Projeto de Programação Não Linear

Support Vector Machine (SVM)

Alunos:
Vinicius Barcellos
Samuel Kutz
Matheus Morishita
Marcelo Baptista

2º Semestre de 2023

UFPR
UNIVERSIDADE FEDERAL DO PARANÁ

1 Introdução

No contexto de Machine Learning, modelos podem ser usados para dois tipos de problemas, *Regressão* ou *Classificação.* Em nosso trabalho, estaremos lidando com um modelo de Classificação, as **Máquinas de Vetores de Suporte** (SVM, do inglês *Support Vector Machine*).

A idéia por trás desse modelo é dividir um conjunto de dados em dois grupos que o modelo considerar distintos.

Vale ressaltar que esse modelo é de aprendizado não supervisionado, o que basicamente significa que o modelo vai tentar resolver o problema sózinho, e não sabemos se o que ele gerou esta certo ou não.

2 O problema

O meio pelo qual **SVM** tenta atingir o objetivo de classificar os dados é encontrando um *Hyperplano* que vai estar entre os dados de uma forma que, dados "acima" do hyperplano pertencem a um grupo enqunato que dados "abaixo" pertencem a outro, ou seja, teremos uma resposta binária, i.e. $y_i \in \{-1, 1\}$.

O problema surge do fato de que poderiamos ter infinítos hyperplanos entre os dados que poderiam dividi-los, ou seja, precisamos tentar encontrar o hyperplano ótimo, que melhor divide os dados.

2.1 Fundamentação Teórica

Support Vector Machine (SVM) é um modelo de classificação. O objetivo do modelo é encontrar um *hiperplano* que passa entre os pontos de um dataset, de maneira a separa-los, para isso o método utiliza-se de modelos de otimização para encontrar o "melhor" hiperplano possível, i.e. o *hiperplano ótimo.*

Os pontos mais importantes são aqueles mais próximos do hiperplano, portanto, todos os pontos que não estão pertos não são levados em conta. Esses pontos que são levados em consideração são chamados *Vetores de Suporte*, pois apenas eles que ajudam na busca pelo hiperplano ótimo.

Os tipos de pontos que estaremos lidando em nossos problemas serão do tipo:

$$\Omega = \{(x_i, y_i) \subset \mathbb{R}^p \times \{-1, 1\}; i = 1, 2, \cdots, n\}$$

Ou seja, como teremos apenas dois grupos, y_i pode apenas pertencer a $\{-1, 1\}$.

Hiperplano

Para o $\mathbb{R}^2$ uma reta pode ser escrito como:

$$y = ax + b \rightarrow ax + by = c$$

Para o $\mathbb{R}^3$ teriamos um plano, que pode ser escrito da forma:

$$ax + by + cz = d$$

Para generalizar para n dimensões poderiamos escrever um hiperplano da forma:

$$w_0 + w_1x_1 + w_2x_2 + \cdots + w_nx_n = 0$$

Chamando $w_0 = b$ teriamos:

$$w_1x_1 + w_2x_2 + \cdots + w_nx_n + b = 0$$

que também pode ser escrito da forma vetorial como:

$$w^Tx + b = 0 \equiv < w, x > +b = 0$$

onde $w, x \in \mathbb{R}^n$ e $b \in \mathbb{R}$

Quando eu sei que o Hiperplano Separa os Dois Conjuntos?

Quando um conjunto de pontos está "acima" do hyperplano, ele pertence a um grupo, quando está "abaixo" do hyperplano, ele pertence a outro. Como o conceito de "cima" e "baixo" não é tão intuitivo em n dimensões, podemos escrever matemáticamente como:

$$\begin{cases} w^Tx_i + b > 0 & \text{p/ } y_i = 1 \\ w^Tx_i + b < 0 & \text{p/ } y_i = -1 \end{cases} \tag{1}$$

2.2 Modelagem Matemática

Considere um hiperplano que separe os ponto no espaço:

$$w^Tx + b = 0$$

nós queremos a *Margem Máxima* entre o hiperplano e os pontos perto do hiperplano, ou sejá, queremos *Maximizar* a distância entre os pontos e o plano. Assim teremos que:

$$(w^Tx_i) + b \geq 1 \text{ se } y_i = 1 \tag{2}$$
$$(w^Tx_i) + b \leq -1 \text{ se } y_i = -1 \tag{3}$$

Queremos que a distância entre os pontos onde $y_i = 1$ e $y_i = -1$ seja igual, portanto pegamos a distância entre(2) e (3) como:

$$\frac{2}{\|w\|} = \frac{2}{\sqrt{w^Tw}}$$

ou seja, podemos modelar esse problema como:

$$\max_{w,b} \frac{2}{\|w\|}$$

porém, minimizar um elemento que está no denominador pode ser problemático númericamente, portanto podemos considerar o caso *Dual*, ou seja:

$$\max_{w,b} \frac{2}{\|w\|} \equiv \min_{w,b} \frac{w^T w}{2}$$

portanto, o problema de otimização que teremos de resolver será:

$$\begin{aligned} \min_{w,b} \quad & \frac{w^T w}{2} \\ \text{s.a.} \quad & y_i((w^T x_i) + b) \geq 1, \\ & i = 1, 2, \cdots \end{aligned}$$

2.3 Escolha de Algorítimo

3 Experimentos Numéricos

3.1 O algorítimo Implementado

3.2 Resultados Obtidos

4 Conclusões