Proceeding Series of the Brazilian Society of Applied and Computational Mathematics, Vol. 3, N. 1, 2015.
Trabalho apresentado no XXXV CNMAC, Natal-RN, 2014.

Um Método de Região de Confiança sem Derivadas com Modelos Construídos por Máquinas de Vetores Suporte para Regressão

Adriano Verdério
PPGMA - Programa de Pós-graduação em Matemática - UFPR
CEM - Centro de Engenharias da Mobilidade - UFSC
Campus Joinville
89218-000, Joinville, SC
E-mail: adriano.verderio@ufsc.br

Katya Scheinberg
Department of Industrial and Systems Engineering - Lehigh University
18015-1582, Bethlehem, Pennsylvania, US
E-mail: katyas@lehigh.edu

Elizabeth W. Karas, **Lucas G. Pedroso**.
Departamento de Matemática - UFPR
Centro Politécnico - Jardim das Américas
81531-980, Curitiba, PR
E-mail: ewkaras@ufpr.br, lucaspedroso@ufpr.br.

Resumo: *As Máquinas de Vetores suporte são uma classe de algoritmos de Aprendizagem de Máquinas motivados por resultados da Teoria de Aprendizagem Estatística. No início, foram usadas para a classificação de padrões e posteriormente extendidas para a regressão de funções. De um certo modo é uma generalização das técnicas usuais de regressão. Nosso objetivo é utilizá-las para construir modelos que aproximam funções as quais temos conhecimento limitado, não conseguindo por exemplo calcular derivadas. Também queremos mostrar que estes modelos são boas aproximações a fim de garantir a convergência global de um método de região de confiança sem derivadas para problemas de otimização com restrições.*

Palavras-chave: *Máquinas Vetores Suporte, Otimização sem Derivadas*

1 Introdução

$$\begin{array}{ll} \text{minimizar} & f(x) \\ \text{sujeita a} & x \in \Omega, \end{array} \tag{1}$$

em que $\Omega \subset \mathbb{R}^n$ é um conjunto convexo, fechado e não vazio e $f : \mathbb{R}^n \to \mathbb{R}$ é uma função diferenciável. Embora assumamos uma função diferenciável, o interesse reside quando as derivadas não estão disponíveis.

Conejo et al. em [2] propõem um algoritmo globalmente convergente de região de confiança para resolver o problema (1) e que gera uma sequência de minimizadores aproximados para

DOI: 10.5540/03.2015.003.01.0439

os subproblemas restritos. O algoritmo permite grande liberdade nas construções e resoluções dos subproblemas, e o caso de interesse é quando as derivadas da função objetivo não estão disponíveis embora é suposto que existam. Problemas desse tipo são conhecidos como problemas de otimização sem derivadas [4], que podem surgir quando queremos otimizar uma função que venha de uma simulação, por exemplo.

Um algoritmo de região de confiança [3] define a cada iteração um modelo da função objetivo e uma região em torno do ponto corrente na qual acreditamos que o modelo é confiável. Calculamos então um minimizador aproximado do modelo na região de confiança. Caso este ponto forneça uma redução razoável no valor da função objetivo aceitamos o iterando repetimos o processo. Caso contrário, pode ser que o modelo não represente adequadamente a função. Neste caso, o ponto é recusado e reduzimos o tamanho da região para encontrar um novo minimizador.

No algoritmo proposto em [2] qualquer modelo pode ser utilizado desde que forneça uma aproximação suficientemente precisa da função objetivo. Nosso objetivo é mostrar que os modelos construídos por Máquinas de Vetores Suporte aproximam bem funções na região de confiança, uma hipótese necessária para a convergência.

2 Máquinas de Vetores Suporte

Sejam $\mathcal{X}$ e $\mathcal{Y}$ subconjuntos de espaços vetoriais normados, normalmente $\mathbb{R}^n$ e $\mathbb{R}$, respectivamente. Suponha que é dado algum conjunto de entradas $X = \{x^1, x^2, \ldots, x^m\} \subset \mathcal{X}$ e um conjunto $Y = \{y^1, y^2, \ldots, y^m\} \subset \mathcal{Y}$ de forma que sejam independentes e identicamente distribuídos de acordo com alguma medida de probabilidade $d\mathbb{P}(x, y)$. Vamos supor que os pontos de Y são as imagens dos pontos de X para a função f que gostariamos de minimizar, ou seja, $f(x^i) = y^i$. Problemas desse tipo podem ser encontrados em simulações ou para um programa ao qual não temos acesso ao código fonte.

Chamaremos X o conjunto de amostra e Y o conjunto de rótulos. As Máquinas de Vetores Suporte para regressão são construídas para encontrar uma função $h : \mathcal{X} \longrightarrow \mathcal{Y}$ que aproxima bem os dados de amostra, ou seja, $h(x^i) \approx y^i$ para $(x^i, y^i) \in X \times Y$, chamaremos essa funçao h de preditor. Também esperamos que nosso preditor h seja uma boa aproximação para a função f.

Primeiramente, precisamos encontrar um preditor que apresenta o menor erro e para tal usaremos uma medida para avaliar sua performance. Uma maneira natural de mensurar o erro cometido é através de uma função perda, que mede o erro que um preditor comete no conjunto de amostra.

Definição 1. *Seja a tripla $(x, y, h(x)) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{Y}$ que consiste de um padrão x, um rótulo y e a predição $h(x)$. Então a aplicação $\ell : \mathcal{X} \times \mathcal{Y} \times \mathcal{Y} \to [0, \infty)$ com a propriedade $\ell(x, y, y) = 0$ para todo $x \in \mathcal{X}$ e $y \in \mathcal{Y}$ é chamada de função perda.*

Uma vez definida a função perda a ser utilizada, conseguimos determinar como os erros são penalizados em cada ponto da amostra. Precisamos agora encontrar uma maneira de combinar essas penalidades locais e conseguirmos avaliar a qualidade de um preditor.

Como os dados utilizados são independentes e identicamente distribuídos de acordo com alguma medida de probabilidade $\mathbb{P}(x, y)$, o valor esperado da função perda é

$$R[h] = \mathbb{E}[\ell(x, y, h(x))] = \int_{\mathcal{X}\times\mathcal{Y}} \ell(x, y, h(x)) \; d\mathbb{P}(x, y),$$

DOI: 10.5540/03.2015.003.01.0439

que é chamado de risco esperado, ou simplesmente risco. Minimizando o risco de um preditor, encontramos o melhor candidato a aproximar nossos dados.

No entanto, minimizar o risco esperado de um preditor é impossível, pois não conhecemos a medida de probabilidade $\mathbb{P}(x, y)$. Para resolver este problema, Vapnik em sua Teoria de Aprendizagem Estatística [9] desenvolveu o Princípio Indutivo da Minimização do Risco Empírico, no qual o Risco é determinado pelo conjunto de amostra.

O Princípio Indutivo da Minimização do Risco Empírico pode ser descrito como

1. Substituir o Risco Esperado pelo Risco Empírico

$$R_{emp}[h] = \frac{1}{m}\sum_{i=1}^{m} \ell(x_i, y_i, h(x_i)),$$

   que é a perda média encontrada no conjunto de amostra.

2. Utilizar como preditor a aplicação h que minimiza o Risco Empírico.

Apesar de em um primeiro momento parecer que o Pricípio Indutivo da Minimização do Risco Empírico resolve o problema, essa técnica sozinha não é suficiente. Para um método de reconhecimento de padrões ser eficaz, deve apresentar pelo menos duas características: boa capacidade de *generalização*, permitindo que dados semelhantes sejam igualmente classificados; boa capacidade de *discriminação*, que assegura a correta separação entre as classes. Minimizar o risco empírico pode gerar instabilidades numéricas ou ainda não alcançar uma boa generalização [1].

Segundo Schölkopf e Smola [5], uma maneira de evitar esses problemas é restringindo a classe de soluções possíveis a um conjunto compacto. Essa técnica foi introduzida por Tykhonov e Arsenin [7] para resolver problemas inversos e tem sido aplicada em problemas de aprendizagem com bastante sucesso, trabalhando com a função risco regularizada.

Em geral, adicionamos um termo de estabilização $\Omega[f]$ à função objetivo original, que em nosso caso é o risco empírico $R_{emp}[f]$. Ou seja, consideramos a seguinte classe de risco regularizado

$$R_{reg}[f] := R_{emp}[f] + \lambda\Omega[f],$$

em que $\lambda > 0$ é o parametro regularizador que específica o balanço entre a minimização do risco empírico e a simplicidade do nosso classificador que é alcançado com um $\Omega[f]$ pequeno.

Vapnik [9] concebeu a chamada função perda $\varepsilon-$insensível

$$\ell(x, y, h(x)) = |y - h(x)|_\varepsilon = \max\{0, |y - h(x)| - \varepsilon\},$$

a qual não penaliza erros menores que uma tolerância $\varepsilon > 0$ escolhida previamente. Seu algoritmo, $\varepsilon-$SVR ($\varepsilon-$*Suppot Vector Regression*), procura estimar funções por um preditor

$$h(x) = w^\top x + b \quad w, x \in \mathbb{R}^n, b \in \mathbb{R}$$

baseado nos dados X e Y.

Assim o problema é minimizar o risco empírico regularizado, ou seja, queremos minimizar

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m} |y_i - h(x_i)|_\varepsilon \tag{2}$$

DOI: 10.5540/03.2015.003.01.0439

onde $\frac{1}{2}\|w\|^2$ é o termo regularizador e C é o parâmetro de regularização.

A interpretação geométrica de utilizar o regularizador $\frac{1}{2}\|w\|^2$ é encontrar a função f mais achatada possível com suficiente qualidade na aproximação, além de capturar a ideia principal da Teoria de Aprendizagem estatística de tentar obter um risco pequeno controlando tanto o erro como a complexidade do modelo [5]. Poderíamos pensar em mínimizar apenas $\frac{1}{2}\|w\|^2$ para encontrar um preditor que aproxime a função, no entanto dependendo da quantidade de pontos o problema de otimização resultante seria inviável.

Desta maneira, para encontrarmos um preditor linear iremos resolver o problema

$$\min \frac{1}{2}\|w\| + C\frac{1}{m}\sum_{i=1}^{m} \left|x^i - h(x^i)\right|_\varepsilon ,$$

que é equivalente a resolver o problema

$$\begin{array}{ll} \min & \frac{1}{2}\|w\|^2 + C\frac{1}{m}\sum_{i=0}^{m}\left(\xi_i + \xi_i'\right) \\ \text{s.a.} & w^\top x_i + b - y_i \leq \varepsilon + \xi_i \\ & y_i - w^\top x_i + b \leq \varepsilon + \xi_i' \\ & \xi_i, \xi_i' \geq 0, \end{array}$$

que por sua vez é um problema quadrático convexo.

Seja

$$P^\top = \begin{bmatrix} (x^1)^\top \\ (x^2)^\top \\ \vdots \\ (x^m)^\top \end{bmatrix}, \quad Q = \begin{bmatrix} PP^\top & -PP^\top \\ -PP^\top & PP^\top \end{bmatrix}, \quad z = \begin{bmatrix} \alpha \\ \gamma \end{bmatrix},$$

$$v = \begin{bmatrix} -f(P) + \varepsilon e \\ f(P) - \varepsilon e \end{bmatrix} \quad \text{e} \quad A = [-e^\top, e^\top],$$

o problema dual pode ser escrito como

$$\begin{array}{ll} \min & z^\top Q z + v^\top z \\ \text{s.a.} & Az = 0 \\ & 0 \leq z \leq C. \end{array}$$

Em geral, resolvemos o problema dual por se tratar de um problema quadrático melhor tratável com apenas restrições de igualdade e caixa.

Para construir um modelo não linear, podemos levar nossos dados em um espaço de dimensão maior por meio de uma aplicação $\varphi : \mathbb{R}^n \to \mathbb{R}^q$, com $q > n$ e construir um modelo linear nesse espaço. Desse modo, queremos encontrar o modelo definido por $q(x) = w^\top \varphi(x) + b$ tal que

$$|y^i - (w^\top \varphi(x^i) + b)| \leq \varepsilon \quad \forall\ i = 1, \ldots, m.$$

Ou seja, presisamos resolver exatamente o mesmo problema que no caso linear, exceto que agora os elementos de Q são definidos por $\varphi(x^i)^\top \varphi(x^j)$ em vez de $(x^i)^\top x^j$.

Para o caso em que queremos modelos quadráticos, definimos

$$\phi(x) = (x_1^2, \sqrt{2}x_1x_2, \sqrt{2}x_1x_3, \ldots, x_2^2, \ldots, \sqrt{2}x_{n-1}x_n, x_n^2, x_1, \ldots, x_n),$$

DOI: 10.5540/03.2015.003.01.0439

de modo que

$$\phi(x^i)^\top \phi(x^j) = (x^i)^\top x^j + ((x^i)^\top x^j)^2,$$

o que facilita a construção da matriz Q do nosso problema de programação quadrática.

Assim, para construirmos modelos de uma função avaliada em alguns pontos, precisamos resolver um problema do tipo

$$\begin{array}{ll} \min & z^\top Q z + v^\top z \\ \text{s.a.} & Az = 0 \\ & 0 \leq z \leq C \end{array}$$

com Q simétrica e semidefinida positiva.

3 Construção dos Modelos

Para construção de modelos de uma função avaliada em alguns pontos precisamos de um controle na geometria dos pontos de amostra, de modo que não formem um espaço afim. Este controle é feito geralmente controlando o posicionamento dos pontos da amostra. Para as Máquinas de Vetores Suporte a geometria é controlada com a definição abaixo.

Definição 2 (Λ-posicionamento Modelos Lineares)**.** *Seja $P = \{x^0, x^1, \ldots, x^m\} \subset \mathbb{R}^n$, com $m \geq n$. Dizemos que P é Λ-posicionado em $B(x^0, \Delta)$, para uma constante $\Lambda > 0$, se existe um subconjunto de índices $J \subset \{1, 2, \ldots, m\}$ com $n+1$ pontos de modo que $\hat{P} = \{x^j - x^0\}$ com $j \in J$ seja um conjunto de vetores linearmente independentes e para todo $x \in B(x^0, \Delta)$,*

$$x - x^0 = \sum_{i=1}^{m} \lambda_i (x^i - x^0) \qquad com \qquad |\lambda_i| \leq \Lambda, i = 1, 2, \ldots, m.$$

Com essa definição conseguimos construir um modelo que aproxima bem a função original.

Teorema 1. *Sejam $P = \{x^0, x^1, \ldots, x^m\}$ um conjunto Λ-posicionado em $B(x^0, \Delta)$ em que conhecemos o valor da função f nos pontos de P e m o modelo linear construído por máquina de vetores suporte com margem $\varepsilon \leq c_1 \Delta_k^2$ e com folgas $\xi \leq c_2 \Delta_k^2$ e $\xi' \leq c_2 \Delta_k^2$ com c_1 e c_2 constantes positivas. Então existe constante $\kappa_1 > 0$ tal que*

$$|f(x) - m(x)| \leq \kappa_1 \Delta^2$$

para todo $x \in B(x^0, \Delta)$.

Caso a quantidade de pontos de amostra seja $n+1$ as folgas serão nulas, caso contrário precisamos controlar o erro entre os pontos da amostra.

Para a construção de modelos quadráticos precisamos de algumas modificações na definião de posicionamento.

Definição 3 (Λ-posicionamento Modelos Não Lineares)**.** *Seja $P = \{x^0, x^1, \ldots, x^q\} \subset \mathbb{R}^n$ com $q \geq (n+1)(n+2)/2$. Dizemos que P é Λ-posicionado com respeito a aplicação φ, para uma constante $\Lambda > 0$, se existe um subconjunto de índices $J \subset \{1, 2, \ldots, q\}$ com $(n+1)(n+2)/2$ pontos de modo que $\hat{P} = \{\varphi(x^j) - \varphi(x^0)\}$ com $j \in J$ seja um conjunto de vetores linearmente independentes e se para todo $x \in B(x^0, \Delta)$,*

$$\varphi(x) - \varphi(x^0) = \sum_{i=1}^{m} \lambda_i (\varphi(x^i) - \varphi(x^0)) \qquad com \qquad |\lambda_i| \leq \Lambda, i = 1, 2, \ldots, q.$$

DOI: 10.5540/03.2015.003.01.0439

Teorema 2. *Sejam $P = \{x^0, x^1, \ldots, x^q\}$ um conjunto Λ-posicionado em $B(x^0, \Delta)$ com respeito a aplicação φ em que conhecemos o valor da função f nos pontos de P e m o modelo quadrático construído por máquina de vetores suporte com margem $\varepsilon \leq c_1\Delta^2$ e com folgas $\xi \leq c_2\Delta^2$ e $\xi' \leq c_2\Delta^2$. Então existe constante $\kappa_2 > 0$ tal que*

$$|f(x) - m(x)| \leq \kappa_2\Delta^2$$

para todo $x \in B(x^0, \Delta)$.

Caso a quantidade de pontos de amostra seja $(n+1)(n+2)/2$ as folgas serão nulas, caso contrário precisamos controlar o erro entre os pontos da amostra.

Com os Teoremas 1 e 2 conseguimos garantir que os modelos contruídos por máquinas de vetores suporte para regressão são uma boa aproximação para a função a ser minimizada, condição necessária para a convergência global do método de região de confiança sem derivadas.

As Figuras 1 e 2 apresentamos modelos construídos por máquinas de vetores suporte para a função de Rosenbrock

$$f(x, y) = (1 - x)^2 + 100(y - x^2)^2.$$

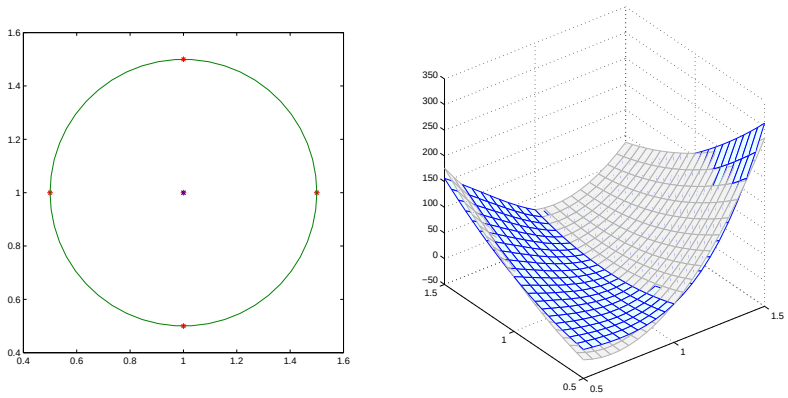

Figura 1: Modelo da Função de Rosenbrock proxímo do ponto $(1, 1)$, com raio $= 0.5$

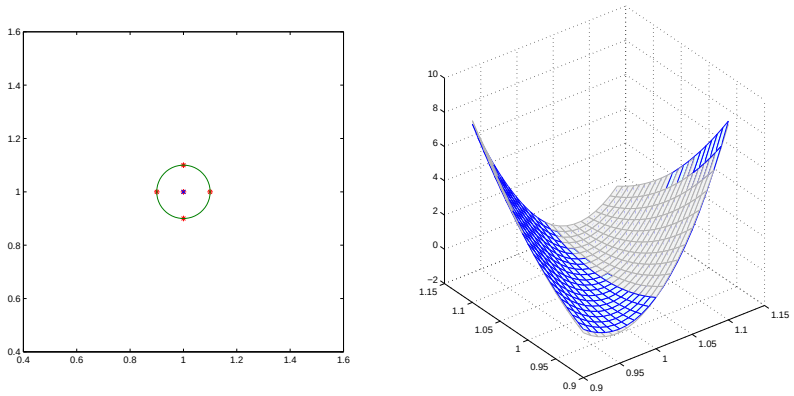

Figura 2: Modelo da Função de Rosenbrock proxímo do ponto $(1, 1)$, com raio $= 0.1$

No lado esquerdo de cada figura apresentamos a região de confiança e os pontos da amostra. Do lado direito apresentamos a função original em azul e o modelo em cinza.

DOI: 10.5540/03.2015.003.01.0439

4 Conclusão

As máquinas de vetores suporte apresentam características importantes para a aproximação de funções. Para a construção de um modelo com a técnica precisamos resolver um problema de programação quadrática.

Algumas vantagens em se utilizar as máquinas de vetores suporte em otimização sem derivadas reside na quantidade de pontos na amostra, pois a técnica permite a construção de modelos conhecendo apenas um ponto da função original. Essa é uma vantagem relevante quando temos um alto custo computacional para avaliar funções a serem minimizadas.

Os modelos construídos por máquinas de vetores suporte são uma boa aproximação para funções as quais temos conhecimento limitado, condição necessária para a convergência global do método de região de confiança sem derivadas para problemas de otimização restrita.

Referências

[1] C. J. C. Burges. A tutorial on support vecor machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.

[2] P. D. Conejo, E. W. Karas, L. G. Pedroso, A. A. Ribeiro, e M. Sachine. Global convergence of trust-region algorithms for convex minimization without derivatives. *Applied Mathematics and Computation*, 220:324–330, 2013.

[3] A. R. Conn, N. I. Gould, e Ph. L. Toint. *Trust-Region Methods*. Society for Industrial and Applied Mathematics, Philadelphia, 2000.

[4] A. R. Conn, K. Scheinberg, e L. N. Vicente. *Intoduction to Derivatife-Free Optimization*. Society for Industrial and Applied Mathematics, Philadelphia, 2009.

[5] B. Schölkopf e A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. The MIT Press, Cambridge, 2002.

[6] A. J. Smola e B. Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 14:199–222, 2004.

[7] A. N. Tikhonov e V. Y. Arsenin. *Solution of Ill-Posed Problems*. V. H. Winston & Sons, Washington, 1977.

[8] V. N. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, New York, 1998.

[9] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, 2000.

