Support Vector Machines
SVM

Gabriel Lechenco V. Pereira

1

1. Introdução

SVM se trata de uma técnica de machine learning e classificação que tem ganhando força nos últimos anos. Esse crescimento em sua popularidade se deve muito por sua eficiência para problemas de classificação linear, oferecendo respostas muitas vezes melhores que Redes Neurais Artificiais, além da evolução de seus kernels, possibilitando a resolução de problemas não-lineares. [Williams 2008]

2. Solução Linear

Considerando uma base de dados com duas classes distintas (indicadas por um coeficiente $+1$ ou -1, por exemplo), busca-se encontrar um classificador que, ao se deparar com um novo dado de mesma natureza, consiga indicar a qual classe aquele dado pertence. Supondo que estes dados sejam linearmente separáveis, podemos propor uma função $f(x)$ cujo o sinal $sgn(f(x))$ indique a qual classe o dado x pertence.

Em sua fase de aprendizagem, uma SVM busca encontrar uma divisão linear entre as duas classes da base de dados, traçando uma reta (ou hiperplano $\hat{w}$) nesta divisão. Desse modo, podemos expressar $f(x)$ pela Equação 1. [Williams 2008]

$$f(x) = \hat{w}x + w_0 \tag{1}$$

2.1. Aprendizagem

Apesar de parecer uma tarefa simples, é importante observar qual a melhor forma de traçar esta divisa. Pode haver um grande conjunto de retas com as quais é possível separar os dados corretamente, porém, caso a escolha seja arbitrária, ao inserir novos dados no classificador para testes, este pode obter respostas errôneas (Figura 1).

Para obter o melhor resultado, a SVM utiliza outros dois hiperplanos para suporte, sendo estes dois paralelos ao hiperplano central e passam sobre o representante de cada classe que se encontra mais próximo a $\hat{w}$. Estes vetores de suporte podem ser chamados de H_+ e H_- e podem ser observados na Figura 2

Para se obter a melhor opção devemos maximizar a margem da solução, ou seja, devemos buscar a maior distancia possível entre os vetores de suporte H_+ e H_- e o hiperplano $\hat{w}$. Após um breve raciocínio, fica evidente que a melhor solução deve ser aquela onde as distâncias d_+ e d_- devem ser iguais. [Williams 2008]

Como pode ser observado na Figura 2, os pontos que tem mais influencia sobre os vetores de suporte são os que se encontram mais perto dos mesmos. Dessa forma, a grande maioria dos dados que caracterizam cada uma das classes podem ser desconsiderados para encontrar a solução. Assim, podemos otimizar $\hat{w}$ utilizando multiplicadores de Lagrange e programação quadrática, obtendo a Equação 2.

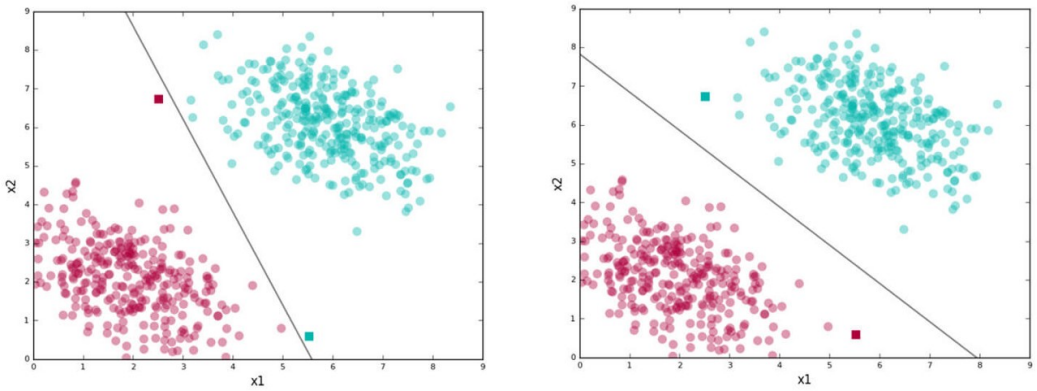

Figure 1. Possíveis retas de separação para uma amostra de dados [Ghose 2017]

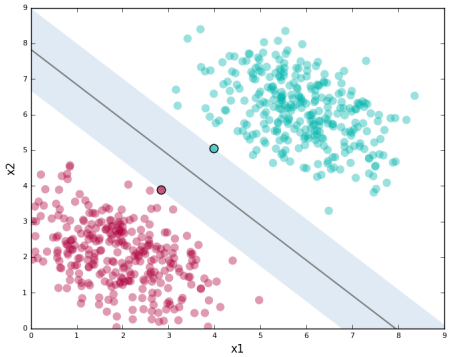

Figure 2. Exemplo de vetores de suporte paralelos a $\hat{w}$ [Ghose 2017]

$$\hat{w} = \sum \alpha_i y_i x_i \tag{2}$$

Onde α_i's são em sua maioria iguais a zero, exceto para aqueles próximos dos vetores de suporte e do limiar de decisão, e y_i refere-se a Equação 3.

$$y_i(x_i\hat{w} + w_0) \geq 1 \tag{3}$$

Este problema para encontrar os coeficientes α_i's para este caso se trata de um problema de otimização convexa, ou seja, um problema sem mínimos locais, isso é o que faz com que as SVM's sejam tão boas em classificar dados linearmente separáveis.

2.2. Predição

Ao conseguir representar $\hat{w}$ com a Equação 3 é possível treinar o SVM a partir de uma base de dados. Porém, o problema não se trata apenas de classificar os dados, mas o algoritmo deve encontrar uma solução capaz de predizer e classificar novos dados de entrada. Para isso podemos utilizar uma função $g(x)$ definida pelas Equações 4 e 5.

$$g(x) = sgn(\hat{w}x + w_0) \tag{4}$$

$$g(x) = sgn(\sum \alpha_i y_i (x_i.x) + w_0) \tag{5}$$

3. Kernel

Entretanto, como podemos utilizar essa técnica de classificação para dados que não linearmente separáveis? Para estes casos podemos utilizar uma característica bem interessante das SVM's, o kernel. O Kernel consiste em diversos grupos de equações que podem ser utilizados para trocar o espaço onde os dados se encontram, podendo inclusive adicionar novas dimensões. Este truque parte do teorema de Cover, que diz que um conjunto de dados tem mais chances de ser linearmente separável quando projetado em dimensões mais altas. [Williams 2008]

Essa projeção pode ser definida por uma função $\phi(x)$, a qual irá ilustrar os dados de um plano $\mathbb{R}^d$ para um plano F com dimensão N_F. Logo, a função de kernel de dois dados, x_i e x_j pode ser ilustrado pela Equação 6.

$$k(x_i, x_j) = \phi(x_i).\phi(x_j) \tag{6}$$

Existem diversos tipos de kernels, então devemos escolher o que melhor se adapta com a base de dados em questão. Alguns exemplos de kernels são os que tratam e classificam equações polinomiais, logarítmicas e sigmóides.

Com isso, a função de predição $g(x)$, descrita pela Equação 5 pode ser adaptada para a Equação 7. Para este caso, a otimização de α_i's se torna um problema de programação quadrático.

$$g(x) = sgn(\sum \alpha_i y_i k(x_i, x) + w_0) \tag{7}$$

Um problema da utilização do kernel é que, com o aumento das dimensões, mais dados serão necessários para uma boa definição dos hiperplanos. Além disso, essa projeção dos dados pode acabar sendo muito custosa, devido a complexidade do kernel escolhido.

4. Dados não separáveis

Para problemas reais, há uma dificuldade muito grande em encontrar dados que podem ser completamente separados, o mais provável é que se encontre algo similar a Figura 3 onde não é possível definir uma reta que divida completamente as duas classes.

Para contornar este problema, podemos adicionar um coeficiente C que determina uma taxa de contribuições relativas nos erros que serão aceitos pelo algoritmo. Para isso consideramos uma "fatia" $\chi_i \geq 0$ a qual define a distância entre um ponto discrepante e o seu vetor de suporte. Logo, esta variável vai ter o valor zero exceto para os pontos que se encontram entre os vetores H_+ e H_- ou que estão do lado errado da classificação.

O valor de C indica o tamanho da margem e qual a taxa de erro que será considerada, o que pode ser observado na Figura 4. Este deve ser ajustado de forma que se encontre o melhor resultado para a solução proposta.

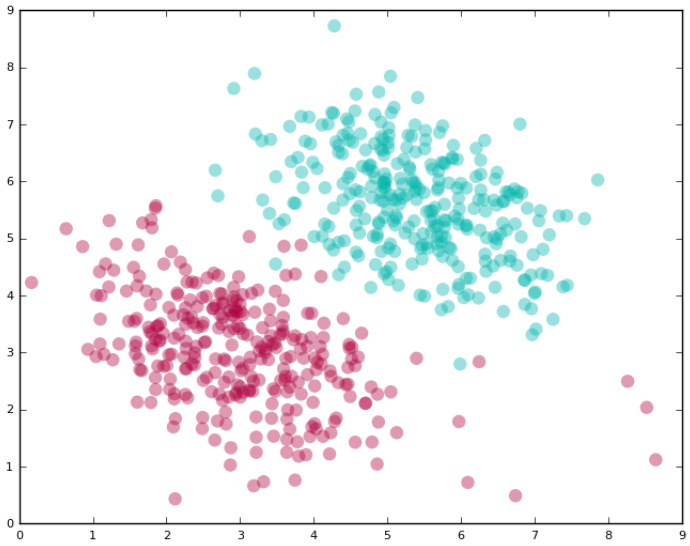

Figure 3. Exemplo de dados que não são totalmente separáveis [Ghose 2017]

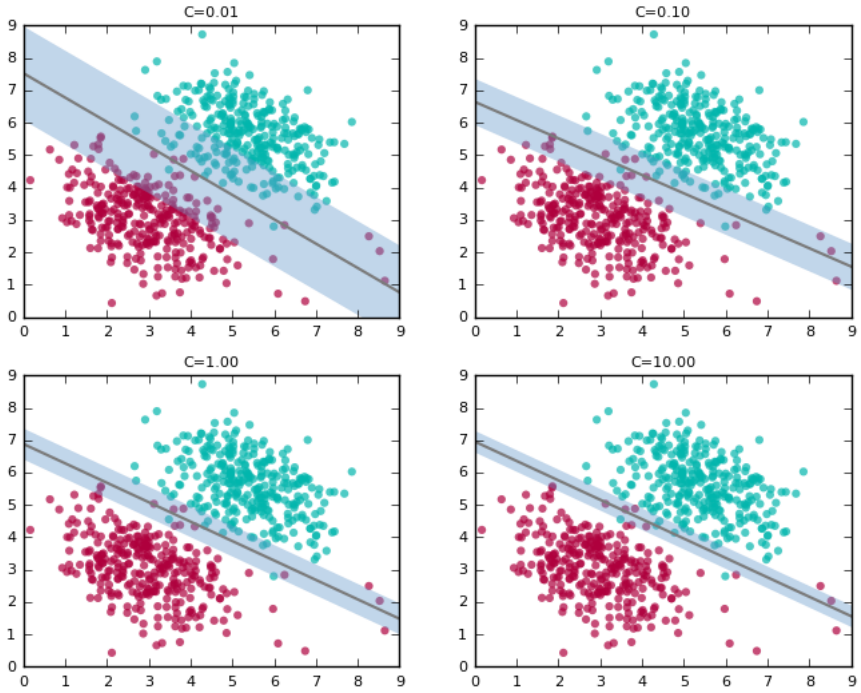

Figure 4. Variação do coeficiente C [Ghose 2017]

5. Conclusão

Com este breve estudo sobre SVM's foi possível notar a ideia principal por trás da técnica de classificação, e dos vários parâmetros e variáveis que devem ser considerados para otimizar a solução e encontrar a melhor função de classificação para cada problema.

Esta técnica pode ser utilizada em diversas aplicações, como detecção do humor de textos, classificação de imagens, podendo ser utilizado para solucionar problemas nas mais diversas áreas, como saúde, finanças, e processamento de sinais.

References

Ghose, A. (2017). Support vector machine (svm) tutorial.

Williams, C. (2008). Support vector machines. *School of Informatics, University of Edinburgh*.