Aula 14 - Máquinas de Vetores de Suporte

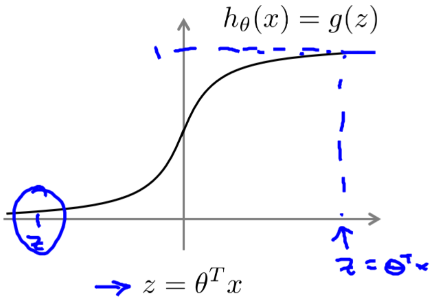
João Florindo

Instituto de Matemática, Estatística e Computação Científica
Universidade Estadual de Campinas - Brasil
florindo@unicamp.br

Outline

Regressão logística:

$$h_\theta(x) = \frac{1}{1 + \mathrm{e}^{-\theta^T x}}.$$



Custo de um exemplo:

$$-(y \log h_\theta(x) + (1 - y) \log(1 - h_\theta(x)))$$

$$= -y \log \frac{1}{1 + \mathrm{e}^{-\theta^T x}} - (1 - y) \log \left(1 - \frac{1}{1 + \mathrm{e}^{-\theta^T x}}\right).$$

- Para $y = 1$ ($z = \theta^T x \gg 0$) temos

$$-\log \frac{1}{1 + \mathrm{e}^{-z}}$$

Já para $y = 0$ ($\theta^T x \ll 0$) temos

$$-\log \left(1 - \frac{1}{1 + \mathrm{e}^{-z}} \right)$$

- SVM: Aproximar as expressões acima por funções lineares por parte ($custo_0(z)$ para $y = 0$ e $custo_1(z)$ para $y = 1$).

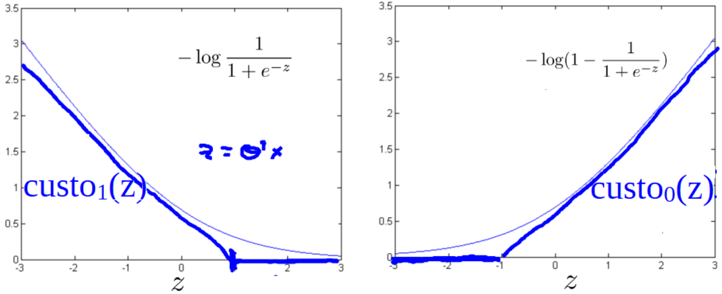
- Para $y = 1$ ($z = \theta^T x \gg 0$) temos

$$-\log \frac{1}{1+\mathrm{e}^{-z}}$$

Já para $y = 0$ ($\theta^T x \ll 0$) temos

$$-\log \left(1 - \frac{1}{1+\mathrm{e}^{-z}}\right)$$

- SVM: Aproximar as expressões acima por funções lineares por parte ($custo_0(z)$ para $y = 0$ e $custo_1(z)$ para $y = 1$).



NOTA

- Na formulação clássica de SVM, adota-se a convenção de classes $y = +1$ e $y = -1$ (em vez de $y = 0$).

- Neste contexto, a função de custo da SVM tem uma expressão geral e é chamada de **hinge loss** (perda de articulação):

$$custo(z) = \max\{0, 1 - yz\},$$

em que $y = \pm 1$.

- Função de custo da regressão logística (sinal negativo para dentro do somatório):

$$\underset{\theta}{\text{argmin}}\ \frac{1}{m}\left[\sum_{i=1}^{m} y^{(i)}\left(-\log h_\theta(x^{(i)})\right) + (1-y^{(i)})\left((-\log(1-h_\theta(x^{(i)})))\right)\right] + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2.$$

- Substituímos pela aproximação linear por partes e cortamos o denominador m (não interfere na minimização):

$$\underset{\theta}{\text{argmin}} \sum_{i=1}^{m} y^{(i)} custo_1(\theta^T x^{(i)}) + (1-y^{(i)}) custo_0(\theta^T x^{(i)}) + \frac{\lambda}{2}\sum_{j=1}^{n}\theta_j^2.$$

- Função de custo da regressão logística (sinal negativo para dentro do somatório):

$$\underset{\theta}{\text{argmin}}\ \frac{1}{m}\left[\sum_{i=1}^{m} y^{(i)}\Big(-\log h_\theta(x^{(i)})\Big) + (1-y^{(i)})\Big((-\log(1-h_\theta(x^{(i)})))\Big)\right] + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2.$$

- Substituímos pela aproximação linear por partes e cortamos o denominador m (não interfere na minimização):

$$\underset{\theta}{\text{argmin}} \sum_{i=1}^{m} y^{(i)} custo_1(\theta^T x^{(i)}) + (1-y^{(i)}) custo_0(\theta^T x^{(i)}) + \frac{\lambda}{2}\sum_{j=1}^{n}\theta_j^2.$$

- CONVENÇÃO: Minimizar $A + \lambda B$ equivale a minimizar $CA + B$ dado que $C = \frac{1}{\lambda}$. Assim:

$$\underset{\theta}{\operatorname{argmin}}\ C \sum_{i=1}^{m} \left[y^{(i)} custo_1(\theta^T x^{(i)}) + (1 - y^{(i)}) custo_0(\theta^T x^{(i)}) \right] + \frac{1}{2} \sum_{j=1}^{n} \theta_j^2$$

- Função de hipótese[1]:

$$h_\theta(x) = \begin{cases} 1 & \text{se } \theta^T x \geq 0 \\ -1 & \text{caso contrário.} \end{cases}$$

[1] Lembrando que agora a classe negativa é -1.

- CONVENÇÃO: Minimizar $A + \lambda B$ equivale a minimizar $CA + B$ dado que $C = \frac{1}{\lambda}$. Assim:

$$\underset{\theta}{\text{argmin}}\ C \sum_{i=1}^{m} \left[y^{(i)} custo_1(\theta^T x^{(i)}) + (1 - y^{(i)}) custo_0(\theta^T x^{(i)}) \right] + \frac{1}{2} \sum_{j=1}^{n} \theta_j^2$$

- Função de hipótese[1]:

$$h_\theta(x) = \begin{cases} 1 & \text{se } \theta^T x \geq 0 \\ -1 & \text{caso contrário.} \end{cases}$$

[1] Lembrando que agora a classe negativa é -1.

Outline

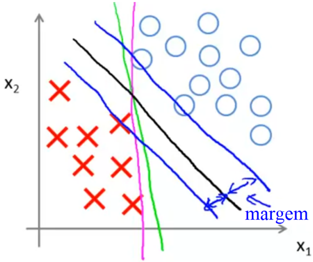
Para zerar o custo, queremos $\theta^T x \geq 1$ para $y = 1$ e $\theta^T x \leq -1$ para $y = -1$.
Suponha uma SVM com C muito grande, p.ex. $C = 100000$:

$$\underset{\theta}{\text{argmin}}\ C \sum_{i=1}^{m} \left[y^{(i)} custo_1(\theta^T x^{(i)}) + (1 - y^{(i)}) custo_0(\theta^T x^{(i)}) \right] + \frac{1}{2} \sum_{j=1}^{n} \theta_j^2.$$

Todo o primeiro somatório vai para zero, restando

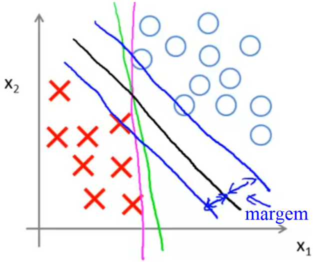
$$\begin{aligned} &\underset{\theta}{\text{argmin}}\ \frac{1}{2} \sum_{j=1}^{n} \theta_j^2 \\ s.t.\ &\begin{array}{lll} \theta^T x^{(i)} \geq 1 & \text{se} & y^{(i)} = 1 \\ \theta^T x^{(i)} \leq -1 & \text{se} & y^{(i)} = -1 \end{array} \end{aligned} \tag{1}$$

- A fronteira de decisão obtida maximiza a separação entre as classes:

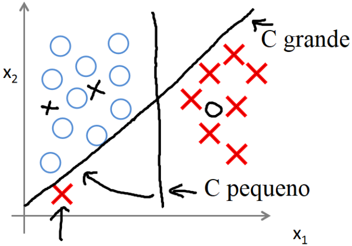


- PROBLEMA: C muito grande gera uma fronteira muito sensível a *outliers*:

- A fronteira de decisão obtida maximiza a separação entre as classes:



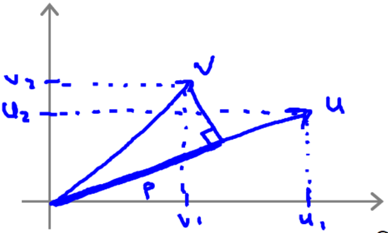
- PROBLEMA: C muito grande gera uma fronteira muito sensível a *outliers*:



Outline

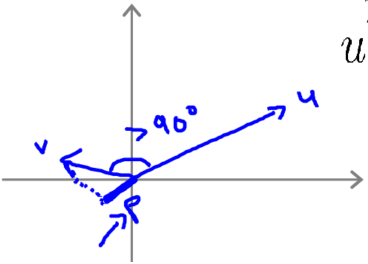
Produto interno

Seja p o comprimento da projeção do vetor **v** sobre o vetor **u**:



Sabe-se que:

$$u^T v = p\|u\|$$



A minimização do custo da SVM para C grande pode ser reescrita:

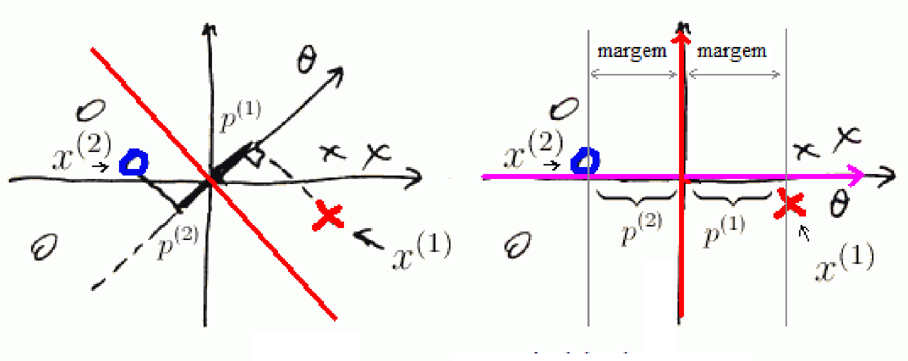
$$\underset{\theta}{\operatorname{argmin}} \frac{1}{2} \sum_{j=1}^{n} \theta_j^2 = \frac{1}{2} \|\theta\|^2$$

$$s.t. \begin{array}{lll} p^{(i)}\|\theta\| \geq 1 & \text{se} & y^{(i)} = 1 \\ p^{(i)}\|\theta\| \leq -1 & \text{se} & y^{(i)} = -1, \end{array}$$

em que $p^{(i)}$ é a projeção de $x^{(i)}$ sobre θ.

Mas sabe-se que θ é perpendicular à fronteira de decisão ($\theta^T x = 0$).

Como $\|\theta\|$ é minimizado, precisamos ter $p^{(i)}$ sempre com magnitude grande, positivo para $y = 1$ e negativo para $y = -1$. Isso implica em margem larga:



Outline

Notação

- Como vimos, na SVM convenciona-se que as classes são $y \in \{-1, 1\}$.
- Outra convenção é a reformulação da combinação linear $\theta^T x$ como
$$w^T x + b,$$
separando o termo de *bias* b.
- Assim, a otimização (1) é reescrita como
$$\begin{aligned} \min_{w,b} \quad & \frac{1}{2}\|w\|^2 \\ s.t. \quad & y^{(i)}(w^T x^{(i)} + b) \geq 1, \qquad i = 1, \cdots, m. \end{aligned} \tag{2}$$
- Este é um problema de programação quadrática.

Notação

- Como vimos, na SVM convenciona-se que as classes são $y \in \{-1, 1\}$.
- Outra convenção é a reformulação da combinação linear $\theta^T x$ como

$$w^T x + b,$$

separando o termo de *bias* b.

- Assim, a otimização (1) é reescrita como

$$\begin{aligned} \min_{w,b} \quad & \frac{1}{2}\|w\|^2 \\ s.t. \quad & y^{(i)}(w^T x^{(i)} + b) \geq 1, \qquad i = 1, \cdots, m. \end{aligned} \tag{2}$$

- Este é um problema de programação quadrática.

Notação

- Como vimos, na SVM convenciona-se que as classes são $y \in \{-1, 1\}$.

- Outra convenção é a reformulação da combinação linear $\theta^T x$ como

$$w^T x + b,$$

separando o termo de *bias* b.

- Assim, a otimização (1) é reescrita como

$$\begin{aligned} \min_{w,b} \quad & \frac{1}{2}\|w\|^2 \\ s.t. \quad & y^{(i)}(w^T x^{(i)} + b) \geq 1, \qquad i = 1, \cdots, m. \end{aligned} \tag{2}$$

- Este é um problema de programação quadrática.

Notação

- Como vimos, na SVM convenciona-se que as classes são $y \in \{-1, 1\}$.
- Outra convenção é a reformulação da combinação linear $\theta^T x$ como

$$w^T x + b,$$

separando o termo de *bias* b.

- Assim, a otimização (1) é reescrita como

$$\begin{aligned} \min_{w,b} \quad & \frac{1}{2}\|w\|^2 \\ s.t. \quad & y^{(i)}(w^T x^{(i)} + b) \geq 1, \qquad i = 1, \cdots, m. \end{aligned} \tag{2}$$

- Este é um problema de programação quadrática.

Multiplicadores de Lagrange

- Seja o problema geral

$$\begin{aligned} \min_{w} \quad & f(w) \\ s.t. \quad & h_i(w) = 0, \quad i = 1, \cdots, l. \end{aligned}$$

- Definimos a Lagrangiana:

$$\mathcal{L}(w, \beta) = f(w) + \sum_{i=1}^{l} \beta_i h_i(w),$$

em que os β_i's são os **multiplicadores de Lagrange**.

- Deve-se resolver então

$$\frac{\partial \mathcal{L}}{\partial w_i} = 0; \qquad \frac{\partial \mathcal{L}}{\partial \beta_i} = 0$$

para w e β.

Multiplicadores de Lagrange

- Seja o problema geral

$$\begin{aligned} \min_{w} \quad & f(w) \\ s.t. \quad & h_i(w) = 0, \quad i = 1, \cdots, l. \end{aligned}$$

- Definimos a Lagrangiana:

$$\mathcal{L}(w, \beta) = f(w) + \sum_{i=1}^{l} \beta_i h_i(w),$$

em que os β_i's são os **multiplicadores de Lagrange**.

- Deve-se resolver então

$$\frac{\partial \mathcal{L}}{\partial w_i} = 0; \qquad \frac{\partial \mathcal{L}}{\partial \beta_i} = 0$$

para w e β.

Multiplicadores de Lagrange

- Seja o problema geral

$$\begin{aligned} \min_{w} \quad & f(w) \\ s.t. \quad & h_i(w) = 0, \quad i = 1, \cdots, l. \end{aligned}$$

- Definimos a Lagrangiana:

$$\mathcal{L}(w, \beta) = f(w) + \sum_{i=1}^{l} \beta_i h_i(w),$$

em que os β_i's são os **multiplicadores de Lagrange**.

- Deve-se resolver então

$$\frac{\partial \mathcal{L}}{\partial w_i} = 0; \qquad \frac{\partial \mathcal{L}}{\partial \beta_i} = 0$$

para w e β.

Dualidade de Lagrange

- Vamos adicionar agora restrições de desigualdade:

$$\begin{aligned} \min_{w} \quad & f(w) \\ s.t. \quad & g_i(w) \leq 0, \quad i = 1, \cdots, k \\ & h_i(w) = 0, \quad i = 1, \cdots, l. \end{aligned} \tag{3}$$

Este é o nosso problema **primal**.

- Lagrangiana generalizada:

$$\mathcal{L}(w, \alpha, \beta) = f(w) + \sum_{i=1}^{k} \alpha_i g_i(w) + \sum_{i=1}^{l} \beta_i h_i(w).$$

Dualidade de Lagrange

- Vamos adicionar agora restrições de desigualdade:

$$\begin{aligned} \min_{w} \quad & f(w) \\ s.t. \quad & g_i(w) \leq 0, \quad i = 1, \cdots, k \\ & h_i(w) = 0, \quad i = 1, \cdots, l. \end{aligned} \tag{3}$$

Este é o nosso problema **primal**.

- Lagrangiana generalizada:

$$\mathcal{L}(w, \alpha, \beta) = f(w) + \sum_{i=1}^{k} \alpha_i g_i(w) + \sum_{i=1}^{l} \beta_i h_i(w).$$

Dualidade de Lagrange

- Definimos então
$$\theta_{\mathcal{D}}(\alpha, \beta) = \min_{w} \mathcal{L}(w, \alpha, \beta).$$

- Temos então o problema **dual**
$$\max_{\alpha, \beta : \alpha_i \geq 0} \theta_{\mathcal{D}}(\alpha, \beta).$$

- Mostra-se que sob certas condições (KKT), que o problema primal (3) é equivalente ao problema **dual**.

Dualidade de Lagrange

- Definimos então
$$\theta_{\mathcal{D}}(\alpha, \beta) = \min_{w} \mathcal{L}(w, \alpha, \beta).$$

- Temos então o problema **dual**
$$\max_{\alpha, \beta : \alpha_i \geq 0} \theta_{\mathcal{D}}(\alpha, \beta).$$

- Mostra-se que sob certas condições (KKT), que o problema primal (3) é equivalente ao problema **dual**.

Dualidade de Lagrange

- Definimos então
$$\theta_{\mathcal{D}}(\alpha, \beta) = \min_{w} \mathcal{L}(w, \alpha, \beta).$$
- Temos então o problema **dual**
$$\max_{\alpha,\beta:\alpha_i \geq 0} \theta_{\mathcal{D}}(\alpha, \beta).$$
- Mostra-se que sob certas condições (KKT), que o problema primal (3) é equivalente ao problema **dual**.

Outline

Aplicação em SVM

- Podemos reescrever a restrição de (2) resultando em

$$\begin{aligned} \min_{w,b} \quad & \frac{1}{2}\|w\|^2 \\ s.t. \quad & g_i(w) = -y^{(i)}(w^T x^{(i)} + b) + 1 \leq 0. \end{aligned}$$

- Lagrangiana:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{m} \alpha_i \left[y^{(i)}(w^T x^{(i)} + b) - 1 \right]. \qquad (4)$$

Note que não temos β_i porque não há restrição de igualdade.

Aplicação em SVM

- Podemos reescrever a restrição de (2) resultando em

$$\begin{aligned} \min_{w,b} \quad & \frac{1}{2}\|w\|^2 \\ s.t. \quad & g_i(w) = -y^{(i)}(w^T x^{(i)} + b) + 1 \leq 0. \end{aligned}$$

- Lagrangiana:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{m} \alpha_i \left[y^{(i)}(w^T x^{(i)} + b) - 1 \right]. \qquad (4)$$

Note que não temos β_i porque não há restrição de igualdade.

Aplicação em SVM

- Para obter $\theta_{\mathcal{D}}$, minimizamos $\mathcal{L}(w, b, \alpha)$ em relação a w e b (α fixo).

- Derivamos $\mathcal{L}$ e igualamos a zero:

$$\frac{\partial}{\partial w}\mathcal{L}(w, b, \alpha) = w - \sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)} = 0$$

$$\Rightarrow w = \sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)}. \tag{5}$$

Além disso:

$$\frac{\partial}{\partial b}\mathcal{L}(w, b, \alpha) = \sum_{i=1}^{m} \alpha_i y^{(i)} = 0. \tag{6}$$

Aplicação em SVM

- Para obter $\theta_{\mathcal{D}}$, minimizamos $\mathcal{L}(w, b, \alpha)$ em relação a w e b (α fixo).
- Derivamos $\mathcal{L}$ e igualamos a zero:

$$\frac{\partial}{\partial w}\mathcal{L}(w, b, \alpha) = w - \sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)} = 0$$

$$\Rightarrow w = \sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)}. \tag{5}$$

Além disso:

$$\frac{\partial}{\partial b}\mathcal{L}(w, b, \alpha) = \sum_{i=1}^{m} \alpha_i y^{(i)} = 0. \tag{6}$$

Aplicação em SVM

- Substituindo (5) em (4):

$$\mathcal{L}(w,b,\alpha)=\sum_{i=1}^{m}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{m}y^{(i)}y^{(j)}\alpha_i\alpha_j(x^{(i)})^T x^{(j)}-b\sum_{i=1}^{m}\alpha_i y^{(i)}.$$

- Mas de (6) temos que o último termo é zero e portanto:

$$\mathcal{L}(w,b,\alpha)=\sum_{i=1}^{m}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{m}y^{(i)}y^{(j)}\alpha_i\alpha_j(x^{(i)})^T x^{(j)}.$$

Aplicação em SVM

- Substituindo (5) em (4):

$$\mathcal{L}(w, b, \alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j (x^{(i)})^T x^{(j)} - b \sum_{i=1}^{m} \alpha_i y^{(i)}.$$

- Mas de (6) temos que o último termo é zero e portanto:

$$\mathcal{L}(w, b, \alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j (x^{(i)})^T x^{(j)}.$$

Aplicação em SVM

- Assim, com as restrições (6) e $\alpha_i \geq 0$ (sempre presente), temos o problema dual:

$$\begin{aligned} \max_{\alpha} \quad & W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j \langle x^{(i)}, x^{(j)} \rangle \\ s.t. \quad & \alpha_i \geq 0, \qquad i = 1, \cdots, m \\ & \sum_{i=1}^{m} \alpha_i y^{(i)} = 0. \end{aligned} \tag{7}$$

Aplicação em SVM

- A partir dos α_i's ótimos em (7) obtemos w de (5) e b do problema primal.

- Além disso, a SVM prevê $y = 1$ se $w^T x + b > 0$. Mas:

$$
\begin{aligned}
w^T x + b &= \left(\sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)} \right)^T x + b \\
&= \sum_{i=1}^{m} \alpha_i y^{(i)} \langle x^{(i)}, x \rangle + b.
\end{aligned}
$$

- Pode-se mostrar ainda (condições KKT) que só temos $\alpha_i \neq 0$ se o par $(x^{(i)}, y^{(i)})$ está sobre a margem máxima (**vetores de suporte**).

Aplicação em SVM

- A partir dos α_i's ótimos em (7) obtemos w de (5) e b do problema primal.

- Além disso, a SVM prevê $y = 1$ se $w^T x + b > 0$. Mas:

$$
\begin{aligned}
w^T x + b &= \left(\sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)} \right)^T x + b \\
&= \sum_{i=1}^{m} \alpha_i y^{(i)} \langle x^{(i)}, x \rangle + b.
\end{aligned}
$$

- Pode-se mostrar ainda (condições KKT) que só temos $\alpha_i \neq 0$ se o par $(x^{(i)}, y^{(i)})$ está sobre a margem máxima (**vetores de suporte**).

Aplicação em SVM

- A partir dos α_i's ótimos em (7) obtemos w de (5) e b do problema primal.

- Além disso, a SVM prevê $y = 1$ se $w^T x + b > 0$. Mas:

$$
\begin{aligned}
w^T x + b &= \left(\sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)} \right)^T x + b \\
&= \sum_{i=1}^{m} \alpha_i y^{(i)} \langle x^{(i)}, x \rangle + b.
\end{aligned}
$$

- Pode-se mostrar ainda (condições KKT) que só temos $\alpha_i \neq 0$ se o par $(x^{(i)}, y^{(i)})$ está sobre a margem máxima (**vetores de suporte**).

Outline

SMO (*Sequential Minimal Optimization*)

- Baseado no método de **coordenada ascendente**.

- Suponha o problema sem restrições

$$\max_{\alpha} W(\alpha_1, \alpha_2, \cdots, \alpha_m)$$

.

- Algoritmo:

Faça até convergir{

Para $i = 1, \cdots, m\{$

$$\alpha_i := \operatorname*{argmax}_{\hat{\alpha}_i} W(\alpha_1, \cdots, \alpha_{i-1}, \hat{\alpha}_i, \alpha_{i+1}, \cdots, \alpha_m)$$

}

}

SMO (*Sequential Minimal Optimization*)

- Baseado no método de **coordenada ascendente**.
- Suponha o problema sem restrições

$$\max_{\alpha} W(\alpha_1, \alpha_2, \cdots, \alpha_m)$$

.

- Algoritmo:
  Faça até convergir{
  Para $i = 1, \cdots, m${

$$\alpha_i := \underset{\hat{\alpha}_i}{\operatorname{argmax}} W(\alpha_1, \cdots, \alpha_{i-1}, \hat{\alpha}_i, \alpha_{i+1}, \cdots, \alpha_m)$$

  }
  }

SMO (*Sequential Minimal Optimization*)

- Baseado no método de **coordenada ascendente**.
- Suponha o problema sem restrições

$$\max_{\alpha} W(\alpha_1, \alpha_2, \cdots, \alpha_m)$$

.

- Algoritmo:

Faça até convergir{

Para $i = 1, \cdots, m${

$$\alpha_i := \underset{\hat{\alpha}_i}{\operatorname{argmax}} W(\alpha_1, \cdots, \alpha_{i-1}, \hat{\alpha}_i, \alpha_{i+1}, \cdots, \alpha_m)$$

}

}

SMO

- No caso regularizado, (7) se converte em

$$\begin{aligned} \max_{\alpha} \quad & W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j \langle x^{(i)}, x^{(j)} \rangle \\ s.t. \quad & 0 \leq \alpha_i \leq C, \qquad i = 1, \cdots, m \\ & \sum_{i=1}^{m} \alpha_i y^{(i)} = 0. \end{aligned} \tag{8}$$

- SMO aplica coordenada ascendente, porém reotimizando $W(\alpha)$ sempre em relação a um par α_i e α_j.

SMO

- No caso regularizado, (7) se converte em

$$\begin{aligned} \max_{\alpha} \quad & W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j \langle x^{(i)}, x^{(j)} \rangle \\ s.t. \quad & 0 \leq \alpha_i \leq C, \qquad i = 1, \cdots, m \\ & \sum_{i=1}^{m} \alpha_i y^{(i)} = 0. \end{aligned} \tag{8}$$

- SMO aplica coordenada ascendente, porém reotimizando $W(\alpha)$ sempre em relação a um par α_i e α_j.

SMO

- Suponha que comecemos com α_1 e α_2, mantendo $\alpha_3, \cdots, \alpha_m$ fixos.
- Da 2ª restrição de (8) temos

$$\alpha_1 y^{(1)} + \alpha_2 y^{(2)} = -\sum_{i=3}^{m} \alpha_i y^{(i)}.$$

- Como $\alpha_3, \cdots, \alpha_m$ é fixado, o lado direito é uma constante ζ:

$$\alpha_1 y^{(1)} + \alpha_2 y^{(2)} = \zeta. \tag{9}$$

- Além disso, da 1ª restrição temos que α_1 e α_2 estão na caixa $[0, C] \times [0, C]$.
- α_2 limitado entre L e H (dependem de ζ).

SMO

- Suponha que comecemos com α_1 e α_2, mantendo $\alpha_3, \cdots, \alpha_m$ fixos.
- Da 2ª restrição de (8) temos

$$\alpha_1 y^{(1)} + \alpha_2 y^{(2)} = -\sum_{i=3}^{m} \alpha_i y^{(i)}.$$

- Como $\alpha_3, \cdots, \alpha_m$ é fixado, o lado direito é uma constante ζ:

$$\alpha_1 y^{(1)} + \alpha_2 y^{(2)} = \zeta. \tag{9}$$

- Além disso, da 1ª restrição temos que α_1 e α_2 estão na caixa $[0, C] \times [0, C]$.
- α_2 limitado entre L e H (dependem de ζ).

SMO

- Suponha que comecemos com α_1 e α_2, mantendo $\alpha_3, \cdots, \alpha_m$ fixos.
- Da 2ª restrição de (8) temos
$$\alpha_1 y^{(1)} + \alpha_2 y^{(2)} = -\sum_{i=3}^{m} \alpha_i y^{(i)}.$$
- Como $\alpha_3, \cdots, \alpha_m$ é fixado, o lado direito é uma constante ζ:
$$\alpha_1 y^{(1)} + \alpha_2 y^{(2)} = \zeta. \tag{9}$$
- Além disso, da 1ª restrição temos que α_1 e α_2 estão na caixa $[0, C] \times [0, C]$.
- α_2 limitado entre L e H (dependem de ζ).

SMO

- Suponha que comecemos com α_1 e α_2, mantendo $\alpha_3, \cdots, \alpha_m$ fixos.

- Da 2ª restrição de (8) temos

$$\alpha_1 y^{(1)} + \alpha_2 y^{(2)} = -\sum_{i=3}^{m} \alpha_i y^{(i)}.$$

- Como $\alpha_3, \cdots, \alpha_m$ é fixado, o lado direito é uma constante ζ:

$$\alpha_1 y^{(1)} + \alpha_2 y^{(2)} = \zeta. \tag{9}$$

- Além disso, da 1ª restrição temos que α_1 e α_2 estão na caixa $[0, C] \times [0, C]$.

- α_2 limitado entre L e H (dependem de ζ).

SMO

- Suponha que comecemos com α_1 e α_2, mantendo $\alpha_3, \cdots, \alpha_m$ fixos.

- Da 2ª restrição de (8) temos

$$\alpha_1 y^{(1)} + \alpha_2 y^{(2)} = -\sum_{i=3}^{m} \alpha_i y^{(i)}.$$

- Como $\alpha_3, \cdots, \alpha_m$ é fixado, o lado direito é uma constante ζ:

$$\alpha_1 y^{(1)} + \alpha_2 y^{(2)} = \zeta. \tag{9}$$

- Além disso, da 1ª restrição temos que α_1 e α_2 estão na caixa $[0, C] \times [0, C]$.

- α_2 limitado entre L e H (dependem de ζ).

SMO

- De (9) temos[2]

$$\alpha_1 = (\zeta - \alpha_2 y^{(2)}) y^{(1)}.$$

- Assim, finalmente reescrevemos $W(\alpha)$:

$$W(\alpha_1, \alpha_2, \cdots, \alpha_n) = W((\zeta - \alpha_2 y^{(2)}) y^{(1)}, \alpha_2, \cdots, \alpha_n).$$

Tratando $\alpha_3, \cdots, \alpha_m$ como constantes, esta é uma função quadrática em α_2, maximizada trivialmente igualando-se a derivada a zero.

[2] Multiplicamos ambos os lados por $(y^{(1)})^2$ e lembramos que $(y^{(1)})^2 = 1$.

SMO

- De (9) temos[2]

$$\alpha_1 = (\zeta - \alpha_2 y^{(2)}) y^{(1)}.$$

- Assim, finalmente reescrevemos $W(\alpha)$:

$$W(\alpha_1, \alpha_2, \cdots, \alpha_n) = W((\zeta - \alpha_2 y^{(2)}) y^{(1)}, \alpha_2, \cdots, \alpha_n).$$

Tratando $\alpha_3, \cdots, \alpha_m$ como constantes, esta é uma função quadrática em α_2, maximizada trivialmente igualando-se a derivada a zero.

[2] Multiplicamos ambos os lados por $(y^{(1)})^2$ e lembramos que $(y^{(1)})^2 = 1$.

SMO

- Note que ignoramos a restrição $L \leq \alpha_2 \leq H$. Por isso, o α_2 obtido é chamado de
$$\alpha_2^{novo,unclipped}.$$

- Devemos então "clipá-lo":
$$\alpha_2^{novo} = \begin{cases} H & \text{se } \alpha_2^{novo,unclipped} > H \\ \alpha_2^{novo,unclipped} & \text{se } L \leq \alpha_2^{novo,unclipped} \leq H \\ L & \text{se } \alpha_2^{novo,unclipped} < L \end{cases}$$

- Por fim, α_1^{novo} é obtido de (9).

- Há mais detalhes, p.ex., uma heurística para escolher os pares α_i e α_j.

SMO

- Note que ignoramos a restrição $L \leq \alpha_2 \leq H$. Por isso, o α_2 obtido é chamado de
$$\alpha_2^{novo,unclipped}.$$
- Devemos então "clipá-lo":
$$\alpha_2^{novo} = \begin{cases} H & \text{se } \alpha_2^{novo,unclipped} > H \\ \alpha_2^{novo,unclipped} & \text{se } L \leq \alpha_2^{novo,unclipped} \leq H \\ L & \text{se } \alpha_2^{novo,unclipped} < L \end{cases}$$
- Por fim, α_1^{novo} é obtido de (9).
- Há mais detalhes, p.ex., uma heurística para escolher os pares α_i e α_j.

SMO

- Note que ignoramos a restrição $L \leq \alpha_2 \leq H$. Por isso, o α_2 obtido é chamado de
$$\alpha_2^{novo,unclipped}.$$

- Devemos então "clipá-lo":
$$\alpha_2^{novo} = \begin{cases} H & \text{se } \alpha_2^{novo,unclipped} > H \\ \alpha_2^{novo,unclipped} & \text{se } L \leq \alpha_2^{novo,unclipped} \leq H \\ L & \text{se } \alpha_2^{novo,unclipped} < L \end{cases}$$

- Por fim, α_1^{novo} é obtido de (9).

- Há mais detalhes, p.ex., uma heurística para escolher os pares α_i e α_j.

SMO

- Note que ignoramos a restrição $L \leq \alpha_2 \leq H$. Por isso, o α_2 obtido é chamado de
$$\alpha_2^{novo,unclipped}.$$

- Devemos então "clipá-lo":
$$\alpha_2^{novo} = \begin{cases} H & \text{se } \alpha_2^{novo,unclipped} > H \\ \alpha_2^{novo,unclipped} & \text{se } L \leq \alpha_2^{novo,unclipped} \leq H \\ L & \text{se } \alpha_2^{novo,unclipped} < L \end{cases}$$

- Por fim, α_1^{novo} é obtido de (9).

- Há mais detalhes, p.ex., uma heurística para escolher os pares α_i e α_j.

Outline

Método de *kernels*

No método de **kernels**, cada *feature* $f_i(\mathbf{x})$ é obtido a partir de alguma medida de similaridade K entre um ponto de treinamento $\mathbf{x}$ e um ponto de referência $l^{(i)}$ (*landmarks*).

- O que são estes *landmarks*?
  - Podem ser simplesmente os próprios pontos de treinamento.
  - Dado o conjunto de treino $(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \cdots, (x^{(m)}, y^{(m)})$, tomamos

$$l^{(1)} = x^{(1)}, l^{(2)} = x^{(2)}, \cdots, l^{(m)} = x^{(m)}.$$

Método de *kernels*

No método de **kernels**, cada *feature* $f_i(\mathbf{x})$ é obtido a partir de alguma medida de similaridade K entre um ponto de treinamento $\mathbf{x}$ e um ponto de referência $l^{(i)}$ (*landmarks*).

- O que são estes *landmarks*?
  - Podem ser simplesmente os próprios pontos de treinamento.
  - Dado o conjunto de treino $(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \cdots, (x^{(m)}, y^{(m)})$, tomamos
$$l^{(1)} = x^{(1)}, l^{(2)} = x^{(2)}, \cdots, l^{(m)} = x^{(m)}.$$

Método de *kernels*

No método de **kernels**, cada *feature* $f_i(\mathbf{x})$ é obtido a partir de alguma medida de similaridade K entre um ponto de treinamento $\mathbf{x}$ e um ponto de referência $l^{(i)}$ (*landmarks*).

- O que são estes *landmarks*?
  - Podem ser simplesmente os próprios pontos de treinamento.
  - Dado o conjunto de treino $(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \cdots, (x^{(m)}, y^{(m)})$, tomamos
$$l^{(1)} = x^{(1)}, l^{(2)} = x^{(2)}, \cdots, l^{(m)} = x^{(m)}.$$

- Função de hipótese $h_\theta(x)$ da SVM com kernel prevê $y = 1$ se

$$\theta_0 f_0(x) + \theta_1 f_1(x) + \cdots + \theta_m f_m(x) \geq 0.$$

- Treinamento:

$$\underset{\theta}{\text{argmin}}\ C \sum_{i=1}^{m} y^{(i)} custo_1(\theta^T f(x^{(i)})) + (1 - y^{(i)}) custo_0(\theta^T f(x^{(i)})) + \frac{1}{2} \sum_{j=1}^{m} \theta_j^2$$

- NOTA 1: Temos j indo de 1 a m agora porque $m = n$.

- NOTA 2: Para fins computacionais quando m é grande, é comum que $\sum_j \theta_j^2 = \theta^T \theta = \|\theta\|^2$ seja substituído por $\theta^T M \theta$ para uma matriz M específica.

- Função de hipótese $h_\theta(x)$ da SVM com kernel prevê $y = 1$ se

$$\theta_0 f_0(x) + \theta_1 f_1(x) + \cdots + \theta_m f_m(x) \geq 0.$$

- Treinamento:

$$\underset{\theta}{\text{argmin}}\ C \sum_{i=1}^{m} y^{(i)} custo_1(\theta^T f(x^{(i)})) + (1 - y^{(i)}) custo_0(\theta^T f(x^{(i)})) + \frac{1}{2} \sum_{j=1}^{m} \theta_j^2$$

- NOTA 1: Temos j indo de 1 a m agora porque $m = n$.
- NOTA 2: Para fins computacionais quando m é grande, é comum que $\sum_j \theta_j^2 = \theta^T \theta = \|\theta\|^2$ seja substituído por $\theta^T M \theta$ para uma matriz M específica.

- Função de hipótese $h_\theta(x)$ da SVM com kernel prevê $y = 1$ se

$$\theta_0 f_0(x) + \theta_1 f_1(x) + \cdots + \theta_m f_m(x) \geq 0.$$

- Treinamento:

$$\underset{\theta}{\operatorname{argmin}}\ C \sum_{i=1}^{m} y^{(i)} custo_1(\theta^T f(x^{(i)})) + (1 - y^{(i)}) custo_0(\theta^T f(x^{(i)})) + \frac{1}{2} \sum_{j=1}^{m} \theta_j^2$$

- NOTA 1: Temos j indo de 1 a m agora porque $m = n$.

- NOTA 2: Para fins computacionais quando m é grande, é comum que $\sum_j \theta_j^2 = \theta^T \theta = \|\theta\|^2$ seja substituído por $\theta^T M \theta$ para uma matriz M específica.

- Função de hipótese $h_\theta(x)$ da SVM com kernel prevê $y = 1$ se

$$\theta_0 f_0(x) + \theta_1 f_1(x) + \cdots + \theta_m f_m(x) \geq 0.$$

- Treinamento:

$$\underset{\theta}{\text{argmin}}\, C \sum_{i=1}^{m} y^{(i)} custo_1(\theta^T f(x^{(i)})) + (1 - y^{(i)}) custo_0(\theta^T f(x^{(i)})) + \frac{1}{2} \sum_{j=1}^{m} \theta_j^2$$

- NOTA 1: Temos j indo de 1 a m agora porque $m = n$.

- NOTA 2: Para fins computacionais quando m é grande, é comum que $\sum_j \theta_j^2 = \theta^T \theta = \|\theta\|^2$ seja substituído por $\theta^T M \theta$ para uma matriz M específica.

Outline

Parâmetros

- Escolha de $C = \frac{1}{\lambda}$:
  - C grande: viés baixo, variância alta (λ pequeno)
  - C pequeno: viés alto, variância baixa (λ grande)
- Já para σ^2:
  - σ^2 grande: f_i varia mais suavemente - viés alto, variância baixa
  - σ^2 pequeno: f_i varia menos suavemente - viés baixo, variância alta
- Kernel:
  - A SVM sem kernel ("kernel linear") é recomendada quando n é grande e m é pequeno.
  - Kernel Gaussiano é recomendado quando n é pequeno e/ou m é grande.

Parâmetros

- Escolha de $C = \frac{1}{\lambda}$:
  - C grande: viés baixo, variância alta (λ pequeno)
  - C pequeno: viés alto, variância baixa (λ grande)
- Já para σ^2:
  - σ^2 grande: f_i varia mais suavemente - viés alto, variância baixa
  - σ^2 pequeno: f_i varia menos suavemente - viés baixo, variância alta
- Kernel:
  - A SVM sem kernel ("kernel linear") é recomendada quando n é grande e m é pequeno.
  - Kernel Gaussiano é recomendado quando n é pequeno e/ou m é grande.

Parâmetros

- Escolha de $C = \frac{1}{\lambda}$:
  - C grande: viés baixo, variância alta (λ pequeno)
  - C pequeno: viés alto, variância baixa (λ grande)
- Já para σ^2:
  - σ^2 grande: f_i varia mais suavemente - viés alto, variância baixa
  - σ^2 pequeno: f_i varia menos suavemente - viés baixo, variância alta
- Kernel:
  - A SVM sem kernel ("kernel linear") é recomendada quando n é grande e m é pequeno.
  - Kernel Gaussiano é recomendado quando n é pequeno e/ou m é grande.

Parâmetros

- Escolha de $C = \frac{1}{\lambda}$:
  - C grande: viés baixo, variância alta (λ pequeno)
  - C pequeno: viés alto, variância baixa (λ grande)
- Já para σ^2:
  - σ^2 grande: f_i varia mais suavemente - viés alto, variância baixa
  - σ^2 pequeno: f_i varia menos suavemente - viés baixo, variância alta
- Kernel:
  - A SVM sem kernel ("kernel linear") é recomendada quando n é grande e m é pequeno.
  - Kernel Gaussiano é recomendado quando n é pequeno e/ou m é grande.

Parâmetros

- Escolha de $C = \frac{1}{\lambda}$:
  - C grande: viés baixo, variância alta (λ pequeno)
  - C pequeno: viés alto, variância baixa (λ grande)

- Já para σ^2:
  - σ^2 grande: f_i varia mais suavemente - viés alto, variância baixa
  - σ^2 pequeno: f_i varia menos suavemente - viés baixo, variância alta

- Kernel:
  - A SVM sem kernel ("kernel linear") é recomendada quando n é grande e m é pequeno.
  - Kernel Gaussiano é recomendado quando n é pequeno e/ou m é grande.

Parâmetros

- Escolha de $C = \frac{1}{\lambda}$:
  - C grande: viés baixo, variância alta (λ pequeno)
  - C pequeno: viés alto, variância baixa (λ grande)

- Já para σ^2:
  - σ^2 grande: f_i varia mais suavemente - viés alto, variância baixa
  - σ^2 pequeno: f_i varia menos suavemente - viés baixo, variância alta

- Kernel:
  - A SVM sem kernel ("kernel linear") é recomendada quando n é grande e m é pequeno.
  - Kernel Gaussiano é recomendado quando n é pequeno e/ou m é grande.

Parâmetros

- Escolha de $C = \frac{1}{\lambda}$:
  - C grande: viés baixo, variância alta (λ pequeno)
  - C pequeno: viés alto, variância baixa (λ grande)

- Já para σ^2:
  - σ^2 grande: f_i varia mais suavemente - viés alto, variância baixa
  - σ^2 pequeno: f_i varia menos suavemente - viés baixo, variância alta

- Kernel:
  - A SVM sem kernel ("kernel linear") é recomendada quando n é grande e m é pequeno.
  - Kernel Gaussiano é recomendado quando n é pequeno e/ou m é grande.

Parâmetros

- Escolha de $C = \frac{1}{\lambda}$:
  - C grande: viés baixo, variância alta (λ pequeno)
  - C pequeno: viés alto, variância baixa (λ grande)

- Já para σ^2:
  - σ^2 grande: f_i varia mais suavemente - viés alto, variância baixa
  - σ^2 pequeno: f_i varia menos suavemente - viés baixo, variância alta

- Kernel:
  - A SVM sem kernel ("kernel linear") é recomendada quando n é grande e m é pequeno.
  - Kernel Gaussiano é recomendado quando n é pequeno e/ou m é grande.

Parâmetros

- Escolha de $C = \frac{1}{\lambda}$:
  - C grande: viés baixo, variância alta (λ pequeno)
  - C pequeno: viés alto, variância baixa (λ grande)

- Já para σ^2:
  - σ^2 grande: f_i varia mais suavemente - viés alto, variância baixa
  - σ^2 pequeno: f_i varia menos suavemente - viés baixo, variância alta

- Kernel:
  - A SVM sem kernel ("kernel linear") é recomendada quando n é grande e m é pequeno.
  - Kernel Gaussiano é recomendado quando n é pequeno e/ou m é grande.

Implementação

- Vários pacotes para obter θ na SVM (liblinear, libsvm,...).
- Usuário fornece $\{x^{(i)}, y^{(i)}\}$, parâmetro C e a função de similaridade $K(x, l)$ do kernel.
- IMPORTANTE: Sempre FAÇA normalização de atributos antes de usar o kernel Gaussiano.

Implementação

- Vários pacotes para obter θ na SVM (liblinear, libsvm,...).
- Usuário fornece $\{x^{(i)}, y^{(i)}\}$, parâmetro C e a função de similaridade $K(x, l)$ do kernel.
- IMPORTANTE: Sempre FAÇA normalização de atributos antes de usar o kernel Gaussiano.

Implementação

- Vários pacotes para obter θ na SVM (liblinear, libsvm,...).
- Usuário fornece $\{x^{(i)}, y^{(i)}\}$, parâmetro C e a função de similaridade $K(x, l)$ do kernel.
- IMPORTANTE: Sempre FAÇA normalização de atributos antes de usar o kernel Gaussiano.

Kernels Alternativos

- Nem toda função de similaridade $K(x, l)$ gera kernels válidos ("Teorema de Mercer").

- Mas existem outras opções além do Gaussiano:
  - Kernel polinomial: $K(x, l) = (x^T l + \text{constante})^{\text{grau}}$. EX.: $(x^T l)^2$, $(x^T l + 5)^4$, etc.
  - Kernels mais exóticos: string (usado para texto), chi-quadrado, intersecção de histograma, etc.

Kernels Alternativos

- Nem toda função de similaridade $K(x, l)$ gera kernels válidos ("Teorema de Mercer").
- Mas existem outras opções além do Gaussiano:
  - Kernel polinomial: $K(x, l) = (x^T l + \text{constante})^{\text{grau}}$. EX.: $(x^T l)^2$, $(x^T l + 5)^4$, etc.
  - Kernels mais exóticos: string (usado para texto), chi-quadrado, intersecção de histograma, etc.

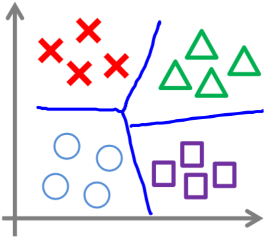
Kernels Alternativos

- Nem toda função de similaridade $K(x, l)$ gera kernels válidos ("Teorema de Mercer").
- Mas existem outras opções além do Gaussiano:
  - Kernel polinomial: $K(x, l) = (x^T l + \text{constante})^{\text{grau}}$. EX.: $(x^T l)^2$, $(x^T l + 5)^4$, etc.
  - Kernels mais exóticos: string (usado para texto), chi-quadrado, intersecção de histograma, etc.

Kernels Alternativos

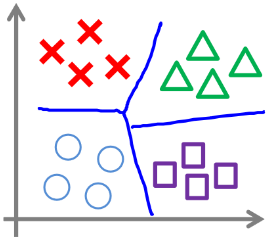
- Nem toda função de similaridade $K(x, l)$ gera kernels válidos ("Teorema de Mercer").
- Mas existem outras opções além do Gaussiano:
    - Kernel polinomial: $K(x, l) = (x^T l + \text{constante})^{\text{grau}}$. EX.: $(x^T l)^2$, $(x^T l + 5)^4$, etc.
    - Kernels mais exóticos: string (usado para texto), chi-quadrado, intersecção de histograma, etc.

SVM Multiclasses



- Existem implementações de SVM com multiclasses nativa.
- Uma alternativa é usar o método "um-contra-todos".

SVM Multiclasses



- Existem implementações de SVM com multiclasses nativa.
- Uma alternativa é usar o método "um-contra-todos".

Quando Usar SVM?

- Se n é grande (em relação a m), p.ex., $n = 10000$ e $m \in [10, 1000]$, usar regressão logística ou SVM com "kernel linear".
- Se n é pequeno e m intermediário, p.ex., $n \in [1, 1000]$ e $m = [10, 10000]$, usar SVM com kernel Gaussiano.
- Se n é pequeno e m é grande, p.ex., $n = [1, 1000]$ e $m \geq 50000$, criar/adicionar mais atributos e usar regressão logística ou SVM sem kernel.
- Redes neurais provavelmente funcionarão bem em todos esses casos, porém, são mais lentas para treinar.

Quando Usar SVM?

- Se n é grande (em relação a m), p.ex., $n = 10000$ e $m \in [10, 1000]$, usar regressão logística ou SVM com "kernel linear".
- Se n é pequeno e m intermediário, p.ex., $n \in [1, 1000]$ e $m = [10, 10000]$, usar SVM com kernel Gaussiano.
- Se n é pequeno e m é grande, p.ex., $n = [1, 1000]$ e $m \geq 50000$, criar/adicionar mais atributos e usar regressão logística ou SVM sem kernel.
- Redes neurais provavelmente funcionarão bem em todos esses casos, porém, são mais lentas para treinar.

Quando Usar SVM?

- Se n é grande (em relação a m), p.ex., $n = 10000$ e $m \in [10, 1000]$, usar regressão logística ou SVM com "kernel linear".
- Se n é pequeno e m intermediário, p.ex., $n \in [1, 1000]$ e $m = [10, 10000]$, usar SVM com kernel Gaussiano.
- Se n é pequeno e m é grande, p.ex., $n = [1, 1000]$ e $m \geq 50000$, criar/adicionar mais atributos e usar regressão logística ou SVM sem kernel.
- Redes neurais provavelmente funcionarão bem em todos esses casos, porém, são mais lentas para treinar.

Quando Usar SVM?

- Se n é grande (em relação a m), p.ex., $n = 10000$ e $m \in [10, 1000]$, usar regressão logística ou SVM com "kernel linear".
- Se n é pequeno e m intermediário, p.ex., $n \in [1, 1000]$ e $m = [10, 10000]$, usar SVM com kernel Gaussiano.
- Se n é pequeno e m é grande, p.ex., $n = [1, 1000]$ e $m \geq 50000$, criar/adicionar mais atributos e usar regressão logística ou SVM sem kernel.
- Redes neurais provavelmente funcionarão bem em todos esses casos, porém, são mais lentas para treinar.