UNIVERSIDADE ESTADUAL DE CAMPINAS
INSTITUTO DE MATEMÁTICA, ESTATÍSTICA E COMPUTAÇÃO CIENTÍFICA
DEPARTAMENTO DE MATEMÁTICA APLICADA

ANNA CLARA DE OLIVEIRA CLARO AMÂNCIO

Um estudo sobre Máquinas de Vetores de Suporte

Monografia apresentada ao Instituto de Matemática, Estatística e Computação Científica da Universidade Estadual de Campinas como parte dos requisitos para obtenção de créditos na disciplina Projeto Supervisionado, sob a orientação do(a) Prof. Dr. Marcos Eduardo Ribeiro do Valle Mesquita.

Campinas
2023

Resumo

Máquinas de Vetores de Suporte são algoritmos utilizados em Aprendizado de Máquina para classificação de padrões. Essa monografia tem como objetivo o estudo das Máquinas de Vetor Suporte Vetorial do ponto de vista matemático, entendendo a técnica como um problema de otimização quadrática.
Palavras Chave: Máquinas de Vetores de Suporte, Aprendizado de Máquina Supervisionado, Otimização Quadrática.

Abstract

Support Vector Machines are algorithms used in Machine Learning for pattern classification. This monograph aims to study Support Vector Machines from a mathematical point of view, understanding the technique as a quadratic optimization problem.
Keywords: Support Vector Machine, Supervised Learning, Quadratic Optimization.

Conteúdo

1 Introdução

O Aprendizado de Máquina é o campo de estudo que explora a construção de algoritmos capazes de "aprender"com seus erros e fazer previsões sobre dados para um dado problema.
Técnicas de aprendizado de máquina adquirem conhecimento do ambiente a partir de um processo de aprendizagem que por sua vez pode ser: supervisionado, não supervisionado, semi supervisionado ou por esforço. Iremos nos concentrar no aprendizado supervisionado, onde existe a presença de um 'professor' que apresenta exemplos de entrada e saída esperados. Os dados de treinamento são usados para construir um modelo do problema.
Nesse texto, iremos explorar as Máquinas de Vetores de Suporte que usam o aprendizado supervisionado para reconhecer e classificar padrões. Podemos usá-las por exemplo, para classificar se um tumor é benigno ou maligno ou classificar qual das espécies possíveis tal flor faz parte, levando em consideração o conjunto de dados estudado, dentre outros exemplos.
O trabalho está dividido em 4 seções. A primeira seção é a introdução. A segunda seção discorre sobre a fundamentação teórica do trabalho, onde é abordado o viés matemático das Máquinas de Vetores de Suporte, apresentamos as noções de hiperplano ótimo para padrões separáveis e não-separáveis linearmente e o truque do Kernel. Na terceira seção apresentamos o desenvolvimento e o resultado do nosso trabalho. A quarta e última seção conclui o texto.

2 Fundamentação Teórica

2.1 Introdução a Máquinas de Vetor Suporte Vetorial

As Máquinas de Vetores de Suporte (SVM) foram criadas, em 1995 por Vapnik, com o intuito de resolver classificações de padrão. Os resultados obtidos pelo método são comparáveis aos obtidos com outras técnicas de aprendizado de máquina como pelas Redes Neurais Artificiais, colocando as SVM como estado-da-arte para classificação de padrões.
A técnica é originalmente desenvolvida para classificação binária, ou seja, tem como objetivo encontrar um hiperplano como superfície de decisão que melhor divide as classes de interesse, ou seja, o hiperplano em que a separação entre as classes é máxima. Para tal feito, é necessário que os padrões sejam linearmente separáveis e, se não forem, é necessário um mapeamento para que os padrões se tornem linearmente separáveis. Podemos estender o processo para classificações multi-classes, onde há mais de duas classe de interesse, usando técnicas como um-contra-todos ou um-contra-um. Nesse trabalho, estudaremos apenas a classificação binária.

2.2 Hiperplano Ótimo para Padrões Linearmente Separáveis

Vamos considerar um conjunto de dados $\{(x_i, d_i)\}_{i=1}^{n}$ onde x_i é o padrão de entrada e d_i é a saída esperada para o i-ésimo exemplo. Vamos assumir que d_i assume valores em $\{-1, +1\}$ e que os subconjuntos de dados onde $d_i = +1$ e $d_i = -1$ são linearmente separáveis.
Podemos definir um hiperplano que realiza a separação entre esses subconjuntos por:

$$\langle w, x \rangle + b = 0$$

onde w é um vetor peso ajustável, x é um vetor de entrada, b é um bias e $\langle\rangle$ é o produto interno.
Dessa forma, podemos escrever:

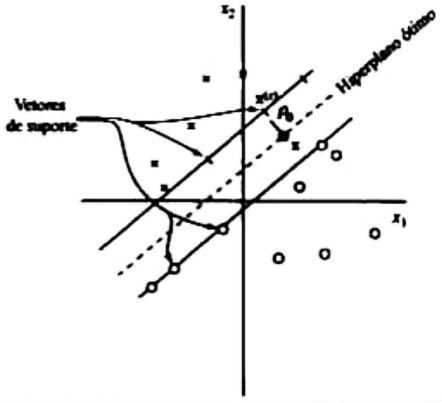
$$\langle w, x_i \rangle + b \geq 0 \quad \text{para } d_i = +1,$$

e

$$\langle w, x_i \rangle + b < 0 \quad \text{para } d_i = -1.$$

Podemos definir uma margem de separação ρ, região dentre o hiperplano até o ponto de dado mais próximo, para um dado vetor w e bias b. A figura a seguir apresenta a esquematização de um hiperplano ótimo para uma entrada bidimensional.

Figura 1: Hiperplano ótimo para padrões linearmente separáveis, onde x_1 e x_2 são as características, ρ_0 é metade da margem de separação.



Fonte: Simon Haykin. Redes Neurais: Princípios e Prática. Bookman, 2nd edition, 2008.

Vamos definir w_0 como o valor ótimo do vetor peso e b_0 como o valor ótimo do bias. Dessa forma, podemos escrever a equação do hiperplano como:

$$\langle w_0, x \rangle + b_0 = 0.$$

Vamos definir, também,

$$g(x) = \langle w_0, x \rangle + b_0 = 0$$

como a função que fornece a distância algébrica de x até o hiperplano ótimo e podemos escrever x como

$$x = x_p + r\frac{w_o}{\|w_0\|}$$

onde x_p é a projeção normal de x sobre o hiperplano ótimo e r é a distância algébrica desejada. Por definição, $g(x_p) = 0$, assim:

$$g(x) = \langle w_0, x \rangle + b_0 = \langle w_0, x_p + r\frac{w_o}{\|w_0\|} \rangle + b_0 =$$

$$\langle w_0, x_p \rangle + r\frac{\langle w_0, w_0 \rangle}{\|w_0\|} + b_0 =$$

$$0 + r\frac{\langle w_0, w_0 \rangle}{\sqrt{\langle w_0, w_0 \rangle}} = r\sqrt{\langle w_0, w_0 \rangle} = r\|w_0\|.$$

Portanto, podemos escrever:

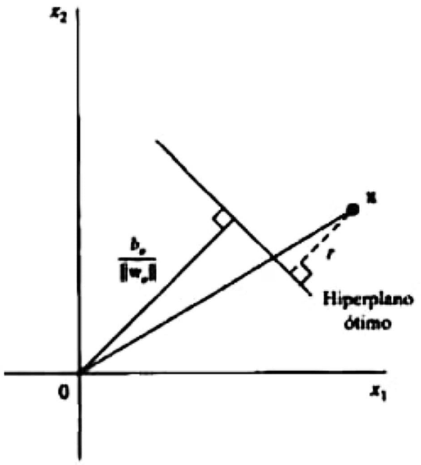
$$r = \frac{g(x)}{\|w_0\|}.$$

Note que a distância da origem ao hiperplano é dada por:

$$r = \frac{g(x)}{\|w_0\|} = \frac{\langle w_0, x\rangle + b_0}{\|w_0\|} = \frac{\langle w_0, 0\rangle + b_0}{\|w_0\|} = \frac{b_0}{\|w_0\|}.$$

Dessa forma, temos que se $b_0 > 0$, a origem está do lado positivo do hiperplano, se $b_0 < 0$ a origem está do lado negativo do hiperplano e se $b_0 = 0$ o hiperplano intercepta a origem. Na figura 2, podemos ilustrar essa distância:

Figura 2: Distância da origem ao hiperplano ótimo, onde x_1 e x_2 são as características.



Fonte: Simon Haykin. Redes Neurais: Princípios e Prática. Bookman, 2nd edition, 2008.

Assumindo que nossos padrões são linearmente separáveis, nosso problema de encontrar os parâmetros w_0 e b_0 do hiperplano ótimo deve satisfazer as seguintes restrições:

$$\langle w_0, x_i\rangle + b_0 \geq 1 \quad \text{para } d_i = +1,$$
$$\langle w_0, x_i\rangle + b_0 < -1 \quad \text{para } d_i = -1.$$

Os pontos que obedecem as restrições acima com sinal de igualdade são chamados de Vetores de Suporte. Esses são os pontos mais próximos à superfície de decisão, e portanto, são os pontos de classificação mais complicada.
A distância algébrica do vetor suporte à superfície de decisão é, portanto, dada por:

$$r = \frac{g(x)}{\|w_0\|} = \frac{1}{\|w_0\|} \quad \text{para } d_i = +1,$$
$$r = \frac{g(x)}{\|w_0\|} = \frac{-1}{\|w_0\|} \quad \text{para } d_i = -1.$$

Agora, considere ρ o valor ótimo da margem de separação entre duas classes,

$$\rho = 2r = \frac{2}{\|w_0\|}.$$

Com isso, nosso problema se resume a maximizar a margem de separação entre as classes e por equivalência, nosso problema se resume em minimizar a norma euclidiana de w.

2.2.1 O Problema Primal

Dada uma amostra de treinamento, $\mathbf{C} = \{(x_i, d_i)\}_{i=1}^{n}$ onde x_i é o padrão de entrada e d_i é a saída esperada para o i-ésimo exemplo, queremos:

$$\text{minimizar} \quad \frac{1}{2}\langle w, w\rangle$$

$$\text{sujeito a:} \quad d_i(\langle w, x_i\rangle + b) \geq 1, \qquad \text{para } i = 1, 2, \cdots, n$$

onde $\frac{1}{2}\langle w, w\rangle$ é uma equação convexa e as restrições são lineares em relação a w.

2.2.2 O Problema Dual

Podemos usar o método dos multiplicadores de Lagrange para resolver o problema proposto.
Vamos introduzir a seguinte função lagrangiana:

$$J(w, b, \alpha) = \frac{1}{2}\langle w, w\rangle - \sum_{i=1}^{n} \alpha_i[d_i(\langle w, x_i\rangle + b) - 1]$$

onde os α_i são as variáveis chamadas multiplicadores de Lagrange. Dessa forma, o problema passa a ser minimizar a função lagrangiana em relação a w, b e a maximização dos α_i. O ponto de sela da função lagrangiana determina a solução ótima, dessa forma, temos as seguintes condições de otimização:

$$\frac{J(w, b, \alpha)}{\delta w} = 0,$$

e

$$\frac{J(w, b, \alpha)}{\delta b} = 0.$$

Podemos escrever a função lagrangiana como:

$$J(w, b, \alpha) = \frac{1}{2}\langle w, w\rangle - \sum_{i=1}^{n} \alpha_i[d_i(\langle w, x_i\rangle + b) - 1] =$$

$$\frac{1}{2}\langle w, w\rangle - \sum_{i=1}^{n} \alpha_i d_i \langle w, x_i\rangle - b\sum_{i=1}^{n} \alpha_i d_i + \sum_{i=1}^{n} \alpha_i.$$

Das condições de otimização, temos:

$$\sum_{i=1}^{n} \alpha_i d_i = 0,$$

e

$$w = \sum_{i=1}^{n} \alpha_i d_i x_i,$$

Logo,

$$\langle w, w\rangle = \sum_{i=1}^{n} \alpha_i d_i \langle w, x_i\rangle = \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j d_i d_j \langle x_i, x_j\rangle.$$

Dessa forma, a função objetivo do problema dual pode ser escrita como uma função $Q(\alpha)$:

$$J(w,b,\alpha) = \frac{1}{2}\langle w,w\rangle - \sum_{i=1}^{n}\alpha_i d_i\langle w,x_i\rangle - b\sum_{i=1}^{n}\alpha_i d_i + \sum_{i=1}^{n}\alpha_i,$$

$$Q(\alpha) = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j d_i d_j\langle x_i,x_j\rangle.$$

No ponto de sela da função lagrangiana, para cada α_i, a restrição $\alpha_i[d_i(\langle w,x_i\rangle + b) - 1] = 0$ para $i = 1,2,\cdots\ n$ e, uma vez que $d_i(\langle w,x_i\rangle + b) - 1 = 0$ os α que satisfazem a restrição são não nulos.
O problema dual pode então ser escrito como:

$$\text{maximizar} \quad \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j d_i d_j\langle x_i,x_j\rangle$$

$$\text{sujeito a:} \quad \sum_{i=1}^{n}\alpha_i d_i = 0 \quad \text{e} \quad \alpha_i \geq 0 \quad \text{para } i = 1,2,\cdots,n.$$

Após calculados os $\alpha_{0,i}$, que seriam os α ótimos, podemos computar w e b ótimos, dados por:

$$w_0 = \sum_{i=1}^{n}\alpha_{0,i}d_i x_i,$$

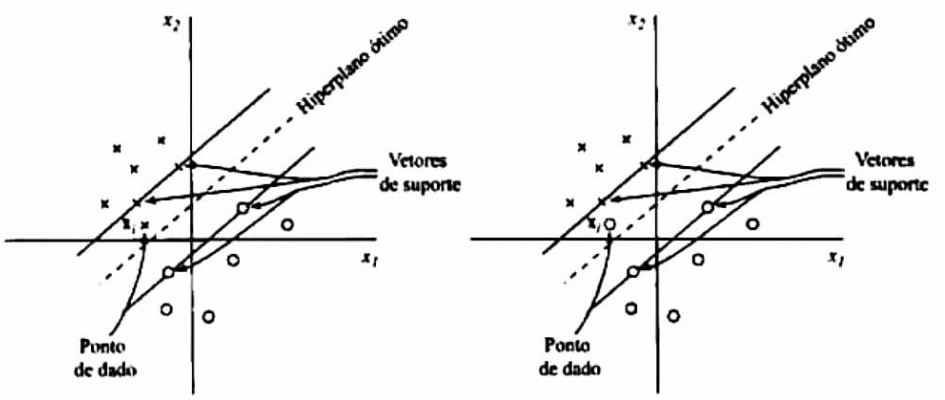
$$b_0 = 1 - \langle w_0, x^{(s)}\rangle \quad \text{para} \quad d^{(s)} = 1,$$

onde $x^{(s)}$ representam os pontos dos vetores de suporte positivo.

2.3 Hiperplano Ótimo para Padrões Lineares Não-Separáveis

Para padrões não separáveis, podemos construir uma superfície de decisão mas encontraremos erros de classificação. Nesse caso, desejamos encontrar um hiperplano ótimo em que o erro de classificação seja minimizado.
Na figura 3 são retratadas duas violações de pontos que não satisfazem $d_i(\langle w,x_i\rangle + b) \geq 1, \quad \text{para} \quad i = 1,2,\cdots,n.$

Figura 3: À esquerda, o ponto dado se encontra dentro da margem de separação e a classificação é correta, à direita o ponto dado se encontra dentro da margem de separação, e ainda mais, se encontra do lado errado da superfície de decisão e sua classificação está incorreta. Na imagem, x_1 e x_2 são as características.



Quando algum dos casos retratados na figura acima acontecem, dizemos que a margem de separação entre as classes é suave.
Para modelarmos o problema de otimização para margens suaves, introduzimos, na definição da superfície de decisão, variáveis de folga, que são escalares e não negativas, $\{\xi_i\}_{i=1}^n$:

$$d_i(\langle w, x_i\rangle + b) \geq 1 - \xi_i \quad \text{para} \quad i = 1, 2, \cdots, n.$$

As variáveis de folga medem o desvio de um ponto de dado da condição ideal de separabilidade de padrões, assim se $1 \leq \xi_i \leq 0$, o ponto dado está dentro da região de separação, e se $\xi_i > 1$ ele se encontra do lado errado da superfície de decisão. Os vetores de suporte são definidos da mesma forma do caso linearmente separável.

2.3.1 O Problema Primal

$$\text{minimizar} \quad \frac{1}{2}\langle w, w\rangle + \frac{C}{n}\sum_{i=1}^{n}\xi_i$$

$$\text{sujeito a:} \quad d_i(\langle w, x_i\rangle + b) \geq 1 - \xi \quad \text{e} \quad \xi_i \geq 0 \quad \text{para } i = 1, 2, \cdots, n$$

onde o parâmetro C é um parâmetro de regularização positivo e decidido empiricamente pelo usuário.

2.3.2 O Problema Dual

Podemos formular o problema dual, seguindo o mesmo processo para os padrões separáveis, usando os multiplicadores de Lagrange.

$$\text{maximizar} \quad \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j d_i d_j\langle x_i, x_j\rangle$$

$$\text{sujeito a:} \quad \sum_{i=1}^{n}\alpha_i d_i = 0 \quad \text{e} \quad 0 \leq \alpha_i \leq C \quad \text{para } i = 1, 2, \cdots, n$$

onde o parâmetro C é um parâmetro de regularização positivo e decidido empiricamente pelo usuário.
Após calculados os $\alpha_{0,i}$, que seriam os α ótimos, podemos computar w e b ótimos, dados por:

$$w_0 = \sum_{i=1}^{n_s}\alpha_{0,i} d_i x_i$$

onde n_s representa o número de vetores suporte.
O bias ótimo é computado levando em conta as restrições:

$$\alpha_i[d_i(\langle w, x_i\rangle) - 1 + \xi] = 0, \quad \text{para} \quad i = 1, 2, \cdots, n$$

$$\mu_i\xi_i = 0 \quad \text{para} \quad i = 1, 2, \cdots, n.$$

onde μ_i são os multiplicadores de Lagrange introduzidos para forçar a não negatividade de ξ_i para todo i. Levando em conta o problema primal, no ponto de sela, a derivada da função lagrangiana em relação a ξ_i é zero, com isso $\alpha_i + \mu_i = C$ e combinando as restrições, temos que $\xi_i = 0$ para $\alpha_i < C$.
Assim, o bias pode ser calculado usando qualquer dado do conjunto de dados onde $0 < \alpha_{0,i} < C$ e por consequência, $\xi_i = 0$, mas, numericamente é mais interessante usar a média de todos os dados que verificam essa restrição.

2.4 Curva de Decisão para Padrões Não-Lineares

Podemos nos deparar com casos em que os padrões são não-lineares e com isso não conseguimos traçar um hiperplano ótimo capaz de dividir as classes estudadas. Assim, vamos definir, um espaço de entradas que é o espaço onde os dados estão e um espaço de características que é um espaço de dimensão maior que a do espaço de entradas. Vamos mapear os dados no espaço de entradas ao espaço de características através de uma transformação.
Considere o conjunto de dados $\mathbf{C} = \{(x_i, d_i)\}_{i=1}^{n}$, onde x_i é o padrão de entrada e d_i é a saída esperada para o i-ésimo exemplo e considere $\mathbf{P} = \{\phi_i\}_{j=1}^{m}$ o conjunto das transformações não-lineares que vão do espaço de entradas ao espaço de características. Vamos assumir que ϕ_j é definido para todo j. Podemos definir a superfície de decisão, no espaço das características, como uma equação de hiperplano dada por:

$$\sum_{j=1}^{m} w_j \phi_j(x) + b = 0$$

onde $\{w_j\}_{j=1}^{m}$ é o conjunto de pesos lineares e b é o bias. Para simplificação, vamos assumir $\phi_0(x) = 1$ para todo x, daí:

$$\sum_{j=0}^{m} w_j \phi_j(x) = 0$$

e assim, w_0 representa o bias b. Vamos definir o vetor:

$$\phi(x) = [\phi_0(x), \phi_1(x), \cdots, \phi_m(x)]^t$$

onde $\phi_j(x)$ representa a entrada fornecida por w_j através do espaço de características, ou seja, podemos enxergar o vetor $\phi(x)$ como uma "imagem"induzida no espaço de características dado o vetor de entrada x. Agora, reescrevendo

$$w = \sum_{i=1}^{n} \alpha_i d_i x_i,$$

de acordo com esse novo contexto, temos:

$$w = \sum_{i=1}^{n} \alpha_i d_i \phi(x_i)$$

onde $\phi(x_i)$ corresponde ao padrão de entrada x_i no i-ésimo exemplo. Podemos, também, reescrever a superfície de decisão como:

$$w^t \phi(x) = 0$$

e assim,

$$w = \sum_{i=1}^{n} \alpha_i d_i \phi^t(x_i)\phi(x) = 0$$

onde $\phi^t(x_i)\phi(x)$ é o produto interno de dois vetores induzidos no espaço de características. Vamos então introduzir o Kernel $K(x, x_i)$ definido por:

$$K(x, x_i) = \phi^t(x_i)\phi(x) = \sum_{j=0}^{m} \phi_j(x)\phi_j(x_i) \quad \text{para} \quad i = 1, 2, \cdots, n$$

e da definição:

$$K(x, x_i) = K(x_i, x) \quad \text{para todo} \quad i.$$

Por fim, o hiperplano pode ser escrito como:

$$\sum_{i=1}^{n} \alpha_i d_i K(x, x_i) = 0.$$

2.4.1 O Problema Dual

Para padrões não-lineares podemos formular o problema dual como:

$$\text{maximizar} \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j d_i d_j K(x, x_i)$$

$$\text{sujeito a:} \quad \sum_{i=1}^{n} \alpha_i d_i = 0 \quad \text{e} \quad 0 \leq \alpha_i \leq C \quad \text{para } i = 1, 2, \cdots, n.$$

onde o parâmetro C é um parâmetro de regularização positivo e decidido empiricamente pelo usuário. Após calculados os $\alpha_{0,i}$, que seriam os α ótimos, podemos computar w e b ótimos, dados por:

$$w_0 = \sum_{i=1}^{n} \alpha_{0,i} d_i \phi(x_i)$$

onde $\phi(x_i)$ é a imagem induzida no espaço das características. E o bias ótimo b_0 é a primeira componente de w_0.

2.4.2 Kernels mais Utilizados

Os Kernels mais utilizados são: o Kernel Polinomial, o Kernel Gaussiano e o Kernel Sigmoidal.
O Kernel Polinomial é dado por:

$$K(x, x_i) = (x^t x_i + 1)^p,$$

onde a potência p é escolhida pelo usuário.
O Kernel Gaussiano é dado por:

$$K(x, x_i) = e^{(-\frac{1}{2\sigma^2}\|x - x_i\|^2)},$$

onde σ^2 é escolhida pelo usuário.
O Kernel Sigmoidal é dado por:

$$K(x, x_i) = tanh(\beta_0 x^t x_i + \beta_1),$$

utilizado somente para alguns valores de β_0 e β_1.

3 Desenvolvimento e Resultados

Foram estudados dois conjuntos de dados: $\mathbf{C}_1$ e $\mathbf{C}_2$, construídos conforme as figuras 4 e 5 a seguir:

Figura 4: Construção do primeiro conjunto de dados, $\mathbf{C}_1$.

```
X1, d1 = make_blobs(n_samples=20, centers=2, n_features=2,random_state=0)
d1 = 2*d1-1
```

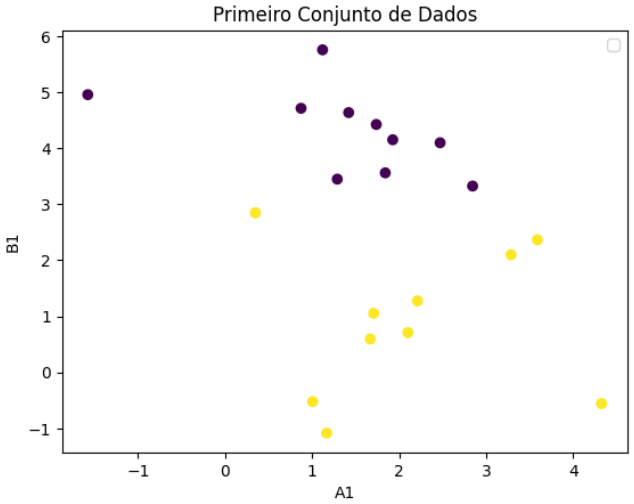
Figura 5: Construção do segundo conjunto de dados, $\mathbf{C}_2$.

```
X2, d2 = make_blobs(n_samples=100, centers=2, cluster_std =4 , n_features=2,random_state=17)
d2 = 2*np.mod(d2,2) -1
```

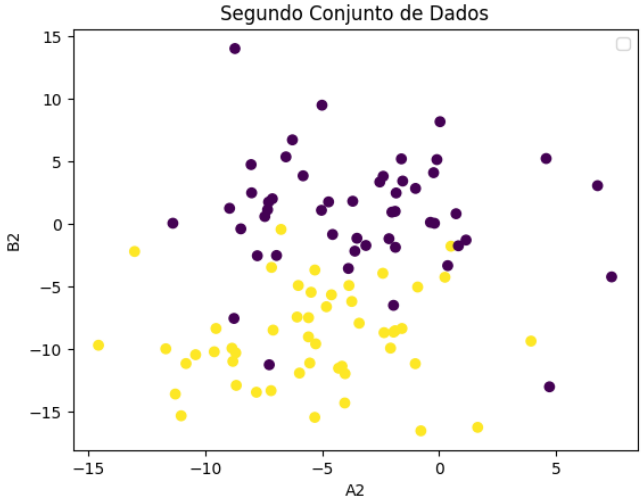
As figuras 6a, 6b a seguir ilustram graficamente os conjunto de dados estudados:

Figura 6: Gráfico dos Dados.

(a) Gráfico do primeiro conjunto de dados, onde A1 e B1 são as características e os pontos amarelos representam os pontos que possuem uma dada classificação e os pontos azuis representam os pontos que possuem a outra classificação.



(b) Gráfico do primeiro conjunto de dados, onde A2 e B2 são as características e os pontos amarelos representam os pontos que possuem uma dada classificação e os pontos azuis representam os pontos que possuem a outra classificação.

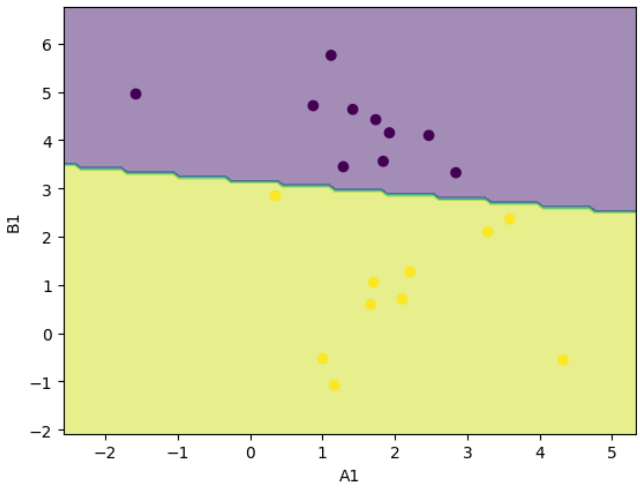


3.1 Resultados

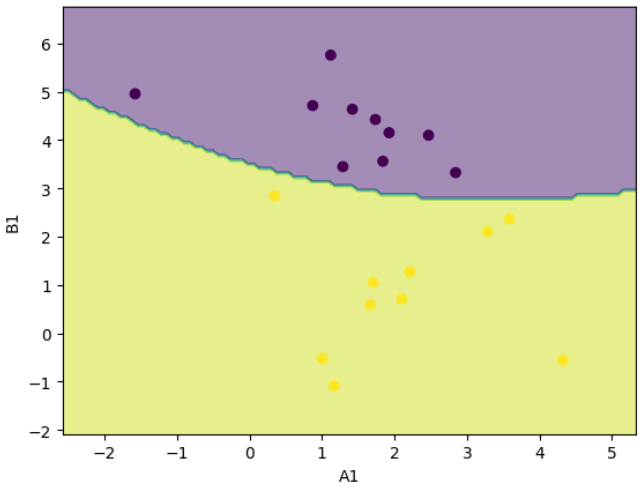
Utilizamos a biblioteca *Scikit-learn*, biblioteca que de aprendizado de máquina, para analisar os conjuntos de dados apresentados.
A seguir, as figuras 7a, 7b, 7c e 7d apresentam os resultados obtidos para o primeiro conjunto de dados.

Figura 7: Hiperplanos encontrados para o primeiro conjunto de dados onde A1 e A2 são características.

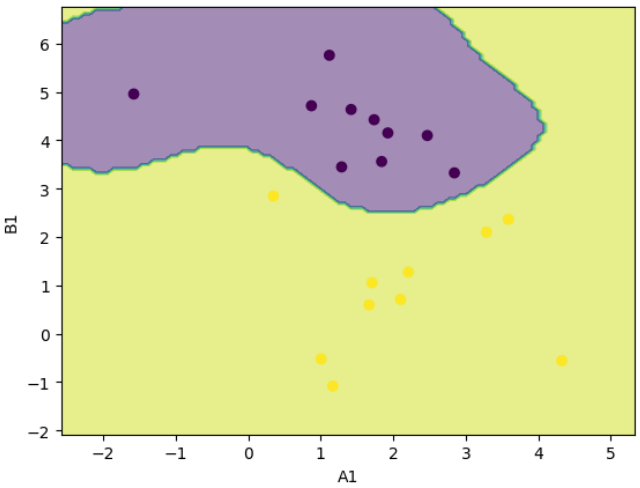
(a) Superfície de Decisão obtida para o primeiro conjunto de dados utilizando o kernel linear.



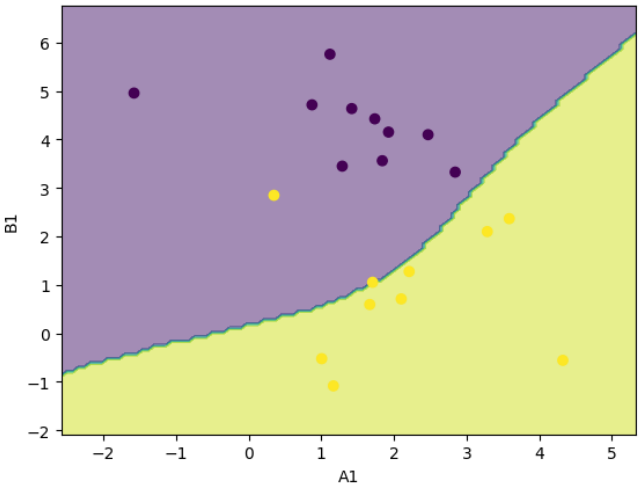
(b) Superfície de Decisão obtida para o primeiro conjunto de dados utilizando o kernel polinomial com grau p = 3.



(c) Superfície de Decisão obtida para o primeiro conjunto de dados utilizando o kernel gaussiano com gamma = $\frac{1}{2\sigma^2}$ = 0.5.



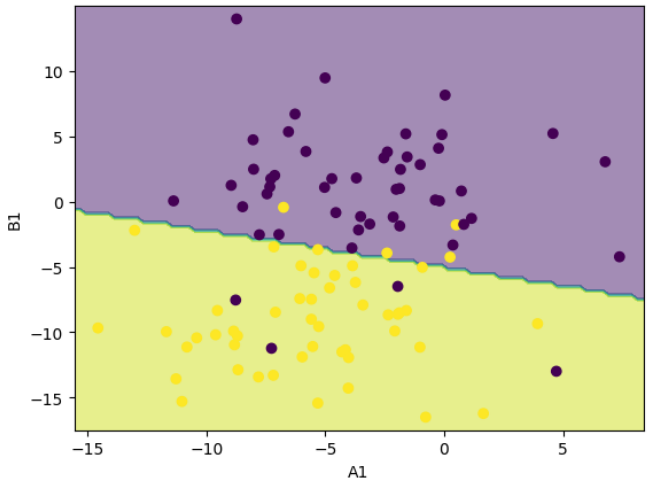
(d) Superfície de Decisão obtida para o primeiro conjunto de dados utilizando o kernel sigmoidal com $\beta_1 = 9.27$.



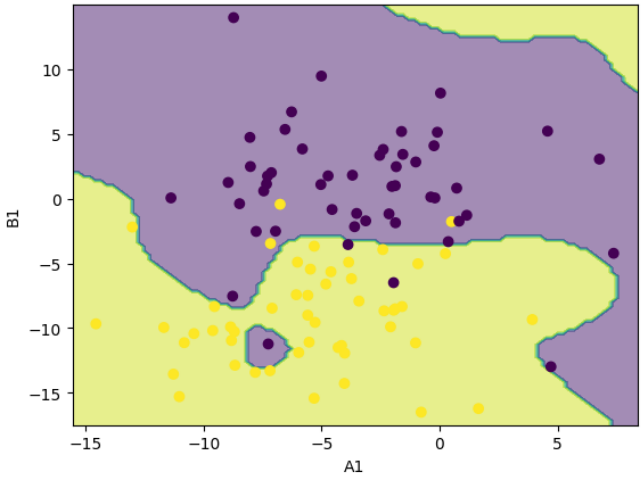
Os parâmetros para cada kernel foram escolhidos empiricamente no objetivo de obter a maior acurácia, para os kernels linear, polinomial e gaussiano a acurácia foi 1, para o kernel sigmoidal a acurácia foi de 0.9.
A seguir, as figuras 8a, 8b, 8c e 8d apresentam os resultados obtidos para o primeiro conjunto de dados.

Figura 8: Hiperplanos encontrados para o segundo conjunto de dados.

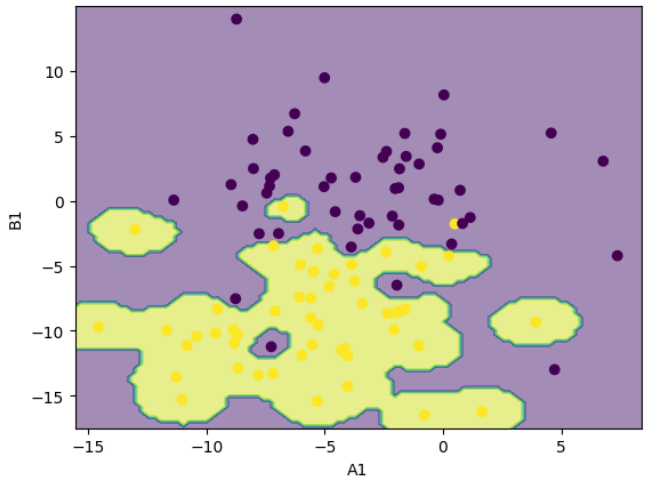
(a) Superfície de Decisão obtida para o segundo conjunto de dados utilizando o kernel linear.



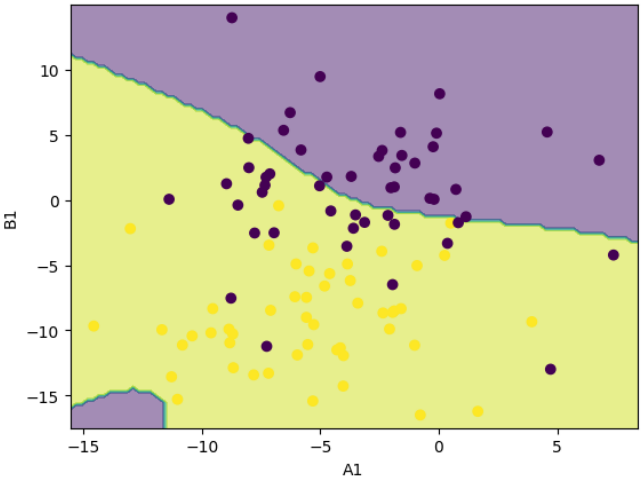
(b) Superfície de Decisão obtida para o segundo conjunto de dados utilizando o kernel polinomial com grau 8.



(c) Superfície de Decisão obtida para o segundo conjunto de dados utilizando o kernel gaussiano com gamma $= \frac{1}{2\sigma^2} = 0.5$.



(d) Superfície de Decisão obtida para o segundo conjunto de dados utilizando o kernel sigmoidal com $\beta_1 = 1$.



Os parâmetros para cada kernel foram escolhidos empiricamente no objetivo de obter a maior acurácia. A acurácia do kernel liner, polinomial, gaussiano e sigmoidal foram, respectivamente: 0.92, 0.95, 0.99 e 0.74.

3.2 Criação e Implementação de um Código Próprio

Estudamos, também, o primeiro conjunto de dados através de um código próprio construído. Abordamos somente o kernel linear.
Implementamos, em *Python*, um código que utiliza as bibliotecas *Cvxpy* e *Cvxopt* criadas para otimização convexa.

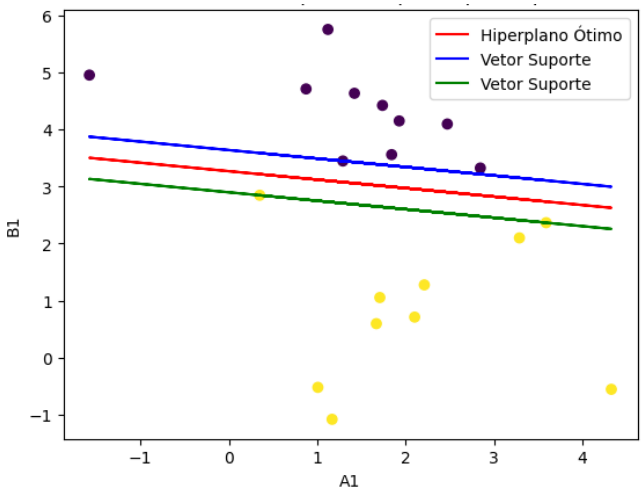
Utilizamos a otimização quadrática, uma vez que podemos escrever o problema primal e dual da forma:

$$\begin{aligned} \min \quad & \frac{1}{2}x^t P x + q^t x \\ \text{sujeito a:} \quad & Gx \leq h \text{ e } Ax = b \end{aligned}$$

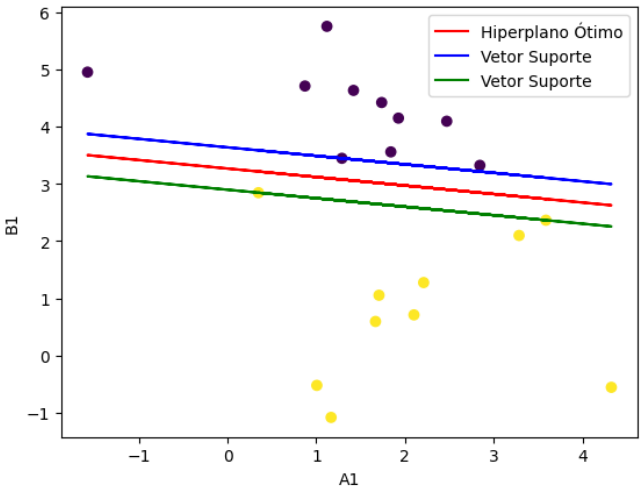
onde, $P \in \mathcal{S}_+^n$, $q \in \mathcal{R}^n$, $G \in \mathcal{R}^{m \times n}$, $h \in \mathcal{R}^m$, $A \in \mathcal{R}^{p \times n}$, $b \in \mathcal{R}^p$ e $x \in \mathcal{R}^n$.
Para o primeiro conjunto de dados, obtivemos os seguintes resultados, expostos nas figuras 9a, 9b, 9c e 9d:

Figura 9: Hiperplanos encontrados para o primeiro conjunto de dados.

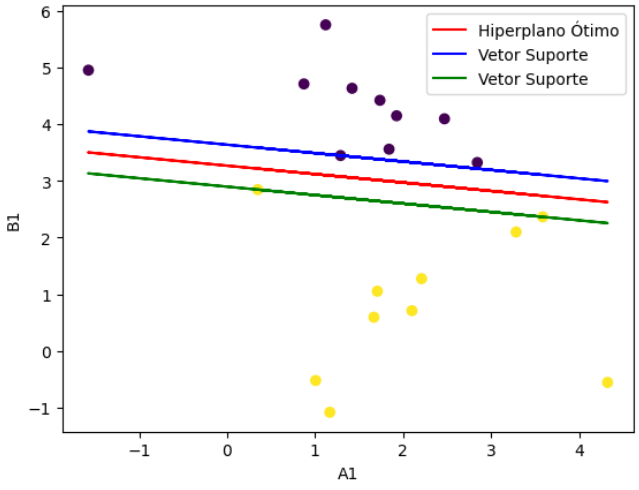
(a) Problema Primal onde A1 e B1 são características.



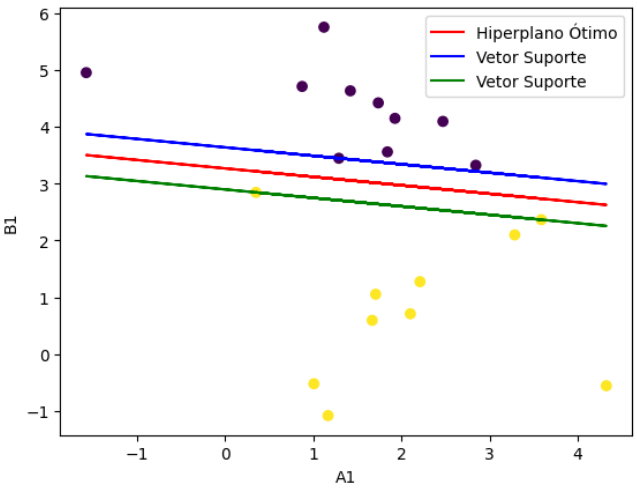
(b) Problema Primal com Variável de Folga e c=10 onde A1 e B1 são características.



(c) Problema Dual onde A1 e B1 são características.



(d) Problema Dual com Variável de Folga e c=10 onde A1 e B1 são características.



A acurácia obtida para as figuras acima foi de 1.

4 Conclusão

Podemos concluir que as Máquinas de Vetor de Suporte possuem grande fundamento teórico matemático. A técnica é capaz de classificar padrões com alta taxa de eficácia mesmo que o conjunto de dados possua

grande dimensionalidade.
Para a melhor classificação dos dados é preciso uma boa escolha de kernel e dos parâmetros envolvidos.
Em comparação a outras técnicas de redes neurais, a técnica pode ser mais devagar computacionalmente uma vez que a busca da solução do problema de otimização se complica à medida que os número de dados aumenta.
Mesmo com ressalvas, as Máquinas de Vetor de Suporte á uma das técnicas mais usadas atualmente por conta de sua robustez e base teórica bem fundamentada.

Referências

[1] Simon Haykin. *Redes Neurais: Princípios e Prática*. Bookman, 2nd edition, 2008.

[2] Bernhard Schölkopf and Alexander J. Smola. *Learning with kernels : support vector machines, regularization, optimization, and beyond*. MIT Press, 2002.