5
Classificador SVM

Support Vector Machines (SVM) consiste numa técnica de aprendizado para problemas de reconhecimento de padrão. A técnica foi introduzida por Vapnik (03) e vem sendo usada com grande sucesso em problemas de categorização de texto. O método SVM é essencialmente uma abordagem geométrica para o problema de classificação. Cada texto do nosso conjunto de treinamento pode ser visto como um ponto x_i num espaço $\mathbb{R}^M$ e o aprendizado consiste em "dividir" os elementos positivos dos negativos neste espaço euclidiano.

5.1
Fundamentos Teóricos

Dados D exemplos de treinamento $\{x_i, y_i\}, \quad i = 1, 2, ..., D$ onde $x_i \in \mathbb{R}^M$ é uma representação vetorial de um documento e $y_i \in \{-1, 1\}$ sua classe associada. Suponha que existe uma distribuição de probabilidade $\Pr(x, y)$ desconhecida da qual os dados de treinamento foram retirados.

O processo de treinamento consiste em treinar um classificador de forma que este aprenda um mapeamento $x \mapsto y$ através dos exemplos de treinamento $\{x_i, y_i\}$ de tal forma que a máquina seja capaz de classificar um exemplo (x,y) ainda não visto que siga a mesma distribuição de probabilidade dos exemplos de treinamento.

Na prática, um classificador pode ser treinado pra aprender um conjunto de possíveis mapeamentos $x \mapsto f(x, \alpha)$

O objetivo do classificador SVM consiste em minimizar a expectativa de erro $R(\alpha)$ numa classificação onde $R(\alpha)$ é dado por:

$$R(\alpha) = \int \frac{1}{2} |y - f(x, \alpha)| \, d\Pr(x, y) \tag{5-1}$$

Note que uma vez que a distribuição $Pr(x, y)$ não é conhecida, não é possível computar esta equação. Por outro lado, o risco empírico, $R_{emp}(\alpha)$, definido como a média da taxa de erro nos elementos do conjunto de treinamento pode ser definido como:

$$R_{emp}(\alpha) = \frac{1}{2D}\sum_{i=1}^{D}|y_i - f(x_i, \alpha)| \quad (5\text{-}2)$$

Note que R_{emp} é fixo para um α arbitrário e um conjunto de treinamento $\{x_i, y_i\}$.

5.1.1
A dimensão VP e o limite do risco esperado

Conforme definido anteriormente, α é um parâmetro ajustável em $f(x, \alpha)$. Assim, ao escolher um valor para α estamos essencialmente selecionando uma determinada função do conjunto de funções contidas em $f(\alpha)$. Para cada conjunto tal, existe uma dimensão Vapnik-Chervonenkis (VC).

Considere dado um problema de reconhecimento de padrão em 2 classes a partir de D pontos de treinamento. Tais pontos podem estar rotulados de 2^D combinações. A dimensão-VC para o conjunto $f(a)$ é definida como o máximo de pontos de treinamento que podem ser corretamente classificado por um membro de $f(a)$. Com esta definição, podemos agora definir um limite para a expectativa de risco.

Seja h a dimensão-VC do conjunto $\{f, (\alpha)\}$, e seja $R_{emp}(\alpha)$ definido pela Equação (5-1). Ao selecionarmos um η tal que $0 <= \eta <= 1$, o seguinte limite é válido com probabilidade de pelo menos $1 - \eta$ para $D > h$:

$$R(\alpha) <= R_{emp}(\alpha) + \sqrt{\frac{h[\log(\frac{2D}{h}) + 1] - \log(\frac{\eta}{4})}{D}} \quad (5\text{-}3)$$

O termo somado a $R_{emp}(\alpha)$ é conhecido como "termo de complexidade". Detalhes de como chega-se a este valor estão além do escopo deste trabalho.

Note que não é possível calcular o lado esquerdo da Equação (5-3). Porém, se conhecemos a dimensão-VC, h, podemos computar o lado direito desta Equação e conseqüentemente encontrar uma máquina com o menor limite de risco esperado. Assim, temos um método para escolher uma máquina dado uma tarefa. Esta é a idéia essencial da minimização de risco estrutural (05).

5.2
Máquinas de Vetores Suporte Lineares

Dados D exemplos de treinamento $\{x_i, y_i\}$, $i = 1, 2, \ldots D$, onde $x_i \in \mathbb{R}$ é uma representação vetorial de um documento e $y_i \in \{-1, 1\}$ sua classe correspondente, suponha que seja possível traçar um hiperplano que separe os exemplos positivos dos negativos. Este hiperplano seria definido

por $x \cdot w + b = 0$ onde x são pontos sobre o hiperplano. w é a normal ao hiperplano, e $|b|\,\|w\|$ é a distância perpendicular do hiperplano à origem.

Sejam d^+ e d^- as distâncas perpendiculares do hiperplano separador aos exemplos positivos e negativos respectivamente. Podemos definir como "margem" do hiperplano separador $d^+ + d^-$. Caso o conjunto de treinamento seja linearmente separável, o algoritmo do *Support Vectors* busca o hiperplano separador tal que a margem seja maximizada. Para formular este algoritmo, suponha que os exemplos de treinamento são linearmente separáveis, isto é, satisfazem as seguintes restrições:

$$x_i \cdot w + b \geq +1 \quad para \quad y_i = +1 \tag{5-4}$$

$$x_i \cdot w + b \leq -1 \quad para \quad y_i = -1 \tag{5-5}$$

Podemos combinar estas desigualdades e obter:

$$y_i(x_i \cdot w + b) - 1 \geq 0, \quad i = 1, 2, ..., D \tag{5-6}$$

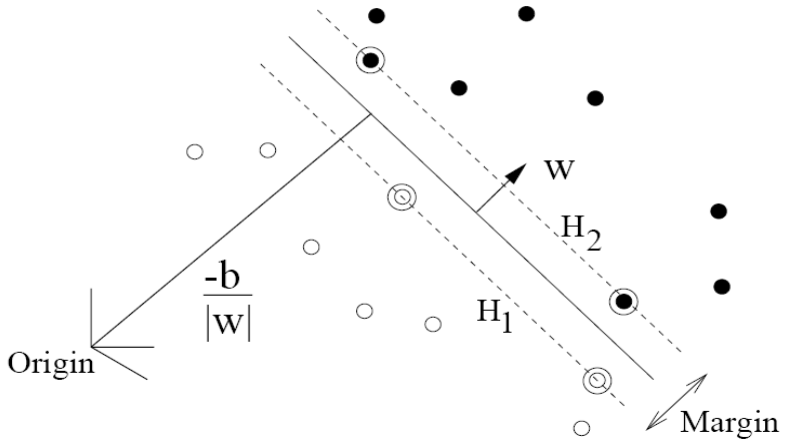

Figura 5.1: Solução para um cenário de 2 dimensões linearmente separável. Os pontos pelos quais a linha pontilhada passa são chamados de *vetores suporte*

Considere agora os exemplos para os quais a *igualdade* na Equação (5-4) é válida. Tais exemplos são pontos sobre o hiperplano $x_i \cdot w + b = 1$ com normal w e distância perpendicular a origem de $\frac{|1-b|}{\|w\|}$. Simetricamente, os exemplos para os quais a *igualdade* na Equação (5-5) é válida são pontos sobre o hiperplano $x_i \cdot w + b = -1$ com normal w e distância perpendicular a origem de $\frac{|-1-b|}{\|w\|}$.

Assim, $d^+ = d^- = \frac{1}{\|w\|}$ representa a largura da margem e possui valor $\frac{2}{\|w\|}$. Portanto, podemos encontrar os dois hiperplanos que geram a margem máxima minimizando $\|w\|^2$. Este é essencialmente o objetivo do algoritmo de Vetores Suporte e pode ser formulado como um problema de otimização quadrática sujeito as restrições definidas na Equação (5-6) como:

$$\min_{w,b} \quad \frac{1}{2}\|w\|^2 \tag{5-7}$$

A solução para o exemplo em 2 dimensões é mostrado na Figura (5.2). Os pontos de treinamento para os quais a igualdade na Equação (5-6) é válida são denominados *support vectors* e encontram-se sob a fronteira da margem.

Adotaremos agora uma formulação Lagrangeana para o problema. O principal objetivo é introduzir multiplicadores lagrageanos à cada restrição definida pela Equação (5-6). Como resultado, as restrições da Equação (5-6) são transferidas para os multiplicadores lagrangeanos facilitando assim a solucionar a Equação (5-7). Sem os multiplicadores lagrangeanos, a minimização da Equação (5-7) seria muito trabalhosa dado que w forma um produto escalar com x na Equação (5-6). Porém, com a introdução dos multiplicadores, temos um problema de otimização dual que será descrito na próxima seção.

Outro motivo para adotarmos uma formulação lagrangeana é que os dados de treinamento aparecem como um produto interno entre vetores, vide Equação (5-11). Isto permite que o algoritmo de Vetores Suporte seja generalizado para o caso em que os exemplos não são linearmente separáveis.

5.2.1 Formulação Lagrangeana

Introduzimos os multiplicadores lagrangeanos $\alpha_i, i = 1, 2, ..., D$, um para cada uma das restrições definidas na Equação (5-6). Subtraímos então este resultado da função objetivo definida pela Equação (5-7) e obtemos a seguinte formulação lagrangeana:

$$Ł(\alpha, w, b) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{D} \alpha_i [y_i(x_i \cdot w + b) - 1] \tag{5-8}$$

O primeiro passo no processo de otimização dual é minimizar a Equação (5-6) com relação a w e b. O segundo passo é maximizar o resultante com relação a $\alpha \geq 0$. Este problema de otimização dual é denominado *Wolfe dual* (05). Num ponto ótimo temos as seguintes equações de ponto de cela:

$$\frac{\partial L}{\partial b} = 0 \quad e \quad \frac{\partial L}{\partial w} = 0 \tag{5-9}$$

Estas equações se traduzem respectivamente em:

$$\sum_{i=1}^{D} \alpha_i y_i = 0 \quad e \quad w = \sum_{i=1}^{D} \alpha_i y_i x_i \tag{5-10}$$

Substituindo estes resultados na Equação (5-8) nos leva ao segundo passo do problema de otimização dual:

$$\begin{aligned} \max_{a} \quad & W(\alpha) = \sum_{i=1}^{D} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{D} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \\ sujeito\ a \quad & \alpha_i \geq 0, \quad i = 1, 2, ..., D \\ & \sum_{i=1}^{D} \alpha_i y_i = 0 \end{aligned} \tag{5-11}$$

Podemos solucionar este problema de maximização iterativamente usando o método gradiente descendente (tal solução é considerada ingênua uma vez que normalmente demora a convergir). Outros algoritmos de aprendizado são: Método de Osuna e otimização minimal seqüêncial (05). Num dado passo $k+1$ iteração, α é atualizado da seguinte maneira:

$$\alpha_i(k+1) = \alpha_i(k) + \eta \frac{\partial W}{\partial \alpha_i} \tag{5-12}$$

onde $\eta > 0$ é a "taxa de aprendizado". Uma vez feita a otimização, temos os coeficientes $\alpha_i, i = 1, 2, \cdots, D$ necessários para expressar a função de decisão. Portanto, dado uma representação vetorial de um documento de texto $z \in \mathbb{R}^v$, podemos inferir as classes da seguinte forma:

$$l(z) = sign \left[\sum_{i=1}^{D} y_i \alpha_i (z \cdot x_i) + b \right] \tag{5-13}$$

onde $l(z)$ é a classe a qual z pertence. Os vetores-suportes são os x_i tais que $\alpha > 0$. Completamos aqui a definição para do algoritmo de Vetores Suporte para dados linearmente separáveis. Apresentamos na próxima seção uma pequena alteração no procedimento para o cenário em que os dados que não são linearmente separáveis.

5.3 Máquinas de Vetores Suporte Não-Lineares

Para conjuntos de dados complexos e ruidosos como texto, dificilmente caímos no caso linearmente separável. Tal caso corresponde a um erro

empírico de zero. Na prática, precisamos buscar um compromisso entre o erro empírico e o "termo de complexidade" da Equação (5-3). Para atingir isso, relaxamos as restrições de margens rígidas da Equação (5-6) através da introdução de variáveis de folga:

$$\begin{aligned} \max_{a} \quad & W(\alpha) = \sum_{i=1}^{D} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{D} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \\ sujeito\ a \quad & 0 \leq \alpha_i \leq C, \quad i = 1, 2, ..., D \\ & \sum_{i=1}^{D} \alpha_i y_i = 0 \end{aligned} \tag{5-14}$$

Note que a única diferença entre o caso linearmente separável e o não-linearmente separável é a restrição nos multiplicadores lagrangeanos. A função de decisão definida na Equação (5-13) se mantém igual.

Para cada vetor suporte x_i existe um α_i associado tal que $0 < \alpha_i < C$. Portanto, para qualquer vetor suporte $x_i, i \in I := \{i : 0 < \alpha_i < C\}$ a seguinte igualdade é valida:

$$\begin{aligned} \max_{a} \quad & W(\alpha) = \sum_{i=1}^{D} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{D} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \\ sujeito\ a \quad & 0 \leq \alpha_i \leq C, i = 1, 2, ..., D \\ \sum_{i=1}^{D} \quad & \alpha_i y_i = 0 \end{aligned} \tag{5-15}$$

Às vezes pode ser útil mapear o espaço de entrada para um outro espaço não-linear de uma dimensão mais alta (i.e. $x \mapsto \Phi(x)$). Intuitivamente isto é o equivalente a "distorcer" o espaço geométrico ou inserir novas dimensões. O estudo desta técnica é denominado Funções Núcelo (*Kernel Functions*) porém está fora do escopo deste trabalho.

Concluímos aqui a apresentação do algoritmo de Vetores Suporte.