Universidade Federal de Juiz de Fora
Departamento de Ciência da Computação
Curso de Ciência da Computação

Roberto Carlos Soares Nalon Pereira Souza

Algoritmo de Margem Incremental para problemas de regressão

Juiz de Fora
2010

Roberto Carlos Soares Nalon Pereira Souza

Algoritmo de Margem Incremental para problemas de regressão

Monografia apresentada ao Curso de Ciência da Computação da UFJF, como requisito para a obtenção parcial do grau de BACHAREL em Ciência da Computação.

Orientador: Raul Fonseca Neto

Pós-Doutor em Modelagem Computacional

Laboratório Nacional de Computação Científica

Juiz de Fora

2010

Souza, Roberto

Algoritmo de Margem Incremental para problemas de regressão /

Roberto Souza - 2010

43.p

1.Aprendizado de Máquinas 2.Redes Neurais.. I.Título.

CDU 536.21

Roberto Carlos Soares Nalon Pereira Souza

Algoritmo de Margem Incremental para problemas de regressão

Monografia apresentada ao Curso de Ciência da Computação da UFJF, como requisito para a obtenção parcial do grau de BACHAREL em Ciência da Computação.

Aprovado em 16 de dezembro de 2010

BANCA EXAMINADORA

Raul Fonseca Neto

Pós-Doutor em Modelagem Computacional

Laboratório Nacional de Computação Científica

Carlos Cristiano Hasenclever Borges

Pós-Doutor em Bioinformática

Medical University of South Carolina

Saulo Moraes Villela

Doutorando em Engenharia de Sistemas e Computação

Universidade Federal do Rio de Janeiro

À minha mãe.

Porque justificativas não seriam suficientes para explicar o quanto ela é responsável por esse trabalho.

Resumo

O presente trabalho procura apresentar um novo algoritmo para solução de problemas de regressão, chamado de Algoritmo de Margem Incremental. Esse algoritmo utiliza uma única formulação baseada em um sistema de inequações, computa soluções equivalentes às soluções SV Regressor, não utiliza pacotes de programação linear ou não linear e garante sempre uma solução. Para tanto vale-se somente de uma estratégia de adaptação para o valor da margem, no caso da classificação, e do valor do raio do tubo, no caso da regressão, e a solução de um sistema de inequações. Inicialmente são apresentados conceitos fundamentais sobre a área de estudo, como conceitos básicos de classificação e regressão, o modelo perceptron e máquinas de vetores suporte. São apresentados também o modelo do perceptron de margem fixa e o algoritmo de margem incremental. Ao final são apresentados testes comparativos e uma proposta de flexibilização da margem.

Palavras-chaves: Aprendizado de Máquina, Redes Neurais, Regressão.

Agradecimentos

Agradeço a Deus pela vida, saúde e sabedoria concedidos para que pudesse superar as adversidades que surgiram ao longo do caminho sem desanimar.

A minha família por todo suporte, carinho e atenção dispensados a mim. À minha mãe Núbia pelo amor, o abraço caloroso que acalma, pelas palavras de sabedoria e conforto no momentos difíceis e pela certeza de que ela sempre estaria torcendo por mim. Ao meu segundo pai Etevaldo pelo carinho, atenção e por todas as conversas. Ao meu pequeno irmão Pedro por ser a alegria da família em todos os momentos. À minha tia Nely e minha avó Manoelina por sempre incentivarem meus estudos na certeza de que esse era o caminho certo. À minha prima Virgínia pelo momentos divertidos.

À minha avó Hilda, que mesmo distante esteve sempre presente, nunca deixando de se preocupar comigo. Ao meu pai Edson e a Érika pela recepção e hospitalidade nos dois primeiros anos na cidade de Juiz de Fora.

Aos meus amigos de longa data e conterrâneos, Bruno, Lucas, Luís, Marcelo, Marconi e Matheus, na certeza de que mesmo distante estamos sempre unidos pelos laços de amizade. Não consigo me lembrar de bons momentos nos tempos de escola em que vocês não estejam presentes. Em especial agradeço ao Luís que foi um irmão e companheiro ao longo desses quatro anos, pelas conversas sem fim, pelas risadas descontroladas e pelo apoio nos momentos complicados. Sem dúvida a caminhada é mais fácil quando não se está sozinho.

Aos amigos do Curso de Estatística. A todos vocês pela amizade valiosa durante esse longo tempo, os bons momentos de descontração e a presença constante. Ao Bruno pela a hospitalidade em Rio Pomba e as visitas na roça do Chico Rato. Iago pelo exemplo de força de vontade e capacidade. Samuel, homem íntegro e de boa fé, dignitário das missões impossíveis. Carol, Laura e Priscila pelo carinho de sempre. Victor e Thiago pelos momentos mais engraçados.

Aos amigos do Curso de Ciência da Computação, sem dúvida a melhor turma de todos os tempos. Em especial, ao Abraão e Felipe pelos vários momentos de sufoco nas

disciplinas, mas muito mais momentos de descontração e gargalhadas. Ao Márcio pela hospitalidade, muitas vezes, decisiva.

Aos pesquisadores, estagiários e demais funcinários da Embrapa Gado de Leite, onde fui bolsista por um bom tempo. Certamente as amizades construídas e o aprendizado adquirido permanecerão para sempre. Em especial agradeço ao pesquisador e amigo Marcos Cicarini Hott, exemplo de profissionalismo, caráter e superação, pela oportunidade concedida, pela amizade e pelos trabalhos e discussões sempre enriquecedores.

Ao professor Raul Fonseca Neto pela orientação, confiança e conselhos durante este trabalho. Ao professor Saulo Moraes Villela por toda ajuda no desenvolvimento deste trabalho e ao professor Carlos Cristiano Hasenclever Borges por aceitar participar dessa banca.

Ao professor Tarcísio de Souza Lima pela oportunidade de participar do Projeto de Universalização da Informática.

Ao professor e amigo Márcio Souza pelo apoio e bons conselhos, além de seis meses de muita risada.

À professora Maria Julieta Ventura Carvalho de Araújo pelas melhores aulas ministradas.

Aos funcinários da coordenação e secretaria do ICE de maneira geral.

À todos que contribuíram para que hoje isso se tornasse realidade.

Sumário

Lista de Figuras

Lista de Tabelas

1 Introdução

O problema de regressão está associado a uma forma de mapeamento, onde o conjunto de valores desejáveis assume valores reais. De maneira geral, deseja-se encontrar uma função que aproxima determinados valores de uma variável contínua. Para isso, aplica-se um algoritmo de aprendizado sobre um conjunto de dados, buscando encontrar uma função que se ajuste bem a esse conjunto. Nesse caso, no sentido de verificar o quanto essa função se aproxima da distribuição dos dados, é essencial a utilização de uma função de perda que deve ser minimizada.

A abordagem clássica da estimativa de regressores, proposta por Gauss, que pode ser encontrada em Principe, Euliano & Lefebvre (2000), considera a minimização da função de perda quadrática que busca minimizar o quadrado das distâncias dos pontos até a posição do regressor, chamado de método dos mínimos quadrados. Laplace sugeriu uma função de perda diferente, baseada na minimização da soma absoluta dos desvios, denominada de método dos mínimos módulos, encontrada em Principe, Euliano & Lefebvre (2000).

Vapnik (1995), introduziu uma nova função de perda, denominada de ρ - insensível, bem como o conceito de tubo. Estes novos elementos permitiram a aplicação de vetores suportes ao problema de regressão, possibilitando, o desenvolvimento de uma máquina de vetores suportes específica para o problema de regressão, denominada regressão SV.

1.1 Objetivo

O objetivo do trabalho é apresentar um novo algoritmo para resolver problemas de regressão, chamado de IMA Regressor. Apesar de permitir a solução de problemas não lineares, nesse trabalho o foco é apenas a solução de problemas de regressão linear, em que a relação entre duas variáveis pode ser representada por uma reta, ou em um caso mais geral, por um hiperplano. Os resultados do IMA regressor são comparados aos de uma Rede Neural implementada no software *NeuroSolutions*, quanto aos critérios que

serão apresentados, para um base de dados de temperatura do mar e pressão atmosférica. Além disso, é proposta uma flexibilização da margem através da eliminação progressiva de alguns pontos suportes do conjunto de treinamento.

1.2 Organização do Trabalho

Esta monografia está organizada em cinco capítulos incluindo a introdução. O capítulo 2 traz uma fundamentação teórica de aprendizado de máquinas, com as principais técnicas conhecidas e um direcionamento para o foco do trabalho. O terceiro capítulo apresenta o algoritmo de margem incremental, seu funcionamento e aplicação aos problemas de regressão. No quarto capítulo são apresentados resultados do IMA regressor, comparação com os resultados obtidos pela rede neural e a proposta de flexibilização da margem. O último capítulo traz as considerações finais acerca do trabalho realizado.

2 Fundamentação Teórica

2.1 Aprendizado de Máquinas

As técnicas de Aprendizado de Máquinas empregam um princípio de inferência denominado indução, no qual obtém-se conclusões genéricas a partir de um conjunto particular de exemplos (LORENA & CARVALHO, 2003).

Classificação e regressão constituem as principais técnicas de aprendizado supervisionado, em que cada amostra x_i do conjunto de dados possui um rótulo y_i.

No aprendizado supervisionado tem-se a figura de um professor externo, o qual apresenta o conhecimento do ambiente por conjuntos de exemplos na forma: entrada e saída desejada (HAYKIN, 1999).

2.1.1 Classificação

Um dos problemas mais comuns em aprendizado de máquinas é a classificação. Seja um conjunto de dados $D = \{(X_1, y_1), (X_2, y_2), ..., (X_n, y_n)\}$, onde X_i é uma amostra associada a classe y_i, e $|D| = n$. Nesse caso y_i assume valores discretos, comumente $(+1)$ ou (-1).

O objetivo da classificação constitui-se em construir um modelo, através do aprendizado adquirido a partir do conjunto de dados, que seja capaz de predizer a classe de uma nova amostra desconhecida, com um certo nível de precisão.

Esse processo de indução de um classificador a partir de uma amostra de dados é denominado treinamento. O classificador obtido também pode ser visto como uma função f, a qual recebe um dado x e fornece uma predição y. (LORENA & CARVALHO, 2003)

2.1.2 Regressão

Diferentemente da classificação, na regressão os valores a serem preditos são contínuos. Assim, para um conjunto de dados $Z = \{(X_1, d_1), (X_2, d_2), ..., (X_n, d_n)\}$, onde $X_i \in R^D$ é

um vetor de variáveis reais ou atributos de dimensão D, e $|Z| = n$. Os valores desejáveis d_i, assumem valores reais, ou seja, $d_i \in R$.

2.1.3 Modelos Lineares

Suponha o conjunto de dados apresentado na Tabela 2.1. A partir dos pares de valores (x_i, d_i) da tabela, a distribuição dos dados pode ser apresentada de forma gráfica, em que os valores de x são representados no eixo das abscissas e os valores d no eixo das ordenadas.

x	d
1	1,72
2	1,90
3	1,57
4	1,83
5	2,13
6	1,66
7	2,05
8	2,23
9	2,89
10	3,04
11	2,72
12	3,18

Tabela 2.1: Exemplo de um conjunto com apenas uma variável de entrada

Observando a Figura 2.1 pode-se depreender que exista um relação aproximadamente linear entre os dados.

Nesse sentido, um regressor linear simples procura ajustar esse conjunto de pontos definindo a equação de uma reta:

$$d \approx w * x + b \tag{2.1}$$

onde w é o coeficiente angular da reta, normalmente chamado de *vetor de pesos*, que indica a inclinação da reta em relação ao eixo das abscissas, e b o coeficiente linear, chamado

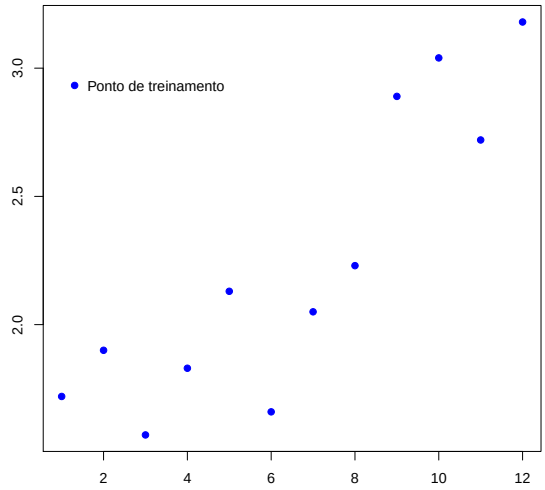


Figura 2.1: Representação dos dados no plano cartesiano

de *bias*, que representa o intercepto, ou seja, o ponto em que a reta corta o eixo das ordenadas. A Figura 2.2 mostra a reta do regressor.

De acordo com os valores de w e b, obtém-se uma reta que irá se ajustar melhor ou pior ao conjunto de pontos.

Mais especificamente, ao se levar em conta um conjunto de medidas, o modelo deverá incluir uma parcela de erro ε_i, obtendo-se:

$$d = w * x + b + \varepsilon_i = y_i + \varepsilon_i \tag{2.2}$$

Assim, o erro individual de cada mapeamento é dado pela diferença entre valor estimado e o valor desejado na forma:

$$\varepsilon_i = d_i - y_i \tag{2.3}$$

Dessa maneira, o erro é representado pela diferença de valores funcionais entre o valor desejado e o valor calculado pela reta do regressor. Para escolher a reta que melhor se ajusta aos dados, é preciso um critério para para determinar o melhor regressor. O critério mais comumente utilizado é o de minimização de uma função de Erro Médio Quadrado (MSE), calculado da seguinte forma:

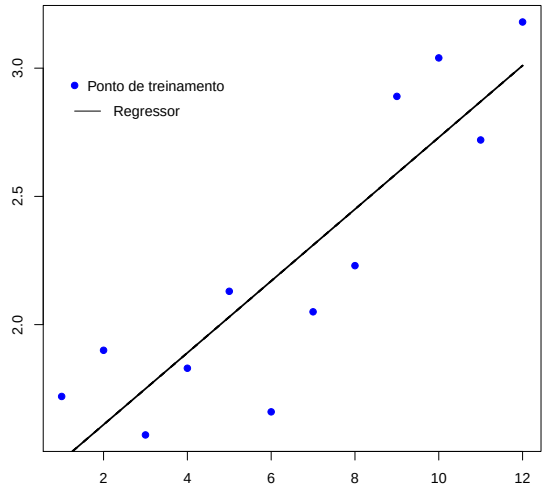


Figura 2.2: Representação da reta do regressor ajustado aos dados de entrada

$$J = \frac{1}{2N} * \sum_{1}^{N} \varepsilon_i^2 \tag{2.4}$$

onde N corresponde ao número de amostras.

A solução do problema pode ser obtida iterativamente através da utilização do método do gradiente estocástico. A função de erro J, também chamada de superfície de erro, tem a forma quadrática, visto na Figura 2.3.

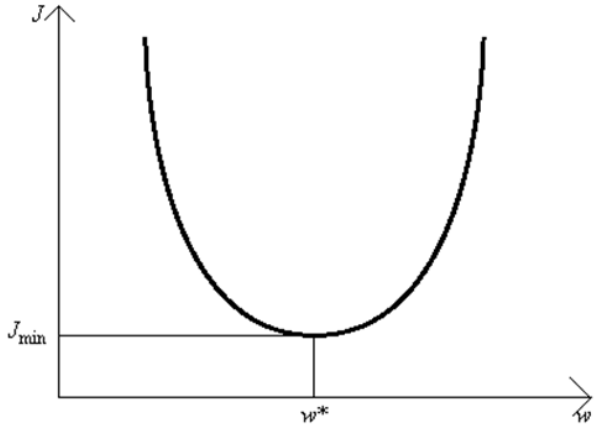


Figura 2.3: Superfície de erro para problemas com uma variável de entrada

2.1.4 Técnicas e Soluções

A solução analítica do problema, proposta por Karl Friedrich Gauss, encontrada em Principe, Euliano & Lefebvre (2000), consiste em obter as derivadas parciais da equação 2.4 em relação aos parâmetros da equação de regressão, w e b, e igualar a zero, como nas equações 2.5 e 2.6.

$$\frac{\partial J}{\partial b} = 0 \tag{2.5}$$

$$\frac{\partial J}{\partial w} = 0 \tag{2.6}$$

Essa solução determina um sistema de equações lineares, chamadas de Equações Normais, apresentadas a seguir.

$$\sum_{i=1}^{N} d_i = N * b + w \sum_{i=1}^{N} x_i \tag{2.7}$$

$$\sum_{i=1}^{N} x_i d_i = b \sum_{i=1}^{N} x_i + w \sum_{i=1}^{N} x_i^2 \tag{2.8}$$

A solução desse conjunto de equações é:

$$b = \frac{\sum\limits_i x_i^2 \sum\limits_i d_i - \sum\limits_i x_i \sum\limits_i x_i d_i}{N \sum\limits_i x_i^2 - (\sum\limits_i x_i)^2} \tag{2.9}$$

$$w = \frac{\sum\limits_i x_i d_i - \frac{\sum\limits_i x_i \sum\limits_i d_i}{N}}{\sum\limits_i x_i^2 - \frac{(\sum\limits_i x_i)^2}{N}} \tag{2.10}$$

Widrow & Hoff (1962) propõem a solução *online* do problema através do cálculo instantâneo do gradiente da função de erro. Através do vetor gradiente, que é calculado através das derivadas parciais da função, pode-se encontrar os seus pontos de máximo ou mínimo. O vetor gradiente aponta sempre para a direção de máximo crescimento da função. Neste caso, como se deseja minimizar a função de erro, basta seguir, então, a direção oposta do gradiente. Por este motivo, o processo de minimização dos erros também é denominado de método do gradiente descendente, mostrado na Figura 2.4.

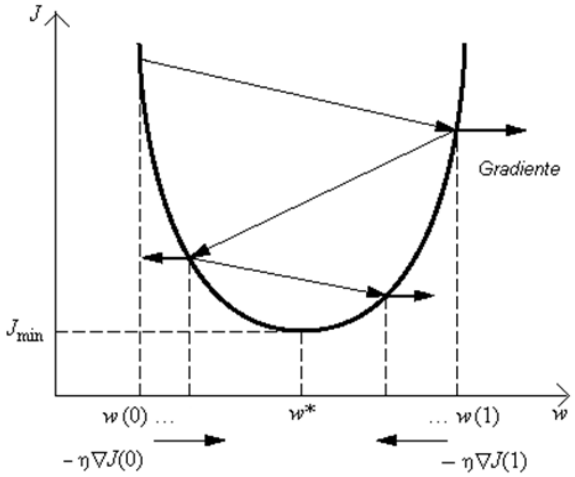


Figura 2.4: Método do gradiente descendente

Cada ponto da superfície de erro representa um valor para w, ($w(0)$, $w(1)$, e assim por diante). Seguindo na direção oposta do vetor gradiente do ponto atual, obtém-se cada vez um valor menor da função a cada iteração e um novo valor de w mais próximo do valor ideal. O valor do gradiente é computado como na Equação 2.11, e o valor de w é corrigido como mostrado na Equação 2.12.

$$\nabla J = \left[\frac{\partial J}{\partial w_0}, \frac{\partial J}{\partial w_1}, ..., \frac{\partial J}{\partial w_D}\right] \tag{2.11}$$

$$w_{t+1} = w_t - \eta * \bigtriangledown J_k \tag{2.12}$$

onde η é a taxa de aprendizagem.

2.2 Perceptron

O algoritmo desenvolvido por Rosenblatt (1958), pode ser utilizado para a determinação do vetor w em um número limitado de iterações. A quantidade de iterações está relacionada à quantidade de atualizações do vetor de pesos e, consequentemente, à quantidade de erros cometidos pelo algoritmo. Neste caso, como o vetor de pesos w, normal ao hiperplano, é determinado com base em sucessivas correções, de modo a minimizar uma função de perda, pode-se dizer que o hiperplano separador é construído de forma iterativa, caracterizando um processo de aprendizado definido como *online*.

Para uma determinada amostra do conjunto de treinamento Z, ocorrerá um erro, ou uma classificação incorreta, se:

$$y_i(< w, \phi(x_i) >) < 0 \tag{2.13}$$

Neste sentido, pode-se adotar como função de perda a quantidade de amostras classificadas incorretamente. Esta função, definida como a função de perda $0-1$, é descrita como:

$$J(w) = \sum_i 1|\{\varphi(f(x_i)) \neq y_i\} = \sum_i Max\{0, \varphi(-y_i(< w, \phi(x_i) >))\} \tag{2.14}$$

$(x_i, y_i) \in Z$

Entretanto, sendo esta função constante por partes e, portanto, não diferenciável, torna-se mais apropriado a utilização de uma nova função de perda, linear por partes, dada pela soma negativa de todos valores funcionais, também chamados de valores de margens, das amostras classificadas incorretamente. Ou seja:

$$J(w) = \sum_i Max\{0, (-y_i(< w, \phi(x_i) >)\} \tag{2.15}$$

$(x_i, y_i) \in Z$, tornando possível a utilização do método do gradiente.

Portanto, caso o problema seja linearmente separável, para se determinar uma solução que minimize a função de perda em relação ao vetor w, é necessário avaliar o vetor gradiente, considerando somente a ocorrência das amostras classificadas incorretamente, relacionadas ao conjunto Z. Este processo, aplicado individualmente a cada amostra, resulta na seguinte regra de correção:

$$w_{t+1} = w_t + \eta * \phi(x_i) * y_i \tag{2.16}$$

$(x_i, y_i) \in Z$, sendo η a taxa de aprendizado.

Esta versão do método do gradiente que produz, ou computa, um conjunto de valores instantâneos considerando uma única amostra de cada vez, é também chamada de *online* ou estocástica. Este processo, que iterativamente estima o verdadeiro valor do gradiente, é similar ao método dos mínimos quadrados ou *LMS*, Widrow & Hoff (1960), o qual minimiza uma função de perda quadrada de modelos lineares.

2.3 Perceptron de Margem Fixa

Duda, Hart & Stork (2000) propõem uma versão alternativa para o algoritmo perceptron incluindo a utilização de uma regra de incremento variável para uma função de perda quadrada e um valor fixo de margem γ, no sentido de adaptar a sua solução ao método de relaxação. Considerando a introdução do parâmetro γ, a solução do problema consiste na determinação de uma solução viável para o sistema de inequações lineares na forma:

$$y_i(w, \phi(x_i)) \geq \gamma \tag{2.17}$$

Entretanto, caso se utilize uma regra de incremento fixo e não seja possível limitar o valor da norma quadrática do vetor w com a adição da restrição adicional de normalização $\parallel w \parallel_2 = 1$, o sistema de inequações, se linearmente separável, apresentará sempre uma solução viável considerando o crescimento da norma e, consequentemente, do valor do produto interno, para qualquer valor de margem γ. Para resolver este problema é necessário estabelecer alguma forma de regularização, no sentido de controlar ou de limitar o valor da norma do vetor w.

Leite & Fonseca Neto(2007) apresentam uma nova formulação para o modelo perceptron no sentido de garantir que o conjunto de exemplos guarde uma distância mínima em relação ao hiperplano separador sem limitar diretamente o valor da norma do vetor w.

Para tanto, é considerada a restrição de que cada amostra deva possuir um valor de margem geométrica correspondente, superior ou igual ao valor estabelecido como margem fixa, sendo o valor da margem geométrica definido como o valor da margem funcional da respectiva amostra, dividido pelo valor da norma euclidiana do vetor w. Isto equivale à realização do produto interno do vetor $\phi(x_i)$ pelo vetor unitário de direção w, representado por $\frac{w}{\|w\|_2}$.

Neste sentido, deve-se resolver o seguinte sistema de inequações não lineares para determinado valor de margem fixa, representado pelo parâmetro γ_f:

$$y_i(< w, \phi(x_i) >) \geq \gamma_f * \parallel w \parallel_2 \tag{2.18}$$

Em função desta modificação, torna-se necessário reescrever a função de perda

do modelo, de forma a possibilitar a obtenção de uma nova regra de correção. A nova função será equivalente à soma dos valores das respectivas margens geométricas dos exemplos, que forem menor que o valor da margem fixa, descontado o valor da margem, ou seja:

$$J(w) = \sum\nolimits_i Max\{0, \gamma_f - \frac{y_i(< w, \phi(x_i) >)}{\parallel w \parallel_2}\} \tag{2.19}$$

$(x_i, y_i) \in Z$

$$J(w) = -\sum\nolimits_i y_i(< w, \phi(x_i) >) - m * \gamma_f * \parallel w \parallel_2 \tag{2.20}$$

$(x_i, y_i) \in M, \mid M \mid = m$

Portanto, ao contrário do algoritmo básico do perceptron, considera-se também como erro, aqueles exemplos que, embora classificados corretamente, não estejam a uma distância mínima, no sentido geométrico, do hiperplano separador. Kivinen, Smola & Willianson (2004) definem este tipo de correção como a ocorrência de erros de margem.

A solução do sistema de inequações pode ser considerada como aquela que minimiza a função de erro J. Neste sentido, tomando-se o gradiente da função em relação ao vetor w, caso ocorra um erro $y_i(< w, \phi(x_i) >) < \gamma_f * \parallel w \parallel_2$, tem-se a seguinte regra de correção aplicada a uma determinada amostra $(x_i, y_i) \in M$:

$$w_{t+1} = w_t - \eta(\gamma_f * \frac{w}{\parallel w \parallel} - \phi(x_i) * y_i) \tag{2.21}$$

onde η se refere à taxa de aprendizagem.

2.4 Máquina de Vetores Suporte

Fundamentada na Teoria do Aprendizado Estatístico, a Máquina de Vetores Suporte, comumente chamada de SVM (*Support Vectors Machine*), foi proposta por Vapnik (1992), com o objetivo de resolver problemas de classificação.

A idéia principal consiste na construção de um hiperplano ótimo, dentre os vários possíveis, de separação entre elementos de classes distintas, através da maximização da margem.

2.4.1 Margem e Vetores Suporte

Sendo $f(x) = (w \cdot x) + b$ um hiperplano, a margem é calculada pela distância entre hiperplano e as amostras do conjunto de treinamento que estão mais próximas a ele, sendo esses vetores chamados de *vetores suportes*. A margem determina quão bem duas classes podem ser separadas Smola et al. (1999). A máxima margem é obtida através do hiperplano ótimo.

2.4.2 Classificação de Dados Linearmente Separáveis

Supondo que o conjunto é linearmente separável, o hiperplano ótimo, que separa as classes com máxima margem, pode ser obtido:

$$< w \cdot x > +b = 0 \tag{2.22}$$

em que w e b, são respectivamente o vetor peso e o bias. A Figura 2.5 apresenta o exemplo um de conjunto de dados linearmente separável.

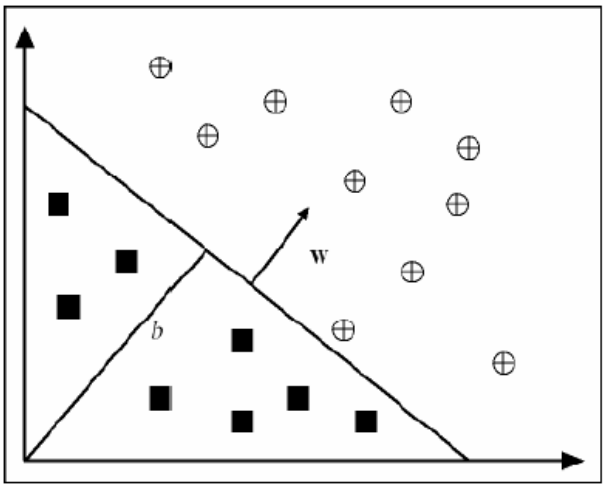


Figura 2.5: Conjunto Linearmente Separável

Assim, o hiperplano é a superfície de decisão entre os elementos das duas classes.

$$< w \cdot x_i > +b \geq +1, \text{ para } y_i = +1$$
$$< w \cdot x_i > +b \leq -1, \text{ para } y_i = -1$$

Combinando as duas inequações obtém-se:

$$y_i * (< w \cdot x_i > +b) \geq 1 \tag{2.23}$$

para $i = \{1, 2, ..., n\}$

Vapnik (1992) propõe a solução do problema definindo um valor mínimo de margem funcional, $\gamma_f = 1$, para os pontos que se situarem nas margens do hiperplano separador. Considerando a margem geométrica, $\gamma_g = \frac{\gamma_f}{||w||_2}$ tem-se:

$$\begin{gathered} \text{Max } \gamma_g \\ \text{Sujeito à:} \\ y_i * \frac{(<w, x_i> + b)}{||w||_2} \geq \gamma_g \end{gathered}$$

para $i = 1, ..., m$

De forma equivalente, tem-se:

$$\begin{gathered} \text{Min } \frac{1}{2} * ||w||^2 \\ \text{Sujeito à:} \\ y_i * (< w, x_i > +b) \geq 1 \end{gathered}$$

para $i = 1, ..., m$

2.4.3 Classificação de Dados Não Linearmente Separáveis

Em problemas reais, a separabilidade linear dificilmente é encontrada. Um conjunto de dados é dito linearmente inseparável, caso não seja possível separar os dados com um hiperplano. A Figura 2.6 mostra um conjunto linearmente inseparável.

Nesse caso, a SVM pode ser estendida para classificação de dados linearmente inseparáveis. Isso é feito basicamente através de duas etapas. A primeira constitui-se em fazer um mapeamento dos dados no espaço de entrada para uma mais alta dimensão, chamado espaço de características. A segunda etapa então é encontrar um hiperplano separador nessa nova dimensão.

Seja o conjunto de entrada $S = \{(X_1, y_1); (X_2, y_2); ...; (X_n, y_n)\}$, em que y_i é o rótulo de cada classe, com $i = 1, 2, ..., n$. O espaço de características é um espaço de

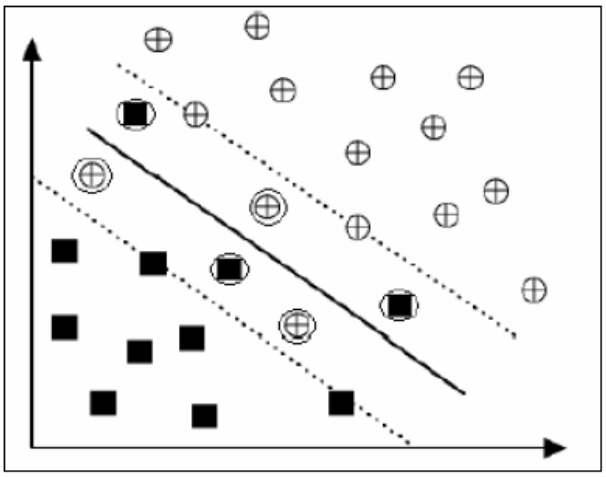


Figura 2.6: Conjunto Não Linearmente Separável

mais alta dimensionalidade no qual serão mapeados o conjunto de entrada, por meio de uma função ϕ para obter um conjunto de dados linearmente separável, representado por $S' = \{\phi((X_1), y_1), \phi((X_2), y_2), ..., \phi((X_n), y_n)\}$.

2.4.4 Funções Kernel

Teorema 2.4.1. *Teorema de Mercer: Uma função é dita ser uma função Kernel, se a matriz K é positiva definida, onde K é obtida por*

$$K = K_{ij} = \kappa(x_i, x_j) \tag{2.24}$$

Uma função *Kernel* recebe dois dados de entrada e calcula o produto interno desses dados no espaço de caracteríscas.

$$\kappa(x_i, x_j) =< \phi(x_i) \cdot \phi(x_j) > \tag{2.25}$$

2.4.5 Kernel Trick

É importante ressaltar que o algoritmo de treinamento de um classificador kernel como uma máquina de vetores suportes depende, somente, do produto interno dos vetores no espaço de entrada, seguido da avaliação da função kernel κ, como forma de determinar uma superfície de decisão linear, ou hiperplano separador, no espaço de características. Este processo é denominado *Kernel Trick*.

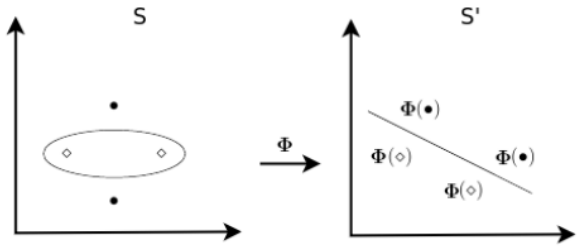


Figura 2.7: Mapeamento do conjunto no espaço de características

Algumas das funções *Kernel* mais conhecidas estão descritas na Tabela 2.2.

Kernel	Função $\kappa(x_i, x_j)$
Polinomial	$(< x_i, x_j > +1)^p$
Gaussiano	$exp(-\frac{\|\|x_i - x_j\|\|^2}{2\sigma^2})$
Sigmóide	$tanh(\beta_0 * < x_i, x_j >) + \beta_1$

Tabela 2.2: Funções Kernel mais conhecidas

2.5 Vetores Suporte e Regressão

Como apresentado em Smola & Scholkopf (1998), a aplicação de vetores suporte ao problema de regressão é feita através da determinação de tubo de raio ρ, fixado a priori, que deverá conter todos os pontos do conjunto de treinamento, sendo cada ponto formado pelo vetor x_i acrescido da componente relativa ao valor do mapeamento y_i.

A compensação entre a complexidade do modelo e os erros residuais está relacionada a existência de um conjunto de variáveis ε_i que flexibilizarão a pertinência dos pontos à região delimitada pelo tubo.

A equação do regressor ideal, definida na forma: $f(x) = w * x + b$, é computada através da minimização da seguinte função de erro:

$$\frac{1}{2} * ||w||^2 + C * \sum_i |y_i - f(x_i)|\rho \tag{2.26}$$

em que $|y_i - f(x_i)|\rho$ é definida como a função de perda de insensibilidade ρ, apresentada por Vapnik (1995) e equivalente a:

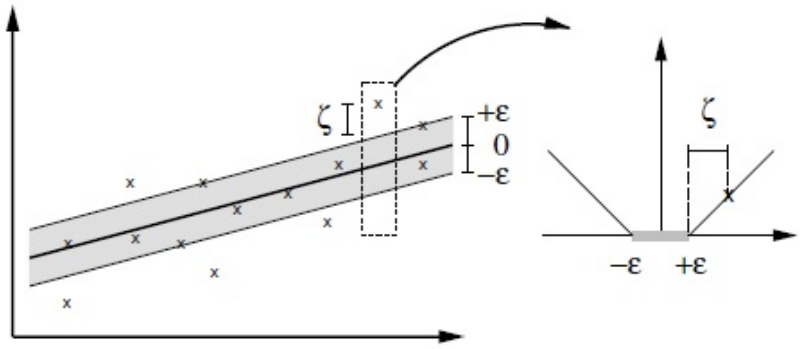


Figura 2.8: Tubo e Função de Perda

$$Max\{0, |y - f(x)| - \rho\} \tag{2.27}$$

Assim, a forma primal para o regressor apresenta-se da seguinte maneira:

$$\text{Min } \tfrac{1}{2} * ||w||^2 + C * \sum_i (\varepsilon_i + \varepsilon_i^{'})$$
$$\text{Sujeito à:}$$
$$((w * x_i) + b) - y_i \leq \rho + \varepsilon_i^{'}, \text{ para } (w * x_i) + b > y_i$$
$$y_i - ((w * x_i) + b) \leq \rho + \varepsilon_i, \text{ para } (w * x_i) + b < y_i$$

para $i = 1, ..., m$ e $\varepsilon_i, \varepsilon_i^{'} \geq 0$.

Outras métricas utilizadas e conhecidas são os critérios *MSE* proposto por Gauss, e *MML* proposto por Laplace que são apresentados nas Equações 2.28 e 2.29. O critério buscado pelo IMA Regressor é apresentado na Equação 2.30.

$$Min_w\{\frac{1}{2N} * \sum_1^N \varepsilon_i^2\} \tag{2.28}$$

$$Min_w\{\frac{1}{2N} * \sum_1^N |\varepsilon_i|\} \tag{2.29}$$

$$Min_w Max_i\{\frac{|\varepsilon_i|}{||w||_q}\} \tag{2.30}$$

2.6 Regressão Como Classificação Binária

(Bi & Bennett, 2003) propõem uma interpretação geométrica do problema de regressão, transformando o problema do regressor SV em um problema de classificação binária, para

determinado valor do raio ρ. Esta transformação é baseada na criação de uma dimensão adicional para os pontos x_i relacionada aos valores do mapeamento y_i. Em seguida, promove-se um deslocamento dos pontos, adicionando ao valores desejáveis, respectivamente, as quantidades $+\rho$ e $-\rho$. Tal procedimento produz um conjunto duplicado de pontos em um espaço de entrada de dimensão $d+1$, sendo d a dimensão do espaço de entrada.

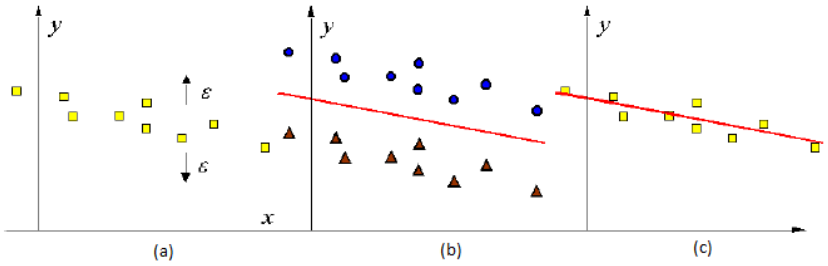


Figura 2.9: Regressão como classificação binária

(a) Dados originais (b) Dados deslocados e hiperplano separador (c) Regressão

A cada ponto é associado um valor de rótulo $+1$ ou -1 conforme o sentido do deslocamento, transformando o problema de regressão em um problema de classificação binária, associado ao valor do raio ρ. Assim, para cada par (x_i, y_i) do conjunto de treinamento do regressor, são criados dois pares $((x_i, -y_i + \rho), +1)$ e $((x_i, -y_i - \rho), -1)$ para o conjunto de treinamento do classificador. Bi & Bennett (2003), afirmam que somente haverá a existência do regressor com largura fixa ρ, chamado de ρ-tubo se o problema análogo de classificação tiver separabilidade linear.

Além disso, afirmam que qualquer solução para o problema fornecerá um tubo de raio ρ' de valor menor ou igual a ρ, e que o menor valor obtido está relacionado a solução de máxima margem obtida na classificação. Pode-se observar, entretanto, que este valor será maior ou igual ao raio ótimo, ou mínimo, ρ^* que poderia ser alcançado. Assim: $\rho^* \leq \rho' \leq \rho$.

A obtenção do raio de valor mínimo, ρ^*, exigiria a solução do seguinte problema de regressão para um parâmetro de raio variável:

$$Min_{w,b,\rho}\rho$$

Sujeito à:

$$\varphi(< w, x_i > +b - y_i) * (< w, x_i > +b - y_i) \leq \rho$$

Neste caso, deve-se empregar uma técnica que possibilite controlar a capacidade do classificador representada pela norma do vetor w. Minimizando diretamente a norma do vetor, retorna-se à formulação proposta originalmente por Vapnik (1995), que introduz um conjunto de variáveis de folga de modo a garantir a viabilidade do sistema para um dado valor fixo de raio.

2.6.1 Sistema de Inequações

Com a introdução da norma do vetor w no sistema de restrições do problema, tem-se, para um valor fixo de raio ρ, o seguinte sistema de inequações aplicado a cada par (x_i, y_i) do conjunto de treinamento:

$$\varphi(f(x_i) - y_i) * \frac{(f(x_i)-y_i)}{||w||_2} \leq \rho, \text{ ou}$$
$$\varphi(f(x_i) - y_i) * (f(x_i) - y_i) \leq \rho * ||w||_2$$

Este sistema representa de forma equivalente um problema de classificação binária com conjunto de treinamento formado pelos pares $((x_i, -y_i + \rho), +1)$ e $((x_i, -y_i - \rho), -1)$ obtidos dos respectivos pares (x_i, y_i) do conjunto de treinamento original. De fato, considerando o primeiro grupo de pares situados abaixo e acima da reta do regressor com bias incorporado, $f(x_i) =< w, x_i >$, tem-se as inequações de viabilidade representadas na forma:

$$< w \cdot x_i > -y_i \leq \rho \tag{2.31}$$

e

$$y_i - < w \cdot x_i > \leq \rho \tag{2.32}$$

equivalentes, portanto, à equação de viabilidade do problema de classificação binária para os pares $((x_i, -y_i + \rho), +1)$ e $((x_i, -y_i - \rho), -1)$, ou seja:

$$-1 * (< w \cdot x_i > +(-y_i - \rho)) \geq 0 \text{ ou } < w \cdot x_i > -y_i \leq \rho \text{ e}$$
$$+1 * (< w \cdot x_i > +(-y_i + \rho)) \geq 0 \text{ ou } y_i - < w \cdot x_i > \leq \rho$$

considerando 1 o valor da componente adicional do vetor w.

3 Algoritmo de Margem Incremental

A técnica de solução desenvolvida por Leite & Fonseca Neto (2008), para se obter uma aproximação da máxima margem consiste de uma estratégia de aprendizado incremental, através da qual, são obtidas sucessivas soluções do problema do perceptron de margem geométrica fixa, para valores crescentes de margem. Este parâmetro, o qual foi denominado de margem fixa, representado por γ_f, inicia com o valor $\gamma_f = 0$, equivalente a solução original do algoritmo perceptron, e tem seus valores incrementados de forma consistente, até aproximar-se do valor da margem máxima.

Assim, para um conjunto de valores $\gamma_f \in \{0, ..., \gamma^*\}$, sendo: $\gamma_f^{t+1} > \gamma_f^t$, para $t = \{1, ..., t-1\}$, $\gamma_f^1 = 0$, $\gamma_f^t \approx \gamma^*$ soluciona-se, sucessivamente, pelo método de relaxação o problema de inequações não lineares:

$$y_k * f(x_k) \geq \gamma_f * ||w||_2 \tag{3.1}$$

$k = 1, ..., m$, sendo cada solução equivalente a solução do problema do perceptron de margem geométrica fixa (FMP).

Para a atualização a cada iteração, do valor da margem fixa, adotam-se as seguintes regras, baseadas na estratégia de balanceamento, que garantem o estabelecimento de um número finito de correções e a convergência para a solução ótima, a medida em que o valor da margem se aproxima do valor da margem máxima.

Primeira regra: caso a solução do problema FMP forneça as margens, negativa e positiva, diferentes, pode-se dizer que a solução obtida não caracteriza uma solução de máxima margem. Portanto, corrigimos o valor da margem fixa na forma:

$$\gamma_f^{t+1} = \frac{(\gamma^+ + \gamma^-)}{2} \tag{3.2}$$

onde γ^+ e γ^- são os valores relacionados, respectivamente, as menores distâncias projetadas dos pontos do conjunto X^+ e X^- ao hiperplano separador da t^{esima} iteração.

Certamente, têm-se as relações: $\gamma^+ > \gamma_f^t$ e $\gamma^- \geq \gamma_f^t$ ou, $\gamma^+ \geq \gamma_f^t$ e $\gamma^- > \gamma_f^t$, garantindo $\gamma_f^{t+1} > \gamma_f^t$, e portanto, um incremento no valor da nova margem fixa.

Pode-se observar que, neste caso, haverá sempre a garantia da solução do novo problema, já que a nova margem fixa estabelecida é sempre inferior a margem ótima, ou seja: $\gamma_f^{t+1} = \frac{(\gamma^+ + \gamma^-)}{2} < \gamma^*$. Tal condição, deriva do fato de que se as margens negativa e positiva são desiguais então a margem total não é máxima, implicando em: $(\gamma^+ + \gamma^-) < 2 * \gamma^*$.

Infelizmente, a condição de desigualdade das margens é uma condição de necessidade e não de suficiência na caracterização de uma margem máxima ou ótima. Não devendo funcionar, portanto, como critério de parada do algoritmo. A solução do problema FMP pode, caso seja estabelecido este critério de interrupção, parar com um hiperplano cujas margens positiva e negativa estão equilibradas, mas não se trata de um ponto de ótimo global e sim de ótimo local.

Segunda regra: caso a solução do problema FMP forneça as margens, negativa e positiva, iguais, pode ser que a solução obtida seja uma solução de ótimo local. Portanto, torna-se necessário garantir um acréscimo no valor da nova margem fixa, na forma:

$$\gamma_f^{t+1} = \gamma_f^t + Max\{\Delta, \frac{(\gamma^+ + \gamma^-)}{2} - \gamma_f^t\} \tag{3.3}$$

sendo Δ uma constante de incremento positiva.

Entretanto, para esta nova forma de atualização, em alguns casos, não se tem mais a garantia de solução do novo problema, já que o novo valor da margem fixa poderá ser igual ou maior que o valor da margem ótima, ou seja: $\gamma_f^{t+1} \geq \gamma^*$.

Para a solução deste contratempo é suficiente a imposição de um número máximo de iterações no número de épocas do algoritmo de treinamento, a partir do qual, caso não haja uma nova solução do problema FMP, adota-se como margem obtida o valor anterior da margem fixa, relacionado à última solução.

3.1 IMA Aplicado à Regressão

Para resolver o sistema de inequações, Fonseca Neto e Borges (2007) propõem, a exemplo do algoritmo Perceptron de Margem Fixa, a minimização do funcional de risco regularizado em sua forma normalizada:

$$Max\{0, \frac{\mid y_i - f(x_i) \mid}{\parallel w \parallel_2} - \rho\} \tag{3.4}$$

associado à função de perda ρ-insensível.

Nota-se, ao dividir o valor da diferença pelo norma do vetor w controla-se implicitamente a norma do vetor e, portanto, a capacidade do regressor. Observe que, ao minimizar o funcional de risco na forma normalizada, minimiza-se simultaneamente a norma do vetor w e o risco empírico associado à função de perda ρ-insensível. Este funcional pode ser reescrito em termos da função sinal e de uma norma arbitrária p na forma:

$$Max\{0, \varphi(y_i - < w, x_i >) * \frac{(y_i - < w, x_i >)}{||w||_p} - \rho\} \tag{3.5}$$

Derivando a seguinte função de perda:

$$J(w) = \sum\nolimits_i Max\{0, \varphi(y_i - < w, x_i >) * \frac{(y_i - < w, x_i >)}{||w||_p} - \rho\} \tag{3.6}$$

$(x_i, y_i) \in Z$

Novamente, a solução do sistema de inequações pode ser considerada como aquela que minimiza a função de erro J em relação aos seus parâmetros primais, representados pelo vetor w que incorpora o valor do bias.

Neste sentido, para obter a solução, é suficiente o emprego do método do gradiente estocástico que fornecerá, caso ocorra o seguinte erro, $\varphi(y_i - < w, x_i >) * (y_i - < w, x_i >) > \rho * ||w||_p$, tem-se a regra de correção, para cada ponto do conjunto de treinamento Z associada ao gradiente da função em relação ao vetor normal w com norma quadrática:

$$w_{t+1} = w_t + \eta * (\rho * \frac{w}{||w||_2} + \varphi(y_i - < w, x_i >) * x_i) \tag{3.7}$$

onde η se refere à taxa de aprendizagem.

3.1.1 Estratégia Adaptativa

Observando a possibilidade da obtenção de soluções factíveis, em um número finito de correções, na solução do sistema de inequações na forma:

$$\varphi(y_i - < w, x_i >) * (y_i - < w, x_i >) \leq \rho * ||w||_2 \tag{3.8}$$

segue a mesma proposta elaborada para o algoritmo IMA Classificador em Leite e Fonseca Neto (2006). É proposta a solução aproximada do problema de regressor-SV para um raio de valor mínimo ρ^*, considerando a minimização explícita e direto do valor do raio. Neste sentido, deve-se resolver o seguinte problema de otimização:

$$Min_w \rho$$
$$\text{Sujeito à:}$$
$$\varphi(y_i - < w, x_i >) * (y_i - < w, x_i >) \leq \rho * ||w||_2$$

Neste caso, o algoritmo computa diretamente o valor do menor raio, o qual se aproxima suficientemente do raio ótimo no sentido de garantir a construção de um regressor-SV. De modo similar ao algoritmo IMA para classificação, existirá uma limitação no crescimento do valor da norma do vetor w, impedindo que o mesmo escape para valores muito altos. Desta forma, o controle da capacidade do regressor será feito de forma implícita na solução do sistema de inequações. É interessante observar que a solução encontrada é exatamente a solução $MinMax$ que minimizará a máxima distância entre os pontos do regressor.

A técnica de solução desenvolvida para se obter uma aproximação do raio mínimo consiste de uma estratégia de aprendizado incremental, através da qual, são obtidas sucessivas soluções para o sistema de inequações, para valores decrescentes do raio. Este parâmetro inicia com um valor admissível, superior a maior distância existente entre os pontos do regressor para um valor de w inicial, e tem seus valores reduzidos de forma consistente, até aproximar-se do valor do raio mínimo, ou seja, para um conjunto de valores $\rho \in \{\rho^0, ..., \rho^*\}$, sendo: $\rho^{t+1} < \rho^t$ para $t = 1, ..., T-1$, com $\rho^1 = \rho^0$ e $\rho^t = \rho^*$soluciona-se, sucessivamente, o sistema de inequações:

$$\varphi(y_i - < w, x_i >) * (y_i - < w, x_i >) \leq \rho * ||w||_2 \tag{3.9}$$

para todo $(x_i, y_i) \in Z$.

Para a atualização a cada iteração do valor do raio, adota-se a seguinte regra que garante o estabelecimento de um número finito de correções e a convergência para a solução ótima, a medida em que o valor do raio se aproxima do valor do raio mínimo:

$$\rho^{t+1} = \rho^t - \frac{(\rho^+ + \rho^-)}{2} \tag{3.10}$$

Sendo ρ^+ e ρ^- as menores distâncias relacionadas aos pontos mais afastados ou que mais se aproximam das bordas do tubo na t^{esima} iteração do algoritmo. Assim:

$$\rho^+ = Min\{\rho^t - \frac{(y_i - w.x_i)}{||w||_2}\}, \text{ para todo } (x_i, y_i), \text{ tal que } y_i \geq w.x_i$$
$$\rho^- = Min\{\rho^t - \frac{(w.x_i - y_i)}{||w||_2}\}, \text{ para todo } (x_i, y_i), \text{ tal que } y_i \leq w.x_i$$

Nota-se haverá sempre a garantia da solução do novo problema, já que pelo critério de balanceamento, o novo raio estabelecido tem valor sempre superior ou igual ao raio mínimo.

Entretanto, em relação a regra de atualização do valor do raio, poderão ocorrer duas situações relacionadas aos valores das distâncias ρ^+ e ρ^-:

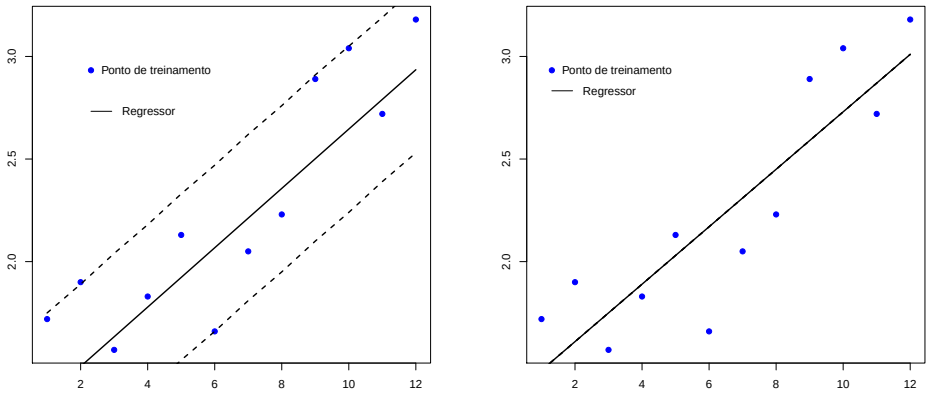
Primeiramente, pode-se ter a reta do regressor situada totalmente acima ou totalmente abaixo dos pontos. Neste caso, uma das distâncias terá o seu valor negativo. Para proceder à atualização do valor do raio considera-se a distância negativa igual a zero e toma-se o valor da correção como a metade da distância positiva.

Em segundo lugar, pode-se ter uma situação de balanceamento em que as duas distâncias terão seus valores iguais a zero. Sabe-se que esta é uma condição de necessidade mas não de suficiência para a obtenção da solução ótima correspondente ao raio mínimo. Neste caso, se o espaço de versões não é um espaço vazio haverá uma nova solução factível para um valor de raio inferior. Para tanto, realiza-se uma pequena redução no valor do raio e resolve-se novamente o sistema de inequações.

É importante observar que a solução final obtida em uma iteração serve de solução inicial ou ponto de partida para a iteração seguinte no sentido de facilitar a obtenção das soluções seguintes.

4 Resultados

Inicialmente o algoritmo foi validado com a base de dados apresentada na Tabela 2.1. É interessante notar que o IMA Regressor constrói uma solução semelhante à apresentada na Figura 2.2, obtida por uma Rede Neural treinada com o algoritmo *Backpropagation*, cujos parâmetros podem ser encontrados em (Principe, Euliano & Lefebvre, 2000). Os gráficos na Figura 4.1 mostram a comparação entre as duas soluções.



IMA Regressor Rede Neural

Figura 4.1: Comparação entre a solução do IMA Regressor e uma Rede Neural treinada com Backpropagation

Em seguida, foram então realizados testes comparativos entre o IMA Regressor e uma Rede Neural implementada no software *NeuroSolutions*, cujos parâmetros podem ser obtidos em (Principe, Euliano & Lefebvre, 2000).

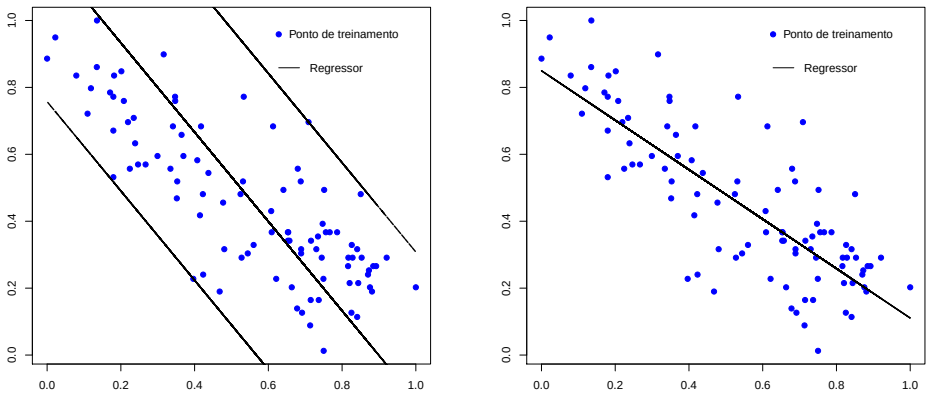
A base de dados utilizada pertence ao *National Oceanic and Atmospheric Administration (NOAA)*, e foi obtida no próprio *NeuroSolutions*. Trata-se de valores reais para temperatura do mar e pressão atmosférica, num total de 144 amostras. A base de dados foi dividida em três conjuntos com a mesma quantidade de amostras, selecionadas aleatoriamente. Foi então realizado um *3-fold-cross validation*, em que dois conjuntos são usados como conjunto de treinamento, totalizando 96 amostras e o terceiro conjunto usado como conjunto de teste, com 48 amostras restantes. Em seguida, o conjunto usado

no teste é inserido no conjunto de treinamento, e um dos dois conjuntos usados anteriormente no treinamento é tomado como conjunto de teste, e esse procedimento é repetido mais uma vez, com o conjunto que ainda não foi usado para teste. Os resultados são apresentados na Tabela 4 em que Conjunto 1, Conjunto 2 e Conjunto 3, representam as fases da validação cruzada como foi apresentado.

	IMA Regressor			Rede Neural		
	MSE	MML	IMA	MSE	MML	IMA
Conjunto 1	0,01853	0,07627	0,26528	0,00592	0,04171	0,59857
Conjunto 2	0,01585	0,07576	0,29124	0,00845	0,04698	0,59210
Conjunto 3	0,01205	0,06327	0,31351	0,01115	0,05942	0,61106

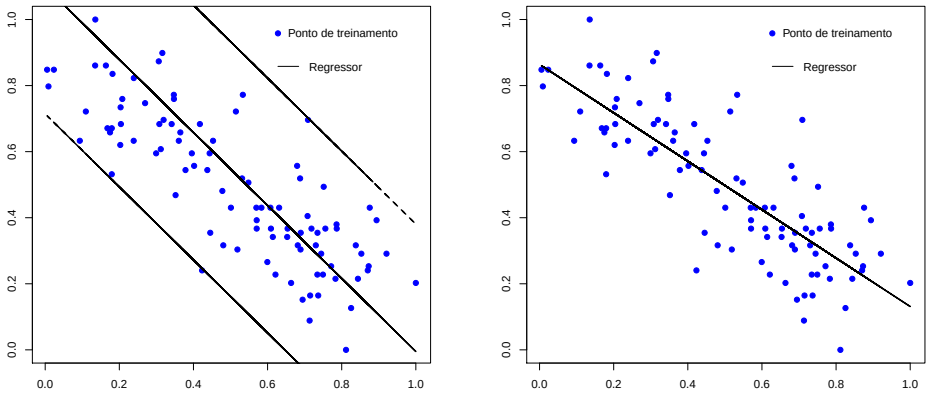
Tabela 4.1: Comparação do IMA Regressor com a Rede Neural quanto aos critérios MSE e MML

Observa-se que o IMA Regressor tem um desempenho muito semelhante ao da Rede Neural quanto aos critérios MSE e MML. No entanto, quanto ao critério do IMA, apresentado anteriormente, nota-se que ele obtém resultados melhores. Os gráficos nas Figuras 4.2, 4.3 e 4.4 apresentam as comparações entre as soluções obtidas com o IMA Regressor e com a Rede Neural implementada pelo software *NeuroSolutions*.



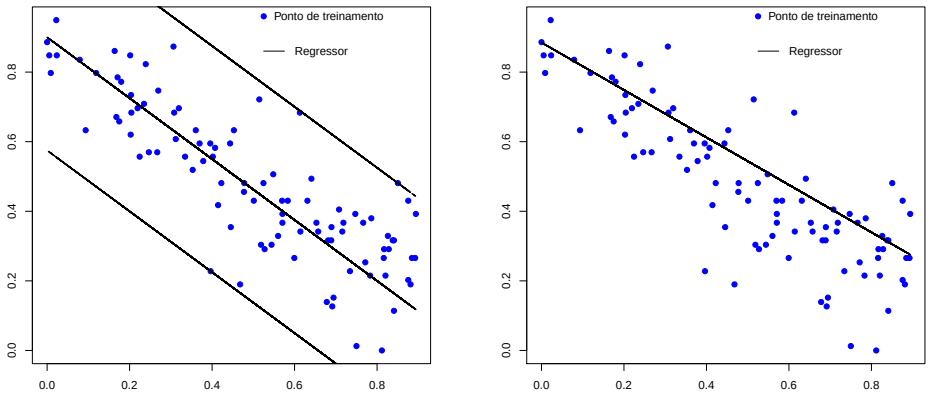
IMA Regressor Rede Neural

Figura 4.2: Resultados do IMA Regressor e da Rede Neural no Treinamento 1



IMA Regressor Rede Neural

Figura 4.3: Resultados do IMA Regressor e da Rede Neural no Treinamento 2



IMA Regressor Rede Neural

Figura 4.4: Resultados do IMA Regressor e da Rede Neural no Treinamento 3

4.1 Flexibilização

Uma forma de flexibilização da margem, sugerida por Boser, Guyon e Vapnik (1992), consiste na eliminação direta e progressiva de alguns pontos suportes. Essa técnica não poderia ser aplicada em um problema de classificação, dado que os pontos suportes são os que definem a margem do hiperplano separador. No caso da regressão porém, esses pontos identificam-se como *outliers*, à medida que seus valores estão afastados do plano do regressor, ou mesmo como valores que não são desejadas.

Para aplicar essa técnica, foi realizado um *3-fold-cross validation* e, inicial-

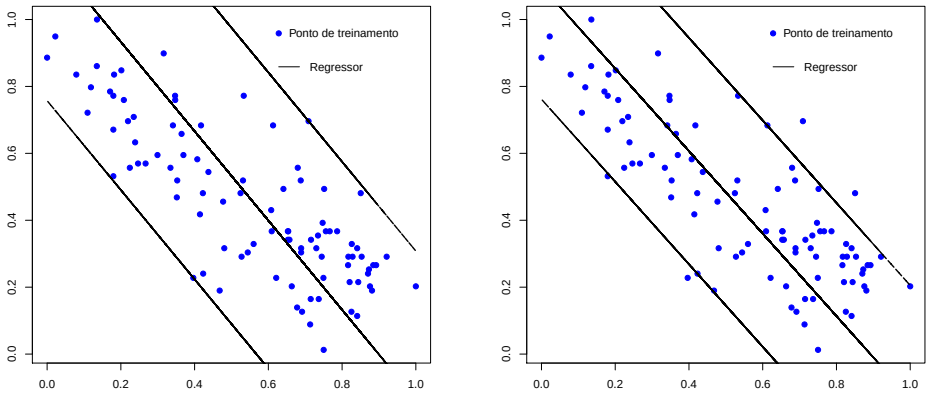
mente, para os conjuntos de treinamento foram usadas todas as amostras. Em seguida, identificou-se os pontos suportes, que foram então retirados do conjunto para a realização de um novo treinamento. Os resultados obtidos foram, então, aplicados nos conjuntos de testes. Essa técnica pode ser aplicada sucessivamente, até que se obtenha uma solução satisfatória.

A Tabela 4.2 apresenta as comparações entre os resultados obtidos nos testes realizados com o IMA Regressor flexibilizando ou não a margem, em que Conjunto 1, Conjunto 2 e Conjunto 3, representam as fases da validação cruzada. Nota-se que, de maneira geral, a flexibilização promoveu uma minimização na função de perda.

	IMA Regressor		IMA Regressor flexibilizado	
	MSE	MML	MSE	MML
Conjunto 1	0,01853	0,07627	0,01344	0,06312
Conjunto 2	0,01585	0,07576	0,01379	0,06766
Conjunto 3	0,01205	0,06327	0,01228	0,06346

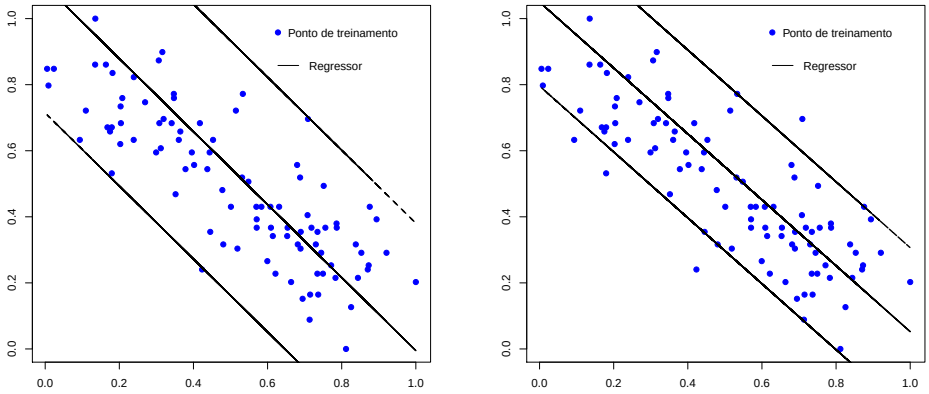
Tabela 4.2: Comparação do IMA Regressor com e sem flexibilização da margem, quanto aos critérios MSE e MML

As Figuras 4.5, 4.6 e 4.7 apresentam as comparações entre reta construída pelo IMA Regressor em comparação com a margem flexibilizada. Observa-se que os suportes eliminados no treinamento violam o raio do tubo, sendo identificados como *outliers*.



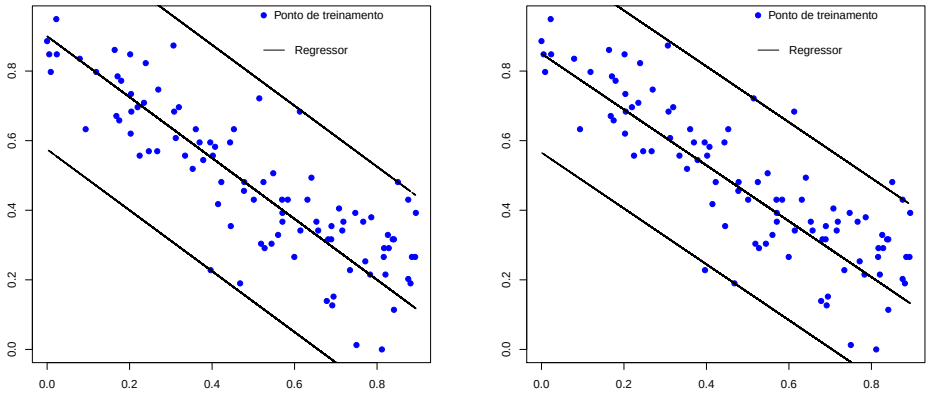
IMA Regressor IMA Regressor com flexibilização

Figura 4.5: Resultados do IMA Regressor com margem flexível no treinamento 1



IMA Regressor IMA Regressor com flexibilização

Figura 4.6: Resultados do IMA Regressor com margem flexível no treinamento 2



IMA Regressor IMA Regressor com flexibilização

Figura 4.7: Resultados do IMA Regressor com margem flexível no treinamento 3

5 Considerações Finais

Neste trabalho foram apresentados conceitos importantes sobre uma área em crescente expansão dentro do aprendizado de máquinas. Além disso, foi apresentado um novo método de treinamento *online* para o problema de regressão através do algoritmo de margem incremental(IMA).

Esse algoritmo utiliza uma única formulação baseada em um sistema de inequações, computa soluções equivalentes às soluções SV Regressor, não utiliza pacotes de programação linear ou não linear e garante sempre uma solução. Para tanto vale-se somente de uma estratégia de adaptação para o valor da margem, no caso da classificação, e do valor do raio do tubo, no caso da regressão, e a solução de um sistema de inequações.

O IMA Regressor apresentou excelentes resultados nos problemas abordados e também na comparação com a Rede Neural, considerado um dos métodos mais robustos. A proposta de flexibilização apresentada aponta caminhos interessantes para o desenvolvimento do estudo.

Como trabalho futuro fica a ideia de estender o IMA Regressor para a versão dual, com a possibilidade do uso de funções kernel, permitindo, assim, a solução de problemas mais complexos que são mais comumente encontrados.

Referências Bibliográficas

[1] Bi, J. e Bennett, K.P. *A geometric approach to support vector regression*. Neurocomputing, vol 55, 79-108, 2003.

[2] Boser, B.E., Guyon, L.M. e Vapnik, V.N. *A training algorithm for optimal margin classifiers*. In Proc. of the 5th Annual ACM workshop on COLT. ACM Press, 144-152, 1992.

[3] Duda, R.; Hart, P.; Stork, D.. *Pattern Classification*, second ed., Wiley, New York, 2000.

[4] Fonseca Neto, R. e Borges, C. C. H.. *Um novo método de treinamento online para o problema de regressão*. CBRN, 2007

[5] Haykin, S. *Redes Neurais, princípios e prática*. Bookman, 1999.

[6] Kivinen, J.; Smola, A.J.; Williamson, R.C.. *Online learning with kernels*, Trans. Signal Process. (2004).

[7] Leite, Saul C. e Fonseca Neto, R. *Incremental Margin Algorithm for large margin classifiers*. Neurocomputing, 2007.

[8] Lorena, A. C. e Carvalho, A. C. P. L. F.. *Introdução às Máquinas de Vetores Suporte*. Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo. São Carlos, 2003. Relatório Técnico. 0103-2569.

[9] Principe, J. C.; Euliano, N. R.; Lefebvre, W. C.. *Neural and adaptative systems*. Wiley, 2000.

[10] Rosenblatt, F.; *The perceptron: a probabilistic model for information storage and organization in the brain*, Psychol. Rev. 65 (1958).

[11] Smola, A. e Scholkopf, B.. *A tutorial on support vector regression*. Technical Report NC2-TR-1998-030, NeuroCOLT, 1998.

[12] SMOLA, A. J. et al. *Introduction to large margin classifiers*. In: . [S.l.]: Morgan-Kauffman, 1999. cap. 1, p. 1-28.

[13] Vapnik, V.. *The nature of statistical learning theory*. Springer-Verlag, New York, 1995.

[14] Widrow, B. e Hoff, M.E. *Adaptive switching circuits*. 1960 IRE WESCON Conv. Record, Part 4, 96-104, 1960.