Aprendizado de Máquina

Aula 9

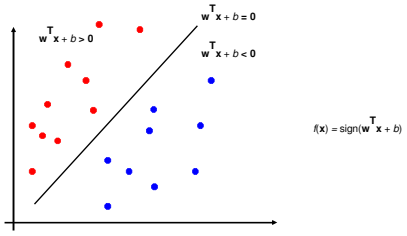
http://www.ic.uff.br/~bianca/aa/

Tópicos

1. Introdução – Cap. 1 (16/03)
2. Classificação Indutiva – Cap. 2 (23/03)
3. Árvores de Decisão – Cap. 3 (30/03)
4. Ensembles - Artigo (13/04)
5. Avaliação Experimental – Cap. 5 (20/04)
6. Aprendizado de Regras – Cap. 10 (27/04)
7. Redes Neurais – Cap. 4 (04/05)
8. Teoria do Aprendizado – Cap. 7 (11/05)
9. **Máquinas de Vetor de Suporte – Artigo (18/05)**
10. Aprendizado Bayesiano – Cap. 6 e novo cap. online (25/05)
11. Aprendizado Baseado em Instâncias – Cap. 8 (01/05)
12. Classificação de Textos – Artigo (08/06)
13. Aprendizado Não-Supervisionado – Artigo (15/06)

Revisão de Perceptrons: Separadores Lineares

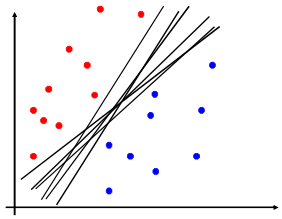
- Classificação binária pode ser vista como a tarefa de separar as classes no espaço de características:



$f(\mathbf{x}) = \text{sign}(\mathbf{w}^T\mathbf{x}+b)$

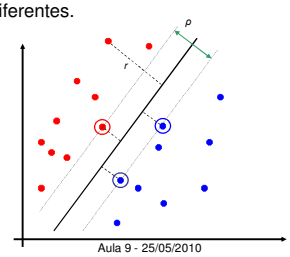
Separadores Lineares

- Qual destes separadores é ótimo?



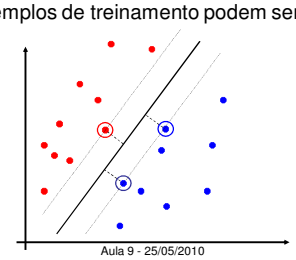
Margem de Classificação

- Distância do exemplo $\mathbf{x}_i$ ao separador é $r = \frac{\mathbf{w}^T\mathbf{x}_i + b}{\|\mathbf{w}\|}$
- Exemplos mais próximos ao hiperplano são ***vetores de suporte***.
- ***Margem*** ρ do separador é a distância entre vetores de suporte de classes diferentes.



Classificação com Máxima Margem

- Maximizar a margem é bom de acordo com a intuição e com a teoria PAC.
- Implica que só os vetores de suporte são importantes; outros exemplos de treinamento podem ser ignorados.



SVM Linear Matematicamente

- Seja um conjunto de treinamento $\{(\mathbf{x}_i, y_i)\}_{i=1..n}$, $\mathbf{x}_i \in \mathbf{R}^d$, $y_i \in \{-1, 1\}$ separável por um hiperplano com margem ρ. Então para cada exemplo de treinamento $(\mathbf{x}_i, y_i)$:

$$\mathbf{w}^T\mathbf{x}_i + b \leq -\rho/2 \quad \text{if } y_i = -1$$
$$\mathbf{w}^T\mathbf{x}_i + b \geq \rho/2 \quad \text{if } y_i = 1 \quad \Leftrightarrow \quad y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq \rho/2$$

- Para cada vetor de suporte $\mathbf{x}_s$ a desigualdade acima é uma igualdade. Depois de dividir $\mathbf{w}$ e b por $\rho/2$ na igualdade, temos que a distância entre cada $\mathbf{x}_s$ e o hiperplano é

$$r = \frac{y_s(\mathbf{w}^T\mathbf{x}_s + b)}{\|\mathbf{w}\|} = \frac{1}{\|\mathbf{w}\|}$$

- A margem pode então ser escrita como:

$$\rho = 2r = \frac{2}{\|\mathbf{w}\|}$$

SVM Linear Matematicamente

- Então podemos formular o *problema de otimização quadrática:*

Encontrar $\mathbf{w}$ e b tais que
$\rho = \frac{2}{\|\mathbf{w}\|}$ seja maximizado
e para todo $(\mathbf{x}_i, y_i)$, $i=1..n$: $y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1$

que pode ser reformulado como:

Encontrar $\mathbf{w}$ e b tais que
$\Phi(\mathbf{w}) = \|\mathbf{w}\|^2 = \mathbf{w}^T\mathbf{w}$ seja minimizado
e para todo $(\mathbf{x}_i, y_i)$, $i=1..n$: $y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1$

Resolvendo o problema de otimização

Encontrar $\mathbf{w}$ e b tais que
$\Phi(\mathbf{w}) = \mathbf{w}^T\mathbf{w}$ seja minimizado
e para todo $(\mathbf{x}_i, y_i)$, $i=1..n$: $y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1$

- Precisamos otimizar uma função quadrática sujeita a restrições lineares.
- Problemas de otimização quadrática são uma classe conhecida de problemas de programação quadrática para a qual existem muitos algoritmos (não-triviais).
- A solução envolve construir um *problema dual* onde um *multiplicador de Lagrange* α_i está associado com cada restrição de desigualdade no problema primal (original):

Encontrar $\alpha_1 \ldots \alpha_n$ tais que
$\mathbf{Q}(\boldsymbol{\alpha}) = \Sigma\alpha_i - \frac{1}{2}\Sigma\Sigma\alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j$ seja maximizado e
(1) $\Sigma\alpha_i y_i = 0$
(2) $\alpha_i \geq 0$ para todo α_i

A solução do problema de otimização

- Dada uma solução $\alpha_1 \ldots \alpha_n$ para o problema dual, a solução do primal é:

$$\mathbf{w} = \Sigma\alpha_i y_i \mathbf{x}_i \qquad b = y_k - \Sigma\alpha_i y_i \mathbf{x}_i^T\mathbf{x}_k \quad \text{para qualquer } \alpha_k > 0$$

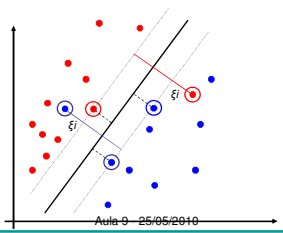
- Cada α_i diferente de zero indica que o $\mathbf{x}_i$ correspondente é um vetor de suporte.
- Então a função de classificação é (note que não precisamos de $\mathbf{w}$ explicitamente):

$$f(\mathbf{x}) = \Sigma\alpha_i y_i \mathbf{x}_i^T\mathbf{x} + b$$

- Note que a função depende de um produto interno entre o vetor de suporte e $\mathbf{x}_i$ – voltaremos a isso depois.
- Note também que resolver o problema de otimização envolve calcular produtos internos $\mathbf{x}_i^T\mathbf{x}_j$ entre todos os exemplos de treinamento.

Classificação "Soft margin"

- E se o conjunto de treinamento não for linearmente separável?
- Variáveis "*slack*" ξ_i podem ser adicionadas para tornar permitir o erro de exemplos difíceis ou ruidosos, resultando em uma margem chamada de "*soft*".



Margem "Soft" Matematicamente

- Formulação anterior:

Encontrar $\mathbf{w}$ e b tais que
$\Phi(\mathbf{w}) = \mathbf{w}^T\mathbf{w}$ seja minimizado
e para todo $(\mathbf{x}_i, y_i)$, $i=1..n$: $y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1$

- Nova formulação:

Encontrar $\mathbf{w}$ e b tais que
$\Phi(\mathbf{w}) = \mathbf{w}^T\mathbf{w} + C\Sigma\xi_i$ seja minimizado
e para todo $(\mathbf{x}_i, y_i)$, $i=1..n$: $y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i$, $\xi_i \geq 0$

- Parâmetro C pode ser visto como uma maneira de controlar o sobre-ajuste: controla a importância relativa de maximizar a margem e se ajustar aos dados de treinamento.

Margem "Soft" Matematicamente

- Problema dual é idêntico ao separável:

  Encontrar $\alpha_1 \ldots \alpha_n$ tais que
  $\mathbf{Q}(\boldsymbol{\alpha}) = \Sigma\alpha_i - \frac{1}{2}\Sigma\Sigma\alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j$ seja maximizado e
  (1) $\Sigma\alpha_i y_i = 0$
  (2) $0 \le \alpha_i \le C$ para todo α_i

- De novo, $\mathbf{x}_i$ com α_i diferente de zero serão vetores de suporte.
- Solução do problema dual é:

  $\mathbf{w} = \Sigma\alpha_i y_i \mathbf{x}_i$
  $b = y_k(1 - \xi k) - \Sigma\alpha_i y_i \mathbf{x}_i^T\mathbf{x}k$ para qualquer k tal que $\alpha k > 0$

- E função de classificação é:

  $f(\mathbf{x}) = \Sigma\alpha_i y_i \mathbf{x}_i^T\mathbf{x} + b$

Justificativa Teórica para Máxima Margem

- Vapnik provou o seguinte:
  A classe de separadores ótimos lineares tem dimensão VC h limitada por:

$$h \le \min\left\{\left\lceil \frac{D^2}{\rho^2} \right\rceil, m_0\right\} + 1$$

  onde ρ é a margem, D é o diâmetro da menor esfera que cobre todos os exemplos de treinamento, e m_0 é a dimensionalidade (número de atributos).
- Intuitivamente, isto implica que não importando a dimensionalidade m_0 podemos minimizar a dimensão VC maximizando a margem ρ.
- Logo a complexidade do classificador fica pequena independentemente da dimensionalidade.

SVM Linear: Resumo

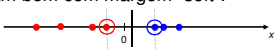
- O classificador é um *hiperplano separador.*
- Exemplos de treinamento mais "importantes" são os vetores de suporte; eles de finem o hiperplano.
- Algoritmos de otimização quadrática podem identificar que exemplos de treinamento $\mathbf{x}_i$ são vetores de suporte com multiplicadores de Lagrange α_i diferentes de zero.
- Tanto na formulação dual do problema quanto na solução, exemplos de treinamento aparecem apenas dentro de produtos internos:

  Encontrar $\alpha_1 \ldots \alpha_n$ tais que
  $\mathbf{Q}(\boldsymbol{\alpha}) = \Sigma\alpha_i - \frac{1}{2}\Sigma\Sigma\alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j$ seja maximizado e
  (1) $\Sigma\alpha_i y_i = 0$
  (2) $0 \le \alpha_i \le C$ para todo α_i

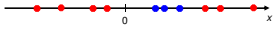
  $f(\mathbf{x}) = \Sigma\alpha_i y_i \mathbf{x}_i^T\mathbf{x} + b$

SVM Não-Linear

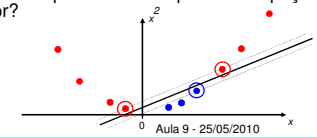
- Dados que são linearmente separáveis com algum ruído funcionam bem com margem "soft":



- Mas o que fazer se este não é o caso?

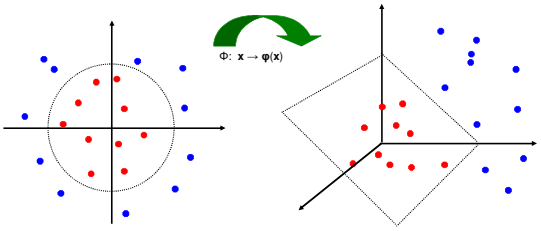


- Que tal mapear os dados para um espaço de dimensão maior?



SVM Não Linear

- Ideia geral: o espaço de atributos original pode ser mapeado em um espaço de atributos de dimensão maior onde o conjunto de treinamento seja linearmente separável:



O "Truque" do Kernel

- O classificador linear depende do produto interno entre exemplos $K(\mathbf{x}_i,\mathbf{x}_j) = \mathbf{x}_i^T\mathbf{x}_j$
- Se cada ponto for mapeado a um espaço de alta dimensão através de uma transformação Φ: $\mathbf{x} \rightarrow \varphi(\mathbf{x})$, o produto interno fica:

  $K(\mathbf{x}_i,\mathbf{x}_j) = \boldsymbol{\varphi}(\mathbf{x}_i)^T\boldsymbol{\varphi}(\mathbf{x}_j)$
- Uma função de kernel é uma função que é equivalente a um produto interno nem espaço de maior de dimensionalidade.
- Example:
  Vetores de 2 dimensões $\mathbf{x} = [x_1\ x_2]$; let $K(\mathbf{x}_i,\mathbf{x}_j) = (1 + \mathbf{x}_i^T\mathbf{x}_j)^2$,
  Precisamos mostrar que $K(\mathbf{x}_i,\mathbf{x}_j) = \boldsymbol{\varphi}(\mathbf{x}_i)^T\boldsymbol{\varphi}(\mathbf{x}_j)$:
  $K(\mathbf{x}_i,\mathbf{x}_j) = (1 + \mathbf{x}_i^T\mathbf{x}_j)^2 = 1 + x_{i1}^2x_{j1}^2 + 2\,x_{i1}x_{j1}\,x_{i2}x_{j2} + x_{i2}^2x_{j2}^2 + 2x_{i1}x_{j1} + 2x_{i2}x_{j2}$
  $= [1\ \ x_{i1}^2\ \ \sqrt{2}\,x_{i1}x_{i2}\ \ x_{i2}^2\ \ \sqrt{2}x_{i1}\ \ \sqrt{2}x_{i2}]^T\,[1\ \ x_{j1}^2\ \ \sqrt{2}\,x_{j1}x_{j2}\ \ x_{j2}^2\ \ \sqrt{2}x_{j1}\ \ \sqrt{2}x_{j2}]$
  $= \boldsymbol{\varphi}(\mathbf{x}_i)^T\boldsymbol{\varphi}(\mathbf{x}_j)$, onde $\boldsymbol{\varphi}(\mathbf{x}) = [1\ \ x_1^2\ \ \sqrt{2}\,x_1x_2\ \ x_2^2\ \ \sqrt{2}x_1\ \ \sqrt{2}x_2]$
- Não precisamos calcular $\boldsymbol{\varphi}(\mathbf{x})$ explicitamente.

Que funções são *Kernels*?

- Para algumas funções $K(\mathbf{x}_i,\mathbf{x}_j)$ checar que $K(\mathbf{x}_i,\mathbf{x}_j)= \varphi(\mathbf{x}_i)^T\varphi(\mathbf{x}_j)$ pode ser difícil.
- Teorema de Mercer:

Toda função simétrica definida semi-positiva é um kernel

Exemplos de Funções *Kernel*

- Linear: $K(\mathbf{x}_i,\mathbf{x}_j)= \mathbf{x}_i^T\mathbf{x}_j$
  - Φ: $\mathbf{x} \rightarrow \boldsymbol{\varphi}(\mathbf{x})$, onde $\boldsymbol{\varphi}(\mathbf{x})=\mathbf{x}$
- Polinomial de expoente p: $K(\mathbf{x}_i,\mathbf{x}_j)= (1+ \mathbf{x}_i^T\mathbf{x}_j)^p$
  - Φ: $\mathbf{x} \rightarrow \boldsymbol{\varphi}(\mathbf{x})$, onde $\boldsymbol{\varphi}(\mathbf{x})$ tem $\binom{d+p}{p}$ dimensões
- Gaussiana (RBF): $K(\mathbf{x}_i,\mathbf{x}_j) = e^{\frac{\|\mathbf{x}_i-\mathbf{x}_j\|^2}{2\sigma^2}}$
  - Φ: $\mathbf{x} \rightarrow \boldsymbol{\varphi}(\mathbf{x})$, onde $\boldsymbol{\varphi}(\mathbf{x})$ tem dimensão infinita: cada ponto é mapeado para uma função gaussiana.
- Separadores lineares correspondem a separadores não lineares no espaço de dimensão maior

SVMs Não Lineares Matematicamente

- Formulação do problema dual:

Encontrar $\alpha_1 \ldots \alpha_n$ tais que
$\mathbf{Q}(\boldsymbol{\alpha}) = \Sigma\alpha_i - \frac{1}{2}\Sigma\Sigma\alpha_i\alpha_j y_i y_j \mathbf{K}(\mathbf{x}_i,\mathbf{x}_j)$ seja maximizado e
(1) $\Sigma\alpha_i y_i = 0$
(2) $\alpha_i \geq 0$ para todo α_i

- A solução é:

$f(\mathbf{x}) = \Sigma\alpha_i y_i \mathbf{K}(\mathbf{x}_i,\mathbf{x}_j) + b$

- Técnicas de otimização para encontrar os α_i's são as mesmas!

SVM applications

- SVMs were originally proposed by Boser, Guyon and Vapnik in 1992 and gained increasing popularity in late 1990s.
- SVMs are currently among the best performers for a number of classification tasks ranging from text to genomic data.
- SVMs can be applied to complex data types beyond feature vectors (e.g. graphs, sequences, relational data) by designing kernel functions for such data.
- SVM techniques have been extended to a number of tasks such as regression [Vapnik *et al.* '97], principal component analysis [Schölkopf *et al.* '99], etc.
- Most popular optimization algorithms for SVMs use *decomposition* to hill-climb over a subset of α_i's at a time, e.g. SMO [Platt '99] and [Joachims '99]
- Tuning SVMs remains a black art: selecting a specific kernel and parameters is usually done in a try-and-see manner.