *f'* com soma mínima de R_{emp} e termo com *h*

Teoria de Aprendizado Estatístico

- Computar *h* não é simples
  - *h* pode ser desconhecido ou infinito
- Limites alternativos para funções lineares





Teoria de Aprendizado Estatístico

- Limite para funções lineares

$$R(f) \le R_\rho(f) + \sqrt{\frac{c}{n}\left(\frac{R^2}{\rho^2}\log^2\left(\frac{n}{\rho}\right) + \log\left(\frac{1}{\theta}\right)\right)}$$

Erro marginal = proporção de dados de treinamento com margem < ρ

Maior ρ → Menor termo de capacidade; Maior R_ρ

Menor ρ → Maior termo de capacidade; Menor R_ρ

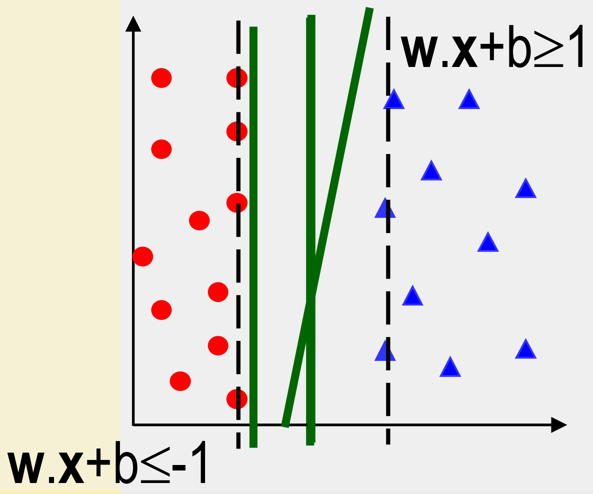
Compromisso entre maximização de ρ e minimização de R_ρ

SVMs lineares

- Funções do tipo

$$g(\mathbf{x}) = \mathrm{sgn}(f(\mathbf{x})) = \mathrm{sgn}(\mathbf{w}\cdot\mathbf{x}+b) = \begin{cases} +1 \; se \; \mathbf{w}\cdot\mathbf{x}+b>0 \\ -1 \; se \; \mathbf{w}\cdot\mathbf{x}+b<0 \end{cases}$$

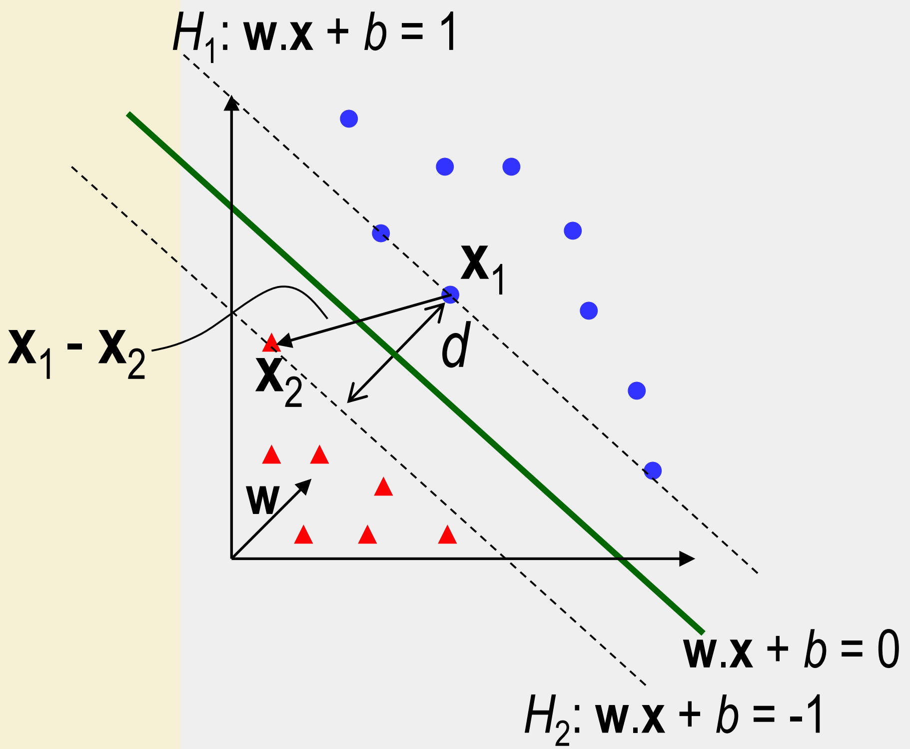
- Conjunto de dados linearmente separável



Hiperplano canônico em relação a T

$$\begin{cases} \mathbf{w}\cdot\mathbf{x}_i + b \geq +1 \;\; se \;\; y_i = +1 \\ \mathbf{w}\cdot\mathbf{x}_i + b \leq -1 \;\; se \;\; y_i = -1 \end{cases}$$

SVMs lineares



$$\begin{cases} \mathbf{w}\cdot\mathbf{x}_1 + b = +1 \\ \mathbf{w}\cdot\mathbf{x}_2 + b = -1 \end{cases} -$$

$$\overline{\mathbf{w}\cdot(\mathbf{x}_1 - \mathbf{x}_2) = 2}$$

Projeção $\mathbf{x}_1$-$\mathbf{x}_2$ na direção de **w**

$$(\mathbf{x}_1 - \mathbf{x}_2)\left(\frac{\mathbf{w}}{\|\mathbf{w}\|}\cdot\frac{(\mathbf{x}_1 - \mathbf{x}_2)}{\|\mathbf{x}_1 - \mathbf{x}_2\|}\right)$$

d = norma da projeção

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

margem = $\dfrac{1}{||\mathbf{w}||}$

SVMs lineares

- Maximização da margem de separação:

$$\text{Minimizar } \frac{1}{2}\|\mathbf{w}\|^2$$

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

Problema de otimização quadrático

- Lagrange:

$$L = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{n}\alpha_i\left(y_i(\mathbf{w}\cdot\mathbf{x}_i + b) - 1\right)$$

SVMs lineares

- Derivando em relação a **w**, b e igualando a 0:

$$\boxed{\sum_{i=1}^{n} \alpha_i y_i = 0} \qquad \boxed{\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i}$$

- Forma dual:

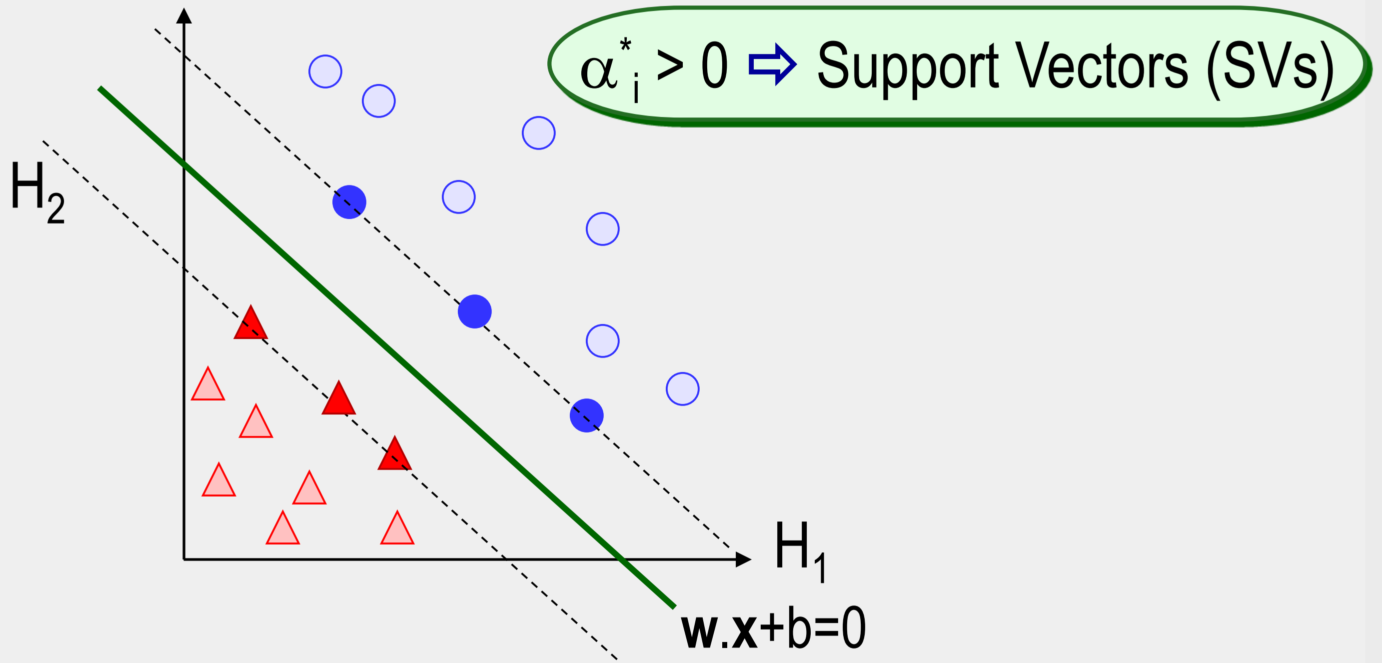
$$\text{Maximizar} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j$$

$$\text{Restrições} \begin{cases} \alpha_i \geq 0 \\ \sum_{i=1}^{n} \alpha_i y_i = 0 \end{cases}$$

SVMs lineares

- Condições de Kühn-Tucker:

$$\alpha_i^*\left(y_i\left(\mathbf{w}^*\cdot\mathbf{x}_i+b^*\right)-1\right)=0$$



SVMs lineares

- Solução:

$$b^* = \frac{1}{n_{SV}} \sum_{\mathbf{x}_j \in SV} \left(\frac{1}{y_j} - \sum_{\mathbf{x}_i \in SV} \alpha_i^* y_i \mathbf{x}_i \cdot \mathbf{x}_j \right)$$

$$\mathbf{w}^* = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$$

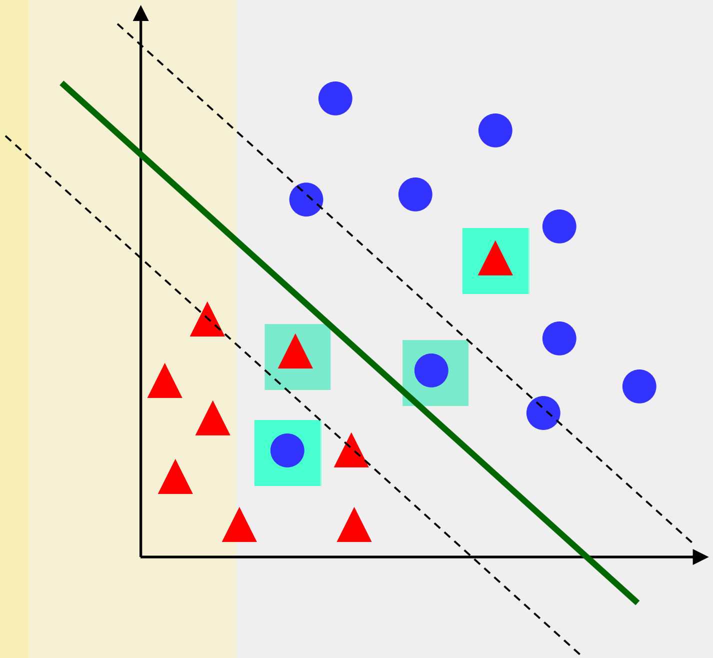
- Classificador:

$$g(\mathbf{x}) = \operatorname{sgn}(f(\mathbf{x})) = \operatorname{sgn}\left(\sum_{x_i \in SV} y_i \alpha_i^* \mathbf{x}_i \cdot \mathbf{x} + b^* \right)$$

SVMs lineares com margens rígidas

SVMs lineares

- Em geral conjuntos não são linearmente separáveis



- Introduz variáveis de folga ξ_i

Permite dados entre as margens e erros de treinamento

$$y_i\left(\mathbf{w}^* \cdot \mathbf{x}_i + b^*\right) \geq 1 - \xi_i, \quad \xi_i \geq 0$$

SVMs lineares com margens suaves

SVMs lineares

- Problema de otimização:

$$\text{Minimizar } \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i$$

$$\text{Restrições } \begin{cases} y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \geq 1 - \xi_i \\ \xi_i \geq 0,\ i = 1,\ldots,n \end{cases}$$

- Forma dual:

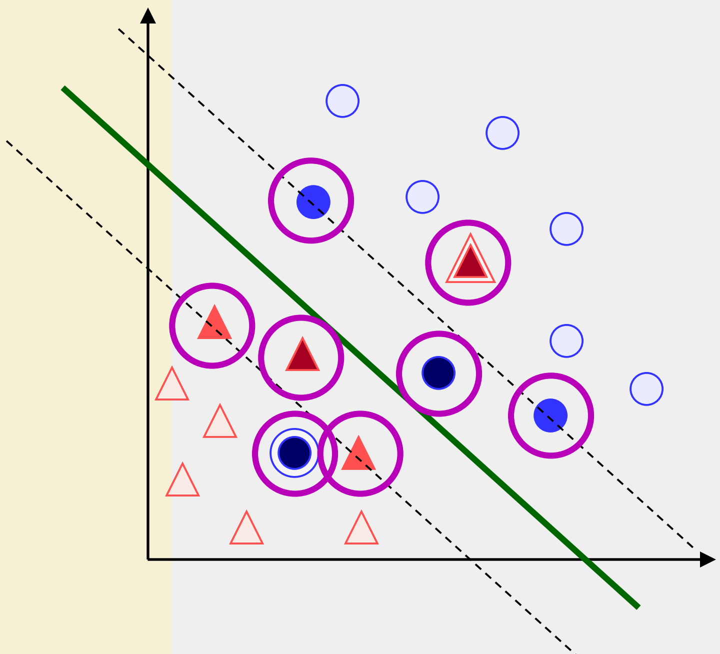
$$\text{Maximizar} \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j \mathbf{x}_i\cdot\mathbf{x}_j$$

$$\text{Restrições } \begin{cases} 0 \leq \alpha_i \leq C \\ \sum_{i=1}^{n}\alpha_i y_i = 0 \end{cases}$$

SVMs lineares

- Condições de Kühn-Tucker:

$$\boxed{\alpha_i^*\left(y_i\left(\mathbf{w}^* \cdot \mathbf{x}_i + b^*\right) - 1 + \xi_i^*\right) = 0} \qquad \boxed{\left(C - \alpha_i^*\right)\xi_i^* = 0}$$



$\alpha^*_i > 0$ ⇨ Support Vectors (SVs)

$\alpha^*_i < C \Rightarrow \xi^*_i = 0$ e sobre as margens

$$\alpha^*_i = C \begin{cases} \text{e } \xi^*_i > 1 \Rightarrow \text{erros} \\ \text{e } 0 < \xi^*_i \leq 1 \Rightarrow \text{entre as margens} \\ \cancel{\text{e } \xi^*_i = 0 \Rightarrow} \text{ raro } \cancel{\text{as margens}} \end{cases}$$

SVMs lineares

- Solução:

$$b^* = \frac{1}{n_{SV:\alpha^* < C}} \sum_{\mathbf{x}_j \in SV:\alpha_j^* < C} \left(\frac{1}{y_j} - \sum_{\mathbf{x}_i \in SV} \alpha_i^* y_i \mathbf{x}_i \cdot \mathbf{x}_j \right)$$

$$\mathbf{w}^* = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$$

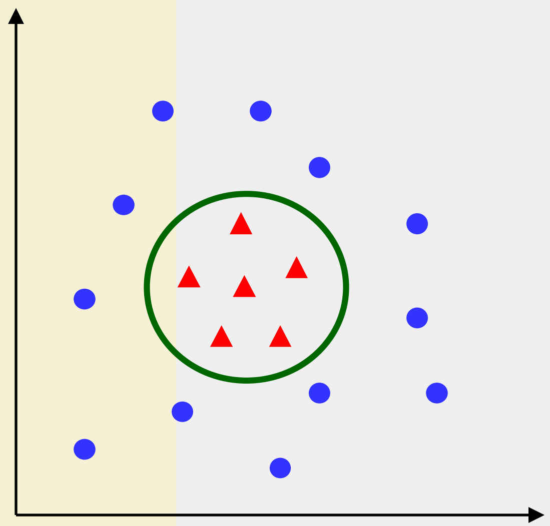
$$\xi_i^* = \max\left\{ 0{,}1 - y_i \sum_{j=1}^{n} y_j \alpha_j^* \mathbf{x}_j \cdot \mathbf{x}_i + b^* \right\}$$

- Classificador:

$$g(\mathbf{x}) = \text{sgn}(f(\mathbf{x})) = \text{sgn}\left(\sum_{x_i \in SV} y_i \alpha_i^* \mathbf{x}_i \cdot \mathbf{x} + b^* \right)$$

SVMs não lineares

- Muitos conjuntos de dados são não lineares



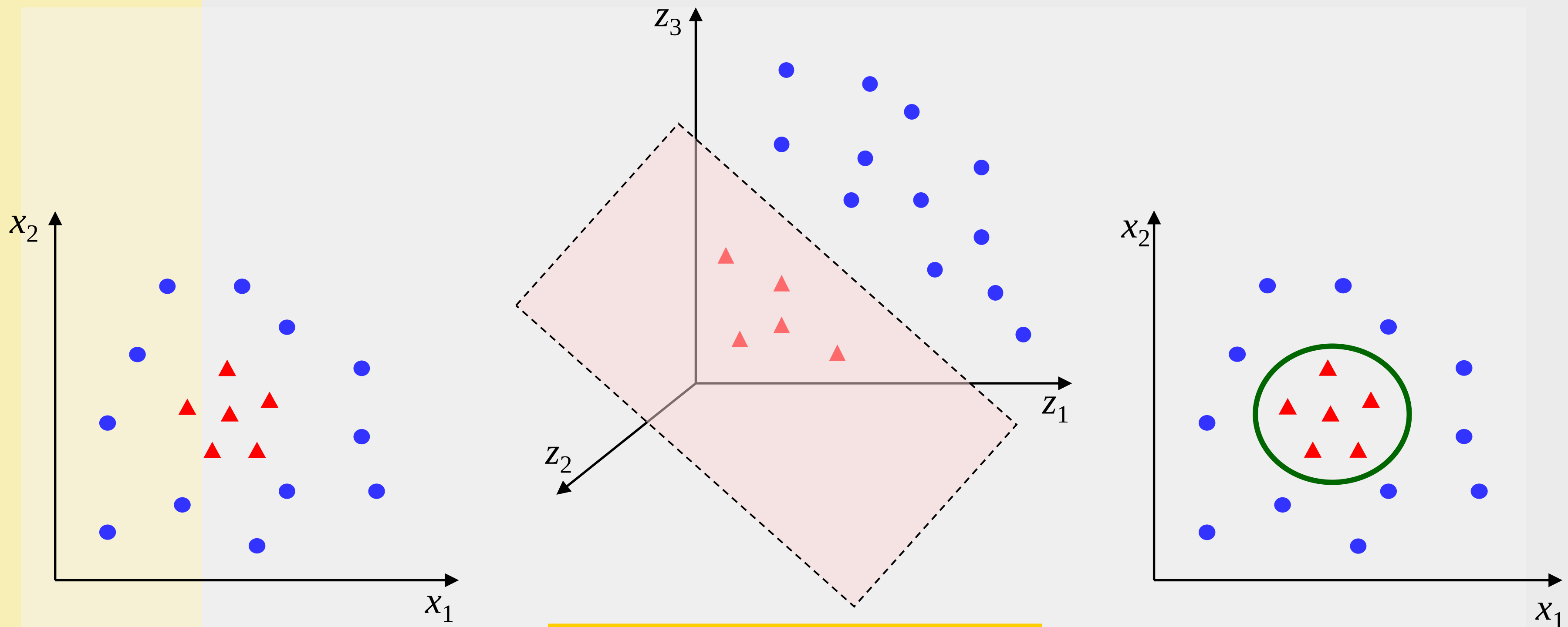
- Mapeia dados para espaço de maior dimensão
  - Teorema de Cover

Escolha apropriada de função de mapeamento

Dados podem ser separados por SVM linear

margens suaves

SVMs não lineares



$$\boldsymbol{\Phi}(\mathbf{x}) = \left(x_1^2, \sqrt{2}x_1x_2, x_2^2\right)$$

$$f(\mathbf{x}) = \mathbf{w}\cdot\mathbf{x} + b = w_1x_1^2 + w_2\sqrt{2}x_1x_2 + w_3x_2^2 + b$$

SVMs não lineares

- Problema de otimização dual:

$$\text{Maximizar} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{\Phi}(\mathbf{x}_i) \cdot \boldsymbol{\Phi}(\mathbf{x}_j)$$

$$\text{Restrições} \begin{cases} 0 \leq \alpha_i \leq C \\ \sum_{i=1}^{n} \alpha_i y_i = 0 \end{cases}$$

- Classificador:

$$g(\mathbf{x}) = \operatorname{sgn}\left(\sum_{x_i \in SV} y_i \alpha_i^* \boldsymbol{\Phi}(\mathbf{x}_i) \cdot \boldsymbol{\Phi}(\mathbf{x}) + b^* \right)$$

$$b^* = \frac{1}{n_{SV:\alpha^* < C}} \sum_{\mathbf{x}_j \in SV:\alpha_j^* < C} \left(\frac{1}{y_j} - \sum_{\mathbf{x}_i \in SV} \alpha_i^* y_i \boldsymbol{\Phi}(\mathbf{x}_i) \cdot \boldsymbol{\Phi}(\mathbf{x}_j) \right)$$

SVMs não lineares

- Computação de ϕ pode ser inviável
  - Produtos internos entre dados $\Rightarrow$ **Kernels**

No exemplo, $K(\mathbf{a},\mathbf{b})=(\mathbf{a}.\mathbf{b})^2$

$$\boxed{K(\mathbf{x}_i,\mathbf{x}_j)=\boldsymbol{\Phi}(\mathbf{x}_i)\cdot\boldsymbol{\Phi}(\mathbf{x}_j)}$$

• É comum empregar Kernel sem conhecer o mapeamento

Simplicidade de cálculo

Capacidade de representar espaços de grande dimensão

Devem seguir Teorema de Mercer

SVMs não lineares

- Principais tipos de Kernel

Tipo	**$K(\mathbf{x}_i,\mathbf{x}_j)$**	**Parâmetros**
Polinomial	$(\delta(\mathbf{x}_i.\mathbf{x}_j) + c)^d$	δ, c, d
Gaussiano (RBF)	$\exp(-\sigma\|\|\mathbf{x}_i - \mathbf{x}_j\|\|^2)$	σ
Sigmoidal	$\tanh(\delta(\mathbf{x}_i.\mathbf{x}_j) + c)$	δ, c

seleção de modelo → tipo de Kernel

seleção de modelo → parâmetros do Kernel

SVMs para problemas multiclasses

- SVMs: originalmente para problemas binários
  - Classes +1 e –1

Problemas multiclasses

Combinação de classificadores binários

Modificação no algoritmo de otimização

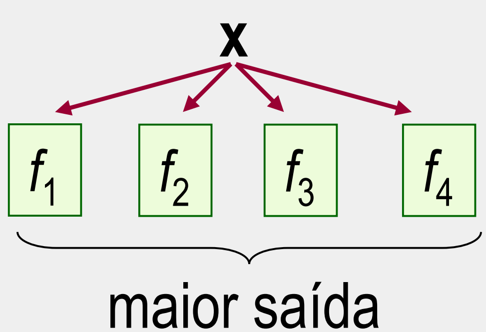
Algoritmos computacionalmente custosos

SVMs para problemas multiclasses

- Decomposições comuns:

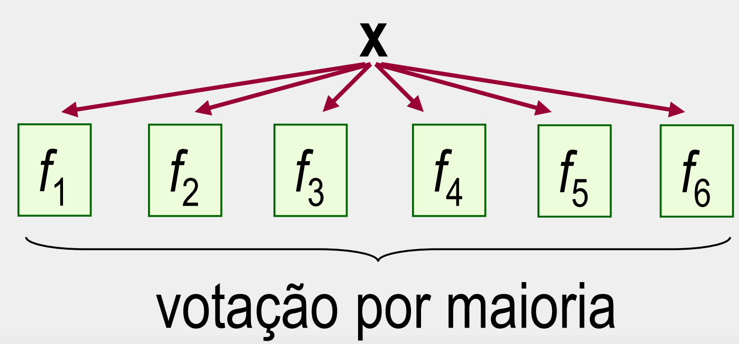
um-contra-todos

	f_1	f_2	f_3	f_4
classe A	+1	-1	-1	-1
classe B	-1	+1	-1	-1
classe C	-1	-1	+1	-1
classe D	-1	-1	-1	+1



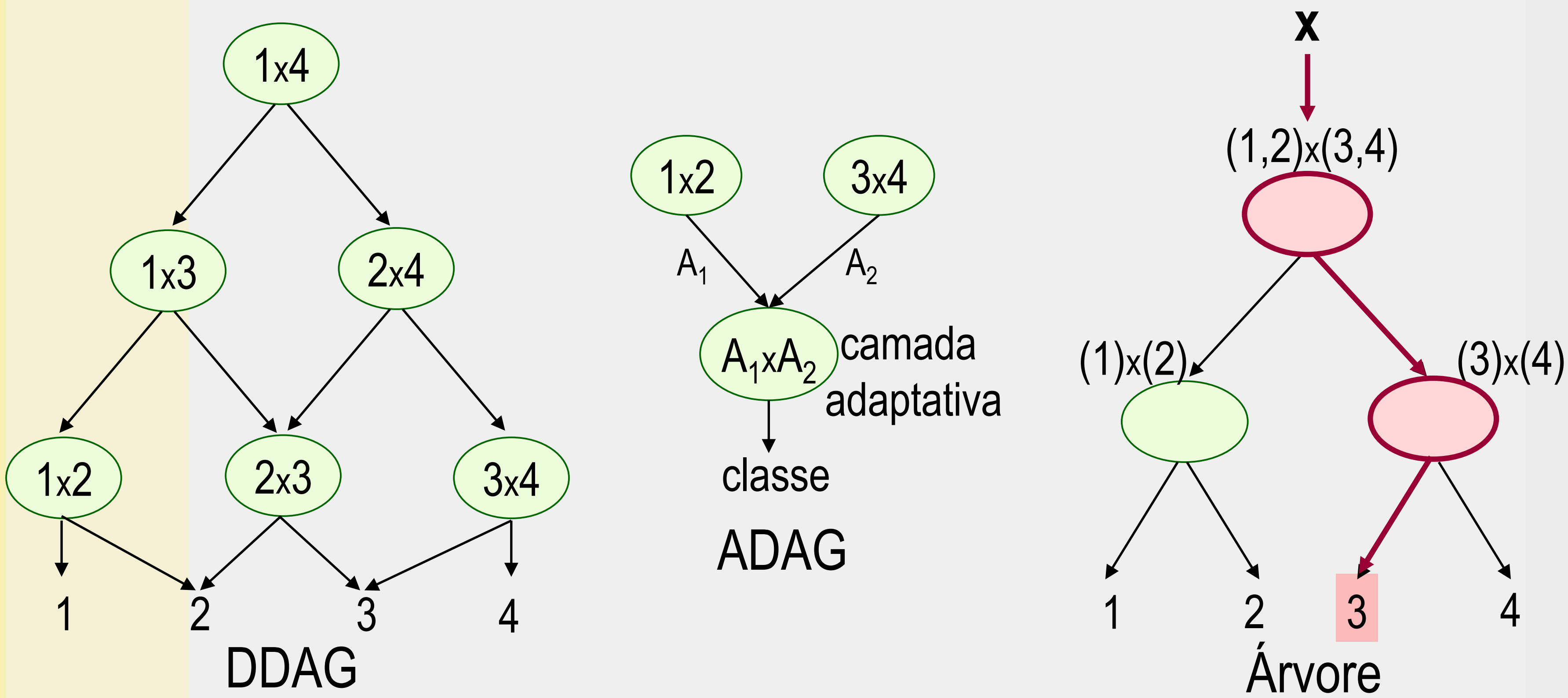
todos-contra-todos

	f_1	f_2	f_3	f_4	f_5	f_6
classe A	+1	+1	+1	0	0	0
classe B	-1	0	0	+1	+1	0
classe C	0	-1	0	-1	0	+1
classe D	0	0	-1	0	-1	-1



SVMs para problemas multiclasses

- Organizando hierarquicamente:



SVMs para problemas multiclasses

- Doutorado:

Adequar a cada problema

Combinação de classificadores binários

Estrutura de DDAGs e ADAGs

Estrutura de árvores

Algoritmos Genéticos

... do desempenho

Árvores Geradoras Mínimas

Decomposições mais simples ⇨ com menos classificadores

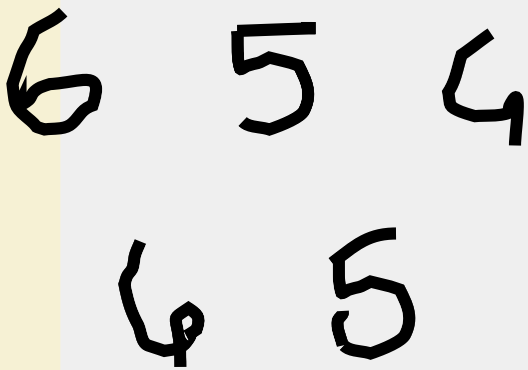
Exemplos de aplicações

- *Benchmarks*:

Conj dados	SVM	RBF
B. Cancer	**26.0 ± 0.47**	27.6 ± 0.47
Diabetes	**23.5 ± 0.17**	23.2 ± 0.16
German	**23.6 ± 0.21**	24.7 ± 0.24
Heart	**16.0 ± 0.33**	17.6 ± 0.33
Image	**3.0 ± 0.06**	3.3 ± 0.06
Ringnorm	**1.7 ± 0.01**	**1.7 ± 0.02**
F. Sonar	**32.4 ± 0.18**	34.4 ± 0.20
Splice	10.9 ± 0.07	**10.0 ± 0.10**
Titanic	**22.4 ± 0.10**	23.3 ± 0.13
Waveform	**9.9 ± 0.04**	10.8 ± 0.06

Exemplos de aplicações

- Reconhecimento de dígitos manuscritos:



Técnica	Taxa erro
K-NN	5.7%
RBF	4.2%
SVM	4.0%
Virtual SVM	3.0%
Boosting	2.6%
Tangent Distance	2.5%
Humano	2.5%

Exemplos de aplicações

- Bioinformática:

  Sítios de início de tradução em cadeias de DNA

Técnica	Taxa erro
Rede Neural	15,4%
Salzberg	13,8%
SVM, Kernel polinomial	13,2%
SVM, Kernel codon	12.2%
SVM, Kernel Salzberg	11.4%

Kernel modificado
Conhecimento a priori

Exemplos de aplicações

- Outras aplicações de sucesso:
  - Categorização de textos
  - Reconhecimento de faces
  - Bioinformática
    - Dobras de proteínas
    - Localização de proteínas
    - Classificação de dados de expressão gênica

Conclusão

- Vantagens das SVMs:

Boa capacidade de generalização

Convexidade da função objetivo

Robustez em grandes dimensões

Teoria bem definida na Matemática e Estatística

Conclusão

- Desvantagens das SVMs:

Velocidade de classificação

Complexidade computacional

Implementação não é simples

Conhecimento não é facilmente interpretável

Conclusão

- Referências:
  - K.-R. Müller et al. (2001). An introduction to Kernel-based learning algorithms. IEEE Transactions on Neural Networks, Vol.12, N. 2, p. 181-202
  - N. Cristianini and J. Shawe-Taylor (2001). An introduction to Support Vector Machines. Cambridge University Press
  - A. J. Smola and B. Schölkopf (2002). Learning with Kernels. The MIT Press
  - www.kernel-machines.org