Learning and Nonlinear Models - Journal of the Brazilian Society on Computational Intelligence (SBIC), Vol. 15, Iss. 1, pp. 16-44, 2017

MÁQUINAS DE VETORES-SUPORTE: UMA REVISÃO

Ajalmar R. da Rocha Neto
Instituto Federal do Ceará (IFCE)
Programa de Pós-Graduação em Ciência da Computação (PPGCC)
Programa de Pós-Graduação em Engenharia de Telecomunicações (PPGET)
ajalmar@ifce.edu.br

Resumo – Neste artigo é apresentada uma revisão detalhada sobre máquinas de vetores-suporte para problemas de classificação. Desta forma são apresentados diversos conceitos relacionados, tais como maximização da margem de separação, formulações das funções custo a serem minimizadas, utilização de margens rígidas, margens flexíveis e truque de *kernel*. Além disso, os algoritmos utilizados para treinamento das máquinas de vetores-suporte são descritos e discutidos. Neste artigo, como estudo de caso, são apresentados ainda diversos resultados obtidos a partir da aplicação das máqunas de vetores-suporte ao diagnóstico de patologias da coluna vertebral. Os dados para o problema alvo se apresentam em duas versões. Na primeira versão existem 3 classes, na qual o conjunto de dados contém indivíduos normais, com hérnia de disco e com espondilolistese; enquanto na segunda existem apenas 2 classes que se referem a indivíduos normais ou patológicos.

Palavras-chave – Máquinas de vetores-suporte, máquinas de vetores-supporte de mínimos quadrados, tarefas de classificação, métodos de kernel, classificadores de margem larga.

Abstract – In this paper we present a detailed review on support vector machines for classification tasks. Several related concept are presented, such as maximization of separation margin between classes, formulation of cost functions to be minimized, use of hard and soft margins and kernel trick. Besides that, training algorithms applied to obtain support vector machines are described and discussed. We also present several results with respect to support vector machines when applied to classify vertebral column pathologies. The data for such a problem has two versions. In the first one, there exists three classes and the individuals in the dataset are labeled as normal (healthy), disc hernia or spondylolisthesis; in the second version, the individuals are labeled as being normal or pathologic.

Keywords – Support vector machiens, least square support vector machines, classification tasks, kernel methods, large margin classifiers.

1 Introdução

O algoritmo *Generalized Portrait*, projetado para resolver problemas linearmente separáveis pode ser considerado o precursor dos classificadores SVM [1, 2]. Posteriormente, classificadores SVM foram também denominados classificadores de margem ótima (*Optimal Margin Classifiers*, por [3]); redes de vetores-suporte (*Support Vector Network*), por [4]; e então, a nomenclatura mais consolidada e difundida, máquinas de vetores-suporte [5].

Uma das justificativas para a difusão do uso de classificadores SVM pode estar em sua teoria matemática bem fundamentada, conforme será mostrado mais adiante neste artigo. Desenvolvimentos desta teoria levaram posteriormente à aplicação de SVMs não só em problemas de classificação de padrões, mas também em problemas de aproximação de funções. A abordagem que aplica SVM a problemas de aproximação de funções é denominada *Support Vector Regression* [6], enquanto a abordagem que aplica SVM a problemas de classificação de padrões é denominada *Support Vector Classification* [7].

O processo de aprendizagem de classificadores SVM tem por objetivo não apenas a minimização do risco empírico (*Empirical Risk Minimization*), como também busca a minimização do risco estrutural (*Structural Risk Minimization*). A minimização do risco empírico está associada à minimização do erro relacionado aos padrões de treinamento, enquanto a Minimização do Risco Estrutural está associada à minimização do erro associado aos padrões de teste (exemplos não-vistos no processo de aprendizagem). Desta maneira, o processo de aprendizagem busca aumentar a capacidade de generalização diretamente no processo de treinamento. Esta característica diferencia classificadores SVM de diversos classificadores de aprendizagem tradicionais, tais como as redes Perceptron Multicamadas (*Multilayer Peceptron* - MLP) e as redes RBF (*Radial Basis Function Networks*).

O processo de indução de classificadores usado em SVM é supervisionado. Desta forma, o processo de aprendizagem é realizado com base nos diversos pares de entrada e saída pertencentes à base de dados de exemplos. Outra característica comum em classificadores SVM consiste na formulação que objetiva a resolução de um problema binário. Apesar de haver formulação de SVM para problemas multiclasses [8], o que torna o problema de aprendizagem consideravelmente mais complexo, a combinação das saídas apresentadas por classificadores binários tem sido amplamente utilizadas. As abordagens um-contra-um [9], um-contra-todos [9], DAGSVM [10] e por códigos corretores de erros [11] são as mais amplamente utilizadas.

2 Definições e Conceitos Preliminares

Formalmente, o objetivo de um classificador SVM é estimar uma função $f : \mathbb{R}^N \rightarrow \{\pm 1\}$ usando um conjunto:

$$(\mathbf{x}_1, d_1), \ldots, (\mathbf{x}_n, d_n) \in \mathbb{R}^N \times \{\pm 1\}, \tag{1}$$

em que $\mathbf{x}_i$ e d_i representam o vetor de características e a classe da i-ésima amostra, respectivamente; e a função f deve classificar de forma correta outros exemplos $(\mathbf{x}_i, d_i)$ não utilizados na estimação da função, isto é, $f(\mathbf{x}_i) = d_i$. O conjunto de dados $(\mathbf{x}_1, d_1), \ldots, (\mathbf{x}_n, d_n) \in \mathbb{R}^N \times \{\pm 1\}$ usado para estimar f é denominado conjunto de treinamento. Enquanto o conjunto de dados $(\tilde{\mathbf{x}}_1, \tilde{d}_1), \ldots, (\tilde{\mathbf{x}}_n, \tilde{d}_n) \in \mathbb{R}^N \times \{\pm 1\}$ com os exemplos não utilizados na estimação de f é denominado conjunto de teste.

Considere um problema linearmente separável, como mostrado na Figura 1. Este tipo de problema apresenta infinitas soluções. Matematicamente, as soluções para este tipo de problema podem ser apresentadas na forma da equação de um hiperplano, ou seja

$$\mathbf{w}^T \mathbf{x} + b = 0 \ , \tag{2}$$

desde que valores específicos para o vetor de pesos $\mathbf{w} \in \mathbb{R}^2$ e intercepto (viés) $b \in \mathbb{R}$ consigam colocar todos os pontos $\mathbf{x} \in \mathbb{R}^2$ de uma determinada classe em lado oposto ao da outra, comparativamente à posição do hiperplano. Nesse sentido a classe de hiperplanos que separam as classes devem satisfazer as seguintes restrições:

$$\begin{aligned} \mathbf{w}^T \mathbf{x}_i + b \geq a \quad \text{para} \ \ d_i = +1 \\ \mathbf{w}^T \mathbf{x}_i + b \leq a \quad \text{para} \ \ d_i = -1 \end{aligned} \tag{3}$$

em que $a > 0$ e $i = 1 \ldots n$. Logo, as restrições acima devem ser satisfeitas para todos os padrões do conjunto de treinamento $(\mathbf{x}_1, d_1), \ldots, (\mathbf{x}_n, d_n)$.

No problema artificial apresentado na Figura 1 tem-se duas classes (com padrões descritos por quadrados para a classe positiva e triângulos para a classe negativa) em que se pode observar 5 hiperplanos, sendo cada um deles uma solução para o problema apresentado. Neste caso específico, as soluções são retas no $\mathbb{R}^2$.

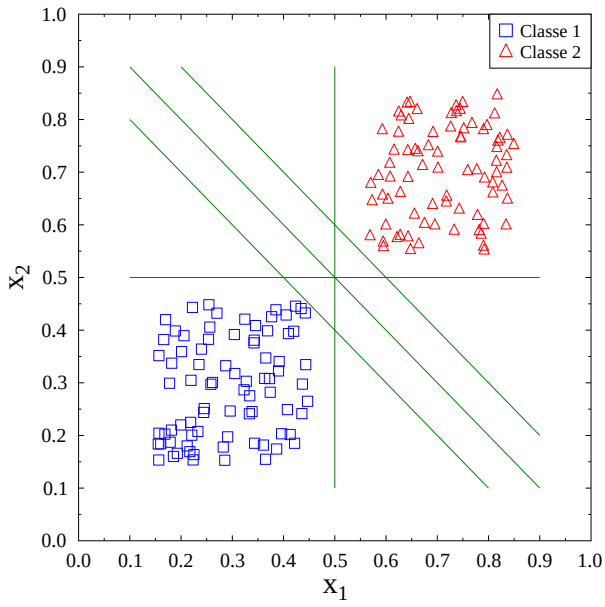
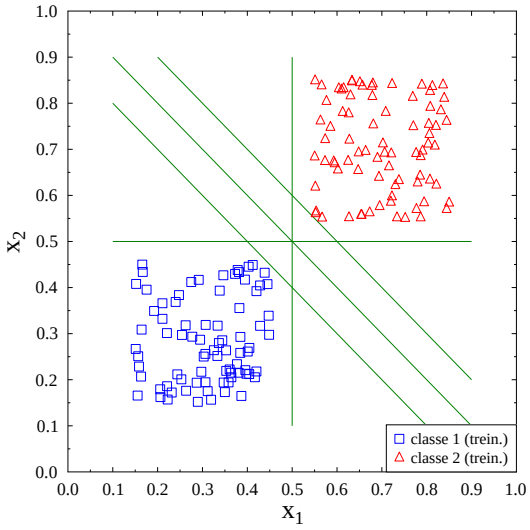


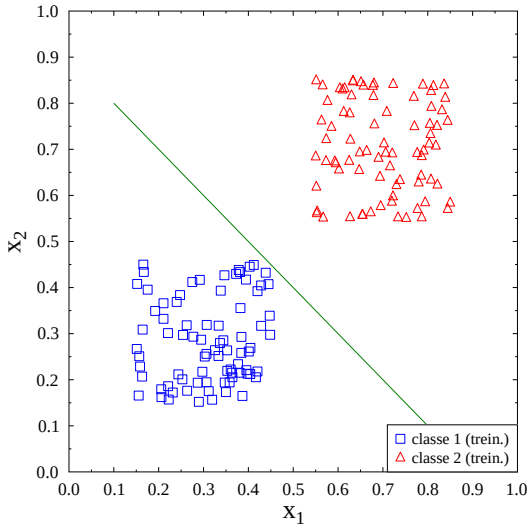
Figura 1: Os hiperplanos (retas) apresentados, descrevem algumas possíveis soluções para um problema linearmente separável. Duas classes hipotéticas são descritas por quadrados e triângulos, apresentando pontos gerados artificialmente.

Vale ressaltar que muitos problemas reais não são linearmente separáveis, porém esta suposição inicial permite apresentar de forma mais simples a idéia que fundamenta a teoria de SVM. Posteriormente, a restrição relacionada à linearidade do problema é removida.

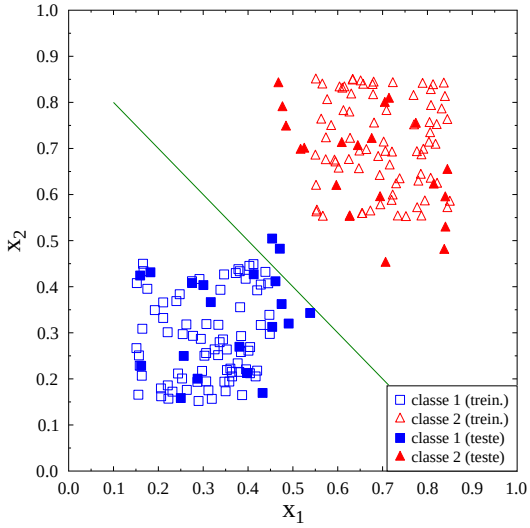
Neste momento, surge uma questão importante: Qual o melhor hiperplano de separação?. Este questionamento decorre da necessidade de não apenas minimizar o risco empírico como também minimizar o risco estrutural. A Figura 2 ilustra o problema relacionado à escolha do melhor hiperplano de separação para um problema linearmente separável. Uma escolha como a realizada na Figura 2(b) resolve o problema corretamente para todos os padrões de treinamento. No entanto, não resolve corretamente o problema quando também são considerados os padrões de teste, como pode ser visto na Figura 2(c). Neste sentido, uma escolha mais adequada seria o hiperplano apresentado na Figura 2(d), visto que este posiciona-se equidistante das classes e, desta forma, padrões que não foram utilizados no processo de treinamento podem ser classificados corretamente.



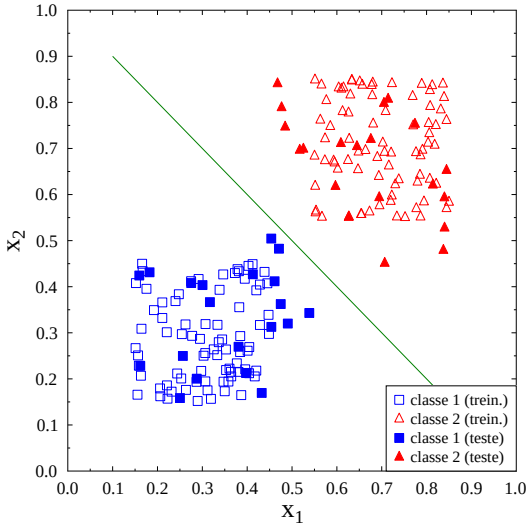
(a) Diversos hiperplanos capazes de resolver o problema linearmente separável. Nessa figura são mostrados apenas os padrões de treinamento.



(b) Problema resolvido por um hiperplano muito próximo a uma das classes.



(c) O mesmo problema descrito em (a), porém aqui são mostrados os padrões de treinamento e teste. Observe que padrões do conjunto de teste apresentam-se do lado aposto ao da sua classe.



(d) Problema resolvido por um hiperplano mais adequado, pois apresenta-se equidistante das duas classes.

Figura 2: Um problema linearmente separável e hiperplanos solução do problema.

Desta maneira, dentre todos os possíveis hiperplanos que solucionam um determinado problema, deve-se escolher um que tenha a máxima distância em relação aos padrões mais próximos do conjunto de treinamento. Denomina-se tal hiperplano de hiperplano ótimo, representado por

$$\mathbf{w}_o^T \mathbf{x} + b_o = 0. \tag{4}$$

Outro conceito importante é o de margem de separação ρ, que representa a menor distância entre o hiperplano ótimo e o padrão de treinamento mais próximo. A margem de separação é maximizada no processo de aprendizagem de classificadores SVM para obtenção da seguinte função discriminante:

$$f(\mathbf{x}) = \mathbf{w}_o^T \mathbf{x} + b_o, \tag{5}$$

a qual fornece uma medida algébrica da distância de $\mathbf{x}$ ao hiperplano ótimo. Em problemas de classificação de padrões, classificadores SVM utilizam a função sinal, ou seja

$$f(\mathbf{x}) = \text{sign}(\mathbf{w}_o^T \mathbf{x} + b_o). \tag{6}$$

tal que

$$f(\mathbf{x}) \quad = \quad \begin{cases} -1, & \text{se} \quad \mathbf{w}_o^T \mathbf{x} + b_o < 0, \\ +1, & \text{se} \quad \mathbf{w}_o^T \mathbf{x} + b_o \geq 0. \end{cases} \tag{7}$$

A fim de obter uma solução ótima, o vetor de pesos ótimo $\mathbf{w}_o$ e o viés ótimo b_o devem ser encontrados a partir do conjunto de treinamento $\{\mathbf{x}_i, d_i\}_{i=1}^n$. Para este fim, o problema apresentado na Equação (3) pode ser reescrito como

$$\begin{aligned} \mathbf{w}_o^T\mathbf{x}_i + b_o \geq +1 \quad \text{para } d_i = +1 \\ \mathbf{w}_o^T\mathbf{x}_i + b_o \leq -1 \quad \text{para } d_i = -1 \end{aligned} \tag{8}$$

Os padrões particulares $\{(\mathbf{x}^{(s)}, d^{(s)}) | s \in \{1 \dots N\}\}$ pertencentes ao conjunto de treinamento $\{\mathbf{x}_i, d_i\}_{i=1}^n$ e que satisfazem a Equação (8) com sinal de igualdade são denominados *vetores-suporte* (VS), ou seja

$$f(\mathbf{x}^{(s)}) \quad = \quad \begin{cases} -1, & \mathbf{w}_o^T\mathbf{x}^{(s)} + b_o = -1 \\ +1 & \mathbf{w}_o^T\mathbf{x}^{(s)} + b_o = +1. \end{cases} \tag{9}$$

Considere $\mathbf{x}_p$ como sendo a projeção de $\mathbf{x}$ sobre o hiperplano ótimo, e r a distância do ponto $\mathbf{x}$ até o hiperplano. Logo, um dado padrão $\mathbf{x}$ pode ser descrito da seguinte forma:

$$\mathbf{x} = \mathbf{x}_p + r\frac{\mathbf{w}_o}{\|\mathbf{w}_o\|}. \tag{10}$$

A distância r do ponto até o hiperplano pode ser obtida com base nas equações 5, 10 e no conhecimento do valor de f no ponto $\mathbf{x}_p$, $f(\mathbf{x}_p) = 0$, da seguinte forma:

$$\begin{aligned} f(\mathbf{x}) &= \mathbf{w}_o^T\mathbf{x} + b_o \\ f(\mathbf{x}) &= \mathbf{w}_o^T\left[\mathbf{x}_p + r\frac{\mathbf{w}_o}{\|\mathbf{w}_o\|}\right] + b_o \\ f(\mathbf{x}) &= \mathbf{w}_o^T\mathbf{x} + b_o + \mathbf{w}_o^T r\frac{\mathbf{w}_o}{\|\mathbf{w}_o\|} \\ f(\mathbf{x}) &= 0 + r\frac{\mathbf{w}_o^T\mathbf{w}_o}{\|\mathbf{w}_o\|} \\ f(\mathbf{x}) &= r\|\mathbf{w}_o\| \\ r &= \frac{f(\mathbf{x})}{\|\mathbf{w}_o\|} \end{aligned} \tag{11}$$

A distância $r^{(s)}$ dos VS ao hiperplano ótimo pode ser obtida a partir da Equação (11) e da Equação (9)[1], a saber

$$\begin{aligned} r^{(s)} &= \frac{f(\mathbf{x}^{(s)})}{\|\mathbf{w}_o\|} \\ r^{(s)} &= \frac{1}{\|\mathbf{w}_o\|} \end{aligned} \tag{12}$$

A interpretação geométrica da distância de um dado padrão $\mathbf{x}$ ao hiperplano ótimo pode ser visualizada na Figura 3. Quando um determinado padrão é um *vetor-suporte* $\mathbf{x}^{(s)}$, tem-se a distância $r^{(s)}$. Nesse contexto, a margem de separação ótima $\rho = 2r^{(s)}$ pode ser obtida a partir da Equação (12), tal que

$$\rho = 2r^{(s)} = \frac{2}{\|\mathbf{w}_o\|}. \tag{13}$$

A partir de [12, 13] pode-se inferir que a maximização da margem de separação ρ implica simultaneamente na minimização da dimensão VC (Vapnik-Chervonenkis). Esta dimensão está associada à complexidade da função discriminante que deve se adequar ao conjunto de treinamento, por exemplo, não sendo tão complexa a ponto de que ocorra um sobre-ajuste. Uma análise da Equação (13) permite verificar que, ao passo que se minimiza a norma $\|\mathbf{w}\|$ do vetor de pesos, obtém-se também a minimização da dimensão VC. Será visto adiante que isto decorre da incorporação do princípio da minimização do risco estrutural ao projeto de classificadores SVM, pois a formulação do problema com base neste classificador usa a norma do vetor de pesos ou termos decorrentes dela.

Seguindo o raciocínio descrito até o momento pode-se notar que a formulação do problema de obtenção do hiperplano ótimo pode começar a ser realizada, a partir da Equação (13), com a maximização da margem de separação ρ:

$$\max \rho = \max \frac{2}{\|\mathbf{w}\|}, \tag{14}$$

que ainda pode ser apresentado como

$$\min \|\mathbf{w}\| \iff \min \sqrt{\mathbf{w}^T\mathbf{w}} \iff \min \frac{1}{2}\mathbf{w}^T\mathbf{w}, \tag{15}$$

[1] A distância deve assumir valor positivo, logo o valor negativo da Equação 9 é desconsiderado.

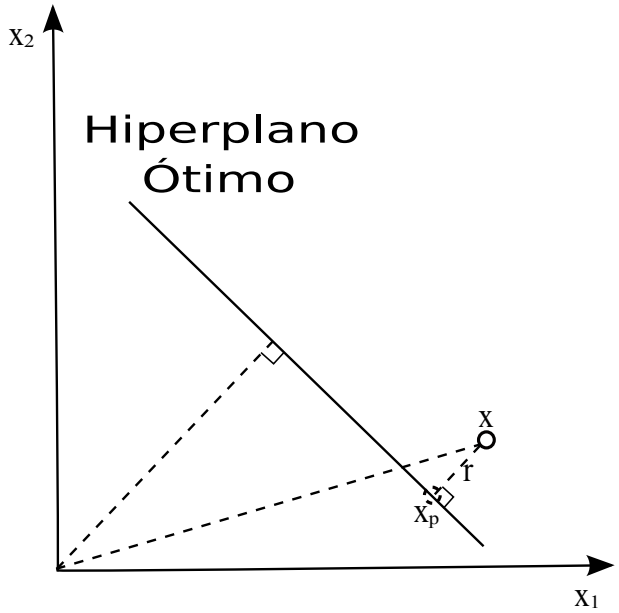


Figura 3: Interpretação geométrica da distância de um padrão $\mathbf{x}$ ao hiperplano ótimo.

com base na minimização da norma do vetor de pesos $\|\mathbf{w}\|$ ou de termos decorrentes.

A partir de agora deve-se considerar a função $\tau(\mathbf{w})$ a ser minimizada, como sendo a seguinte:

$$\tau(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{w}. \tag{16}$$

Toda a discussão realizada até este ponto é válida tanto para classificadores SVM, quanto para classificadores LSSVM. Na Seção 3 são apresentados os detalhes da formulação do problema de obtenção dos parâmetros ótimos ($\mathbf{w}_o$ e b_o) para classificadores SVM, enquanto os detalhes da teoria para classificadores LSSVM são descritos na Seção 6.

3 Fundamentos Teóricos do Classificador SVM

Maximizar a margem de separação $\rho = \frac{2}{\|\mathbf{w}\|}$ é equivalente a minimizar $\frac{1}{2}\|\mathbf{w}\|^2$ ou $\frac{1}{2}\mathbf{w}^T\mathbf{w}$. Portanto, o hiperplano que separa os dados de entrada pode ser descrito como um que minimize

$$\tau(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{w}, \tag{17}$$

satisfazendo a restrição

$$d_i(\mathbf{w}^T\mathbf{x}_i + b) \geq +1, \tag{18}$$

a qual é obtida pela combinação das duas linhas da Equação (8).

Logo, pode-se apresentar o problema clássico de obtenção dos parâmetros ótimos do classificador SVM como o seguinte problema de otimização:

$$\begin{aligned} &\min \tau(\mathbf{w}) = \min \frac{1}{2}\|\mathbf{w}\| = \min \frac{1}{2}\mathbf{w}^T\mathbf{w} \\ &s.a.\ \ d_i[(\mathbf{w}^T\mathbf{x}_i) + b)] \geq 1,\ \ i = 1, \ldots, n \end{aligned} \tag{19}$$

em que $\tau(\mathbf{w})$ representa a função-custo que deve ser minimizada.

Conforme mencionado anteriormente o problema apresentado na Equação (19) baseia-se na suposição de separabilidade linear das classes. Em outras palavras, assume-se que as duas classes sejam totalmente separáveis por um único hiperplano. Quando o problema de otimização é formulado com base nesta imposição, o classificador resultante é denominado SVM com margem rígida.

3.1 Classificador SVM com Margem Rígida

O problema de otimização com restrição apresentado na Equação (19) é chamado de problema primal. A função $\tau(\mathbf{w})$ é uma função convexa em $\mathbf{w}$, enquanto as restrições são lineares em $\mathbf{w}$. Usando o método de Lagrange pode-se construir a seguinte função lagrangeana:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{n} \alpha_i(d_i(\mathbf{x}_i^T\mathbf{w} + b) - 1), \tag{20}$$

em que os multiplicadores de Lagrange $\{\alpha_i\}_{i=1}^n$ são grandezas não-negativas (i.e. $\{\alpha_i \geq 0\}_{i=1}^n$).

A forma expandida da função lagrangeana apresenta-se como a seguir:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{n} \alpha_i d_i \mathbf{w}^T\mathbf{x}_i - b\sum_{i=1}^{n} \alpha_i d_i + \sum_{i=1}^{n} \alpha_i. \tag{21}$$

A solução é determinada pelo ponto de sela da função lagrangeana $L(\mathbf{w}, b, \boldsymbol{\alpha})$, que deve ser minimizada em relação a $\mathbf{w}$ e b e maximizada em relação a $\boldsymbol{\alpha}$. Assim, diferenciando em relação a $\mathbf{w}$ e b e igualando a zero, obtém-se as seguintes condições de otimização:

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\alpha})}{\partial \mathbf{w}} = 0 \tag{22}$$

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\alpha})}{\partial b} = 0 \tag{23}$$

que resultam em

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i d_i \mathbf{x}_i \tag{24}$$

e

$$\sum_{i=1}^{n} \alpha_i d_i = 0, \tag{25}$$

respectivamente.

Vale notar que o terceiro termo $(-b\sum_{i=1}^{n} \alpha_i d_i)$ da forma expandida da função lagrangeana apresentada na Equação (21) é zero, devido ao resultado obtido na Equação (25). Desta forma, pode-se reescrever a Equação (21) da seguinte forma:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{n} \alpha_i d_i \mathbf{w}^T\mathbf{x}_i + \sum_{i=1}^{n} \alpha_i, \tag{26}$$

Ao continuar no processo de resolução da função lagrangeana, pode-se notar que o segundo termo $(-\sum_{i=1}^{n} \alpha_i d_i \mathbf{w}^T\mathbf{x}_i)$ da Equação (21), quando aplicado o resultado obtido na Equação (24), equivale a $-\mathbf{w}^T\mathbf{w}$. Logo, pode-se mostrar que

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \mathbf{w}^T\mathbf{w} + \sum_{i=1}^{n} \alpha_i, \tag{27}$$

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = -\frac{1}{2}\mathbf{w}^T\mathbf{w} + \sum_{i=1}^{n} \alpha_i,$$

ou ainda, substituindo a equação acima pelo resultado apresentado na Equação (24), obtém-se

$$L(\boldsymbol{\alpha}) = -\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T\mathbf{x}_j + \sum_{i=1}^{n} \alpha_i \tag{28}$$

ou

$$L(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T\mathbf{x}_j. \tag{29}$$

A partir da análise da Equação (29) percebe-se que a mesma apresenta-se apenas em função dos multiplicadores de Lagrange $\{\alpha_i\}_{i=1}^n$.

A problema de otimização dual é formulado como

$$\max L(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T\mathbf{x}_j \tag{30}$$

$$s.a. \ \sum_{i=1}^{n} \alpha_i d_i = 0$$

$$s.a. \ \alpha_i \geq 0 \ \ para \ \ i = 1, \ldots, n$$

No processo de resolução do problema dual são obtidos os multiplicadores de Lagrange ótimos $\{\alpha_i^o\}_{i=1}^n$. Em seguida, o vetor de pesos $\mathbf{w}_o$ e o bias b_o, podem ser computados por

$$\mathbf{w}_o = \sum_{i=1}^{N} \alpha_i^o d_i \mathbf{x}_i \tag{31}$$

e

$$b_o = 1 - \mathbf{w}_o^T \mathbf{x}^{(s)}, \tag{32}$$

quando $d^{(s)} = 1$, em que $(\mathbf{x}^{(s)}, d^{(s)})$ representam um vetor suporte.

A função discriminante, como definida no problema primal pela Equação (5) para o hiperplano ótimo, pode ser reescrita para um função com base no problema dual, aplicando-se a Equação (31), resultando em

$$f(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i^o d_i \mathbf{x}_i^T \mathbf{x} + b_o. \tag{33}$$

3.2 Classificador SVM com Margem Flexível

Um hiperplano que separe sem erros todos os padrões pertencentes às duas classes, nem sempre existe. Principalmente, quando ocorre sobreposição entre os dados que compõem as classes, como pode ser verificado em diversos problemas reais. Assim, faz-se necessário uma formulação para o problema do classificador SVM que considere tal dificuldade, permitindo que alguns padrões sejam incorretamente classificados.

Uma motivação para tal situação é impedir que a função discriminante torne-se mais complexa do que se deve no espaço de entrada. A complexidade da função pode ser diminuída ao se permitir alguns erros com a intenção de obter um melhor desempenho. Essa situação pode ser percebida, facilmente, em problemas em que existem padrões que apresentam-se fora dos seus valores típicos, as chamadas amostras discrepantes (*outliers*). Assim, evita-se que por conta de uns poucos padrões a função tenha que ser demasiadamente complexa para a resolução do problema. Outro motivo para a flexibilidade da margem é evitar o sobre-ajustamento da superfície de decisão aos dados (*overfitting*).

Isto posto, a formulação a seguir permite uma relaxamento das restrições dos classificadores SVM com margens rígidas com base em um limiar, ou seja

$$d_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i, i = 1, \ldots, n, \tag{34}$$

em que $\xi_i \geq 0, i = 1, \ldots, n$. Os limiares ξ_i são chamados de variáveis de folga (*slack variables*).

Vale ressaltar que as variáveis de folga são obtidas automaticamente no processo de aprendizagem do classificador SVM [4]. No processo de aprendizagem, as variáveis de folga assumem valores não-negativos, ou seja, $\xi_i \geq 0$. Os vetores de treinamento que após o processo de aprendizagem não são *vetores-suporte* têm valor igual a zero. Vetores que ultrapassam a margem de separação, porém estão do lado correto em relação ao hiperplano de separação ótimo, possuem valores no intervalo $(0 < \xi_i \leq 1)$. Por fim, os padrões que estão em localização oposta ao da sua classe e classificados incorretamente, devido à sua posição em relação ao hiperplano ótimo assumem valores $\xi_i > 1$.

Nesse contexto, o problema de otimização (primal) para classificadores SVM com margem flexível é formulado como

$$\begin{aligned} \min \tau(\mathbf{w}, \boldsymbol{\xi}) = \min \left\{ \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^{n} \xi_i \right\} \\ s.a.\ \ d_i[(\mathbf{w}^T \mathbf{x}_i) + b) \geq 1 - \xi_i,\ \ i = 1, \ldots, n \\ s.a\ \ \xi_i \geq 0, i = 1, \ldots, n. \end{aligned} \tag{35}$$

em que a constante C é um parâmetro que faz uma regularização entre o primeiro $(\frac{1}{2}\mathbf{w}^T\mathbf{w})$ e o segundo termo $(C \sum_{i=1}^{n} \xi_i)$ da função-custo. Percebe-se que, além de minimizar a dimensão VC pela maximização da margem, classificadores SVM objetivam também minimizar os valores assumidos pelas variáveis de folga, buscando conseqüentemente a minimização dos erros permitidos.

De forma semelhante ao realizado para o classificador SVM com margem rígida a solução é determinada pelo ponto de sela da função lagrangeana $L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta})$, a qual deve ser minimizada em relação a $\mathbf{w}$, b e $\boldsymbol{\xi}$ e maximizada em relação a $\boldsymbol{\alpha}$ e $\boldsymbol{\beta}$, sendo a mesma da seguinte forma:

$$L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i (d_i(\mathbf{x}_i^T \mathbf{w} + b) - 1 + \xi_i) - \sum_{i=1}^{n} \beta_i \xi_i, \tag{36}$$

em que todos os elementos dos conjuntos $\boldsymbol{\alpha} = \{\alpha_i\}_{i=1}^n$, $\boldsymbol{\beta} = \{\beta_i\}_{i=1}^n$ e $\boldsymbol{\xi} = \{\xi_i\}_{i=1}^n$ possuem valores não-negativos. A forma expandida da função lagrangeana acima apresenta-se como

$$L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i d_i \mathbf{w}^T \mathbf{x}_i - b \sum_{i=1}^{n} \alpha_i d_i + \sum_{i=1}^{n} \alpha_i - \sum_{i=1}^{n} \alpha_i \xi_i - \sum_{i=1}^{n} \beta_i \xi_i. \tag{37}$$

Similarmente, a solução do problema exige que a função-custo seja diferenciada em relação a $\mathbf{w}$, b e $\boldsymbol{\xi}$ e igualada a zero, para que se obtenha as seguintes condições de otimização:

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha})}{\partial \mathbf{w}} = 0 \tag{38}$$

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha})}{\partial b} = 0 \tag{39}$$

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha})}{\partial \boldsymbol{\xi}} = 0 \tag{40}$$

A primeira condição, ao ser resolvida resulta em

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i d_i \mathbf{x}_i. \tag{41}$$

Enquanto a segunda, ao ser resolvida, resulta em

$$\sum_{i=1}^{n} \alpha_i d_i = 0. \tag{42}$$

E por último, ao resolver a terceira condição, tem-se

$$C = \alpha_i + \beta_i. \tag{43}$$

Ao aplicar no segundo termo $(C \sum_{i=1}^{n} \xi_i)$ da Equação (37) o resultado obtido na Equação (43), como também rearranjando o último $(\sum_{i=1}^{n} \alpha_i \xi_i)$ e o penúltimo $(\sum_{i=1}^{n} \beta_i \xi_i)$ termos da Equação (37) da seguinte forma $(\sum_{i=1}^{n} \xi_i(\alpha_i + \beta_i))$, então o resultado das duas operações apresentadas acima permite que a Equação (37) seja reescrita como:

$$L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + \sum_{i=1}^{n} \xi_i(\alpha_i + \beta_i) - \sum_{i=1}^{n} \alpha_i d_i \mathbf{w}^T \mathbf{x}_i - b\sum_{i=1}^{n} \alpha_i d_i + \sum_{i=1}^{n} \alpha_i - \sum_{i=1}^{n} \xi_i(\alpha_i + \beta_i). \tag{44}$$

Pode-se verificar que o segundo e o último termos da Equação (44) podem ser eliminados, resultando em:

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{n} \alpha_i d_i \mathbf{w}^T \mathbf{x}_i - b\sum_{i=1}^{n} \alpha_i d_i + \sum_{i=1}^{n} \alpha_i. \tag{45}$$

Daqui em diante o processo de resolução é similar ao realizado para o classificador SVM com margem rígida, como pode-se perceber analisando as Equações (45) e (21). Logo, pode-se notar que o terceiro termo $(-b \sum_{i=1}^{n} \alpha_i d_i)$ da forma expandida da função lagrangeana apresentada na Equação (45) é zero, devido ao resultado obtido na Equação (42). Então a Equação (45) pode ser reescrita da seguinte forma:

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{n} \alpha_i d_i \mathbf{w}^T \mathbf{x}_i + \sum_{i=1}^{n} \alpha_i, \tag{46}$$

Nota-se também que ao aplicar o resultado obtido na Equação (41) ao segundo termo $(-\sum_{i=1}^{n} \alpha_i d_i \mathbf{w}^T \mathbf{x}_i)$ da Equação (46), obtém-se $-\mathbf{w}^T\mathbf{w}$. Logo, tem-se que:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \mathbf{w}^T\mathbf{w} + \sum_{i=1}^{n} \alpha_i, \tag{47}$$

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = -\frac{1}{2}\mathbf{w}^T\mathbf{w} + \sum_{i=1}^{n} \alpha_i.$$

Substituindo a equação acima pelo resultado apresentado na Equação (41), a fim de que a função não dependa de $\mathbf{w}$, obtém-se então

$$L(\boldsymbol{\alpha}) = -\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T \mathbf{x}_j + \sum_{i=1}^{n} \alpha_i, \tag{48}$$

ou

$$L(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T \mathbf{x}_j. \tag{49}$$

Por conseqüência, a Equação (49) apresenta-se apenas em função dos multiplicadores de Lagrange $\{\alpha_i\}_{i=1}^n$. O problema de otimização dual para o classificador SVM com margem flexível é dado por

$$\max L(\boldsymbol{\alpha}) = \max \left\{ \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T \mathbf{x}_j \right\} \tag{50}$$

$$s.a. \ \sum_{i=1}^{n} \alpha_i d_i = 0$$

$$s.a. \ 0 \leq \alpha_i \leq C \ \ para \ \ i = 1, \ldots, n$$

Ao analisar a Equação (50) pode-se perceber apenas uma diferença em relação à Equação (30) que é a limitação superior dos multiplicadores de Lagrange $0 \leq \alpha_i \leq C$. Todo o resto da equação mantém-se da mesma forma como a apresentada anteriormente. Este fato surge em decorrência da relação $\alpha_i + \beta_i = C$ apresentada na Equação (43). Como $\beta_i \geq 0$, então o menor valor assumido por β é zero, justamente quando $\alpha_i = C$.

Apesar do uso da margem flexível pelo classificador SVM, pode haver situações em que o problema não é resolvido de forma satisfatória, mesmo tolerando alguns erros de classificação. Um exemplo deste tipo de problema é ilustrado na Figura (4), que traz um problema de classificação binário hipotético, de natureza não-linearmente separável.

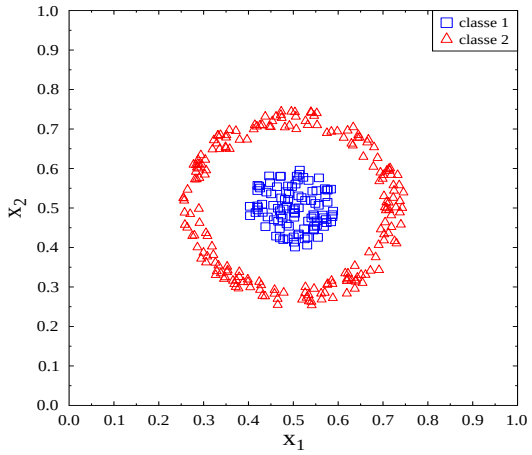


Figura 4: Exemplo de problema binário em que um hiperplano não resolve de forma satisfatória, mesmo considerando a possibilidade de alguns erros.

4 O Truque do *Kernel*

Usar um mapeamento não-linear é um conceito chave para se manipular problemas não linearmente separáveis, como o apresentado na Figura (4). Assim, a fim de se obter um problema linear a partir de um não-linear, o conjunto de dados deve ser transformado para um espaço de características de elevada dimensão. Ou seja, o espaço de entrada de um padrão $\mathbf{x}_i \in \mathbb{R}^N$ é mapeado para um espaço de características $\phi(\mathbf{x}) \in \mathbb{R}^M$, tal que $M > N$. Afortunadamente, a construção explícita de um mapeamento $\phi(\mathbf{x})$ ou do espaço de características não é necessária em métodos baseados em SVM.

Ademais, para qualquer função contínua e simétrica $K(\mathbf{x}, \mathbf{y})$ que satisfaça o teorema de Mercer [14], há um espaço de Hilbert H, um mapeamento $\phi : \mathbb{R}^N \rightarrow H$ e um número $\beta_i > 0$, tal que

$$K(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{M} \beta_i \tilde{\phi}_i(\mathbf{x}) \tilde{\phi}_i(\mathbf{y}), \tag{51}$$

em que $\mathbf{x}, \mathbf{y} \in \mathbb{R}^N$ e M é a dimensionalidade do espaço de Hilbert. O teorema de Mercer exige que para qualquer função quadrada integrável $g(.)$, tal que $g(.) \neq 0$,

$$\int K(\mathbf{x}, \mathbf{y}) g(\mathbf{x}) g(\mathbf{y}) d\mathbf{x} d\mathbf{y} \geq 0. \tag{52}$$

Por conseguinte, ao se definir $\phi_i(.) = \tilde{\phi}_i(.)\sqrt{\beta_i}$ pode-se escrever a Equação (51) como

$$K(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{M} \tilde{\phi}_i(\mathbf{x}) \sqrt{\beta_i} \tilde{\phi}_i(\mathbf{y}) \sqrt{\beta_i}, \tag{53}$$

$$K(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{M} \phi_i(\mathbf{x}) \phi_i(\mathbf{y}),$$

de modo que a função kernel $K(\mathbf{x}, \mathbf{y})$ pode ser representada pelo produto interno dos vetores no espaço de características, ou seja

$$K(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{M} \phi_i(\mathbf{x})\phi_i(\mathbf{y}) \tag{54}$$

$$K(\mathbf{x}, \mathbf{y}) = \langle \boldsymbol{\phi}_i(\mathbf{x}), \boldsymbol{\phi}_i(\mathbf{y}) \rangle \tag{55}$$

$$K(\mathbf{x}, \mathbf{y}) = \boldsymbol{\phi}_i^T(\mathbf{x})\boldsymbol{\phi}_i(\mathbf{y}). \tag{56}$$

A Equação (56) é comumente chamada de truque de *kernel* (*Kernel Trick*), pois permite que se evite o uso explícito do espaço de características de elevada dimensão. O truque de *kernel* permite que não seja necessário realizar o mapeamento para que se obtenha o vetor $\boldsymbol{\phi}_i(\mathbf{x})$, desde que se conheça uma função que descreva o produto interno $\langle \boldsymbol{\phi}_i(\mathbf{x}), \boldsymbol{\phi}_i(\mathbf{y}) \rangle$. Exemplos destas funções (de kernel) são apresentadas na Tabela 1. Desta forma, pode-se trabalhar indiretamente em um este espaço de elevada dimensão, desde que as computações sejam realizadas em um outro espaço, denominado Espaço *Kernel* (*Kernel Space*). Neste contexto, deve-se considerar que é mais provável a obtenção de um hiperplano de separação no espaço de elevada dimensão do que no espaço de entrada e, assim, um problema que é não linearmente separável no espaço de entrada pode ser resolvido adequadamente neste espaço aumentado. As representações dos Espaços de Entrada, de Características e de *Kernel* podem ser visualizadas na Figura 5. Esta figura é derivada de outra apresentada em [15].

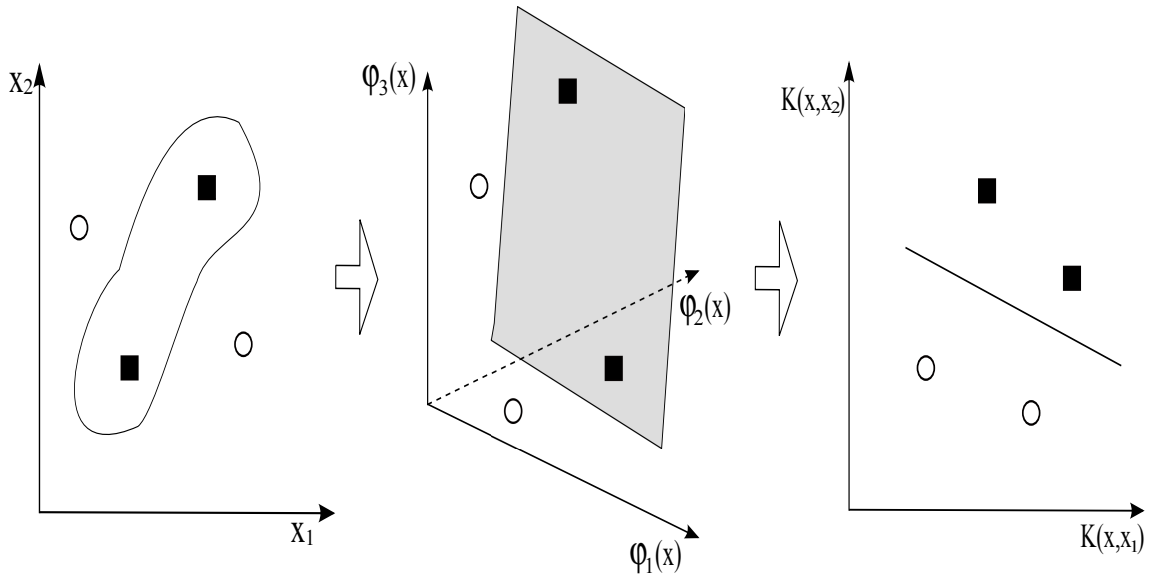


Figura 5: Espaço de entrada no $\mathbb{R}^2$, Espaço de Características no $\mathbb{R}^3$ e Espaço *Kernel*.

Nesse sentido, pode-se redefinir as Equações (30) e (50) referentes ao classificador SVM com margem rígida e ao com margem flexível para

$$\max L(\boldsymbol{\alpha}) = \max \left\{ \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j d_i d_j K(\mathbf{x}_i, \mathbf{x}_j) \right\} \tag{57}$$

$$s.a. \ \sum_{i=1}^{n} \alpha_i d_i = 0$$

$$s.a. \ \alpha_i \geq 0 \ \ para \ \ i = 1, \ldots, n$$

e

$$\max L(\boldsymbol{\alpha}) = \max \left\{ \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j d_i d_j K(\mathbf{x}_i, \mathbf{x}_j) \right\} \tag{58}$$

$$s.a. \ \sum_{i=1}^{n} \alpha_i d_i = 0$$

$$s.a. \ 0 \leq \alpha_i \leq C \ \ para \ \ i = 1, \ldots, n.$$

Similarmente, a Equação (33), que descreve a função discriminante no espaço de características, pode ser reescrita como

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i^o d_i K(\mathbf{x}_i, \mathbf{x}) + b_o. \tag{59}$$

Desta forma a classe para um dado vetor de teste $\mathbf{x}$ pode ser determinada a partir da função sinal, ou seja

$$f(\mathbf{x}) = sign\left(\sum_{i=1}^{n} \alpha_i^o d_i K(\mathbf{x}_i, \mathbf{x}) + b_o\right), \tag{60}$$

em que valores positivos indicam que o padrão $\mathbf{x}$ pertence à classe $y_i = +1$, enquanto valores negativos indicam que o padrão $\mathbf{x}$ pertence à classe $y_i = -1$.

Os *kernels* mais comumente utilizados são o linear, polinomial e gaussiano (RBF). Nesta trabalho utiliza-se ainda o kernel KMOD (*Kernel with MODerate decreasing*) [16]. Estes *kernels* são apresentados na Tabela 1, enquanto o KMOD é detalhado na subseção seguinte.

Kernel	Descrição
Linear	$K(\mathbf{x}, \mathbf{x}_i) = \mathbf{x}_i^T.\mathbf{x}$
Polinomial	$K(\mathbf{x}, \mathbf{x}_i) = \left(\mathbf{x}_i^T.\mathbf{x} + 1\right)^d$, em que d é o grau do polinômio.
Gaussiano (RBF)	$K(\mathbf{x}, \mathbf{x}_i) = \exp\left\{-\|\|\mathbf{x} - \mathbf{x}_i\|\|^2/\sigma^2\right\}$, em que σ é uma constante.

Tabela 1: Listagem de importantes *kernels*.

4.1 Kernel with MODerate decreasing (KMOD)

O *kernel* KMOD [16] caracteriza-se por apresentar um rápido decrescimento da imagem do pontos originais próximo à origem e um decrescimento moderado em direção ao infinito. Estas características permitem que sejam considerados vetores de entrada bastante distantes e ao mesmo tempo mantidas as informações de proximidade.

O *kernel* KMOD é descrito por

$$K(\mathbf{x}, \mathbf{x}_i) = a.\left[\exp\left\{\frac{\gamma}{(||\mathbf{x} - \mathbf{x}_i||^2 + \sigma^2)}\right\} - 1\right], \tag{61}$$

em que a é uma constante de normalização, tal que

$$a = \frac{1}{\exp\left(\frac{\gamma}{\sigma^2}\right) - 1}, \tag{62}$$

e σ e γ são os parâmetros do *kernel*.

Vale destacar que tanto o *kernel* KMOD quanto o RBF são descritos em função da distância, ou seja

$$K(\mathbf{x}, \mathbf{x}_i) = K(||\mathbf{x} - \mathbf{x}_i||^2). \tag{63}$$

E por isto, os seus comportamentos podem ser comparados com base em curvas que descrevam os valores de $K(\mathbf{x}, \mathbf{x}_i)$ em função da distância. As características de decrescimento mais significativo para pequenos valores de distância e decrescimento mais moderado em direção ao infinito para o *kernel* KMOD podem ser verificadas nas Figura 6. Para fins de comparação, nesta figura apresenta-se ainda o comportamento do *kernel* RBF.

5 Obtenção dos Parâmetros Ótimos para o Classificador SVM

5.1 Solução baseada em Programação Quadrática

Há diversas métodos disponíveis na literatura para resolução de problemas de Programação Quadrática (do inglês, *Quadratic Programming Problem* ou *QP Problem*). Bem como, há diversas implementações de tais métodos em pacotes de software, dentre os quais destacam-se Matlab, Scilab e Octave [2].

No contexto da resolução de problemas quadráticos, e considerando o conjunto de treinamento $\{\mathbf{x}_i\}_{i=1}^n$ e os multiplicadores de Lagrange $\{\alpha_i\}_{i=1}^n$, o problema de otimização formulado na Equação (58) pode ser apresentado como

$$\min L(\boldsymbol{\alpha}) = \frac{1}{2}\boldsymbol{\alpha}^T Q \boldsymbol{\alpha} - \boldsymbol{\alpha}^T \mathbf{1} \tag{64}$$

[2]No Octave e no Matlab há a função *qp* para fins de resolução de problemas quadráticos, enquanto no Scilab há a função *qpsolve*.

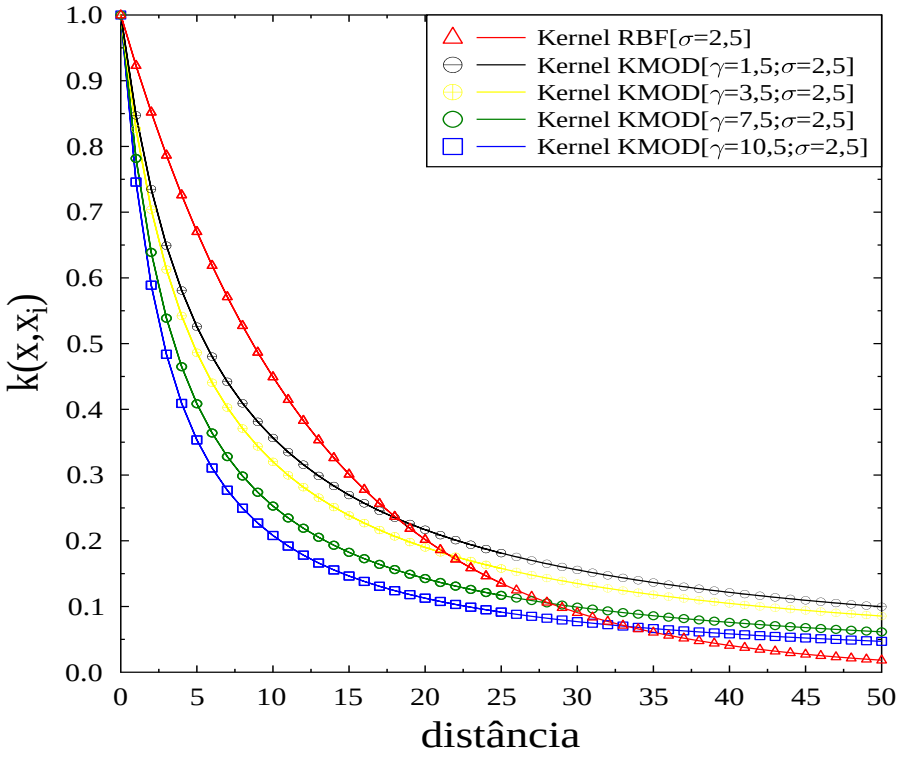


Figura 6: Comportamento dos *kernels* RBF e KMOD.

sujeito a

$$\boldsymbol{\alpha}^T \mathbf{d} = 0 \tag{65}$$
$$0 \leq \alpha_i \leq C \quad i = 1 \ldots n,$$

em que Q é uma matriz nxn, sendo seus elementos descritos por $q_{i,j} = d_i d_j K(\mathbf{x}_i, \mathbf{x}_j)$ tal que $i, j = 1 \ldots n$. Em geral, a matriz Q é densa e semi-definida positiva, e caso apresente-se com um valor elevado para n, tal matriz pode ser grande demais para ser armazenada [17]. Em virtude disto, um método *online* pode ser útil, pois este não exige uma grande quantidade de memória.

5.2 Solução baseada no Algoritmo SMO

Sequential Minimal Optimization (SMO) é um algoritmo iterativo para solucionar o problema de otimização dual dos classificadores SVM [18]. O algoritmo seleciona um par de parâmetros, α_p e α_q, do conjunto de multiplicadores de Lagrange, $\{\alpha_i\}_{i=1}^{l}$, e otimiza o valor da função objetivo conjuntamente para ambos os valores α_p e α_q. Ao final do algoritmo, o valor do viés b é ajustado com base no novo conjunto de parâmetros. Este processo é repetido até a convergência do conjunto de multiplicadores de Lagrange. O pseudocódigo do algoritmo SMO é descrito a seguir.

1. Iniciar $\alpha_i \leftarrow 0$ e $b \leftarrow 0$;
2. Considerar $f^{'}(\mathbf{x}) = \sum_{i=1}^{l} y_i \alpha_i k(\mathbf{x}, \mathbf{x}_i) + b$;
3. Considerar $E_i = f^{'}(\mathbf{x}_i) - y_i$;
4. Considerar λ como a tolerância;
5. Laço principal
   (a) Usar heurísticas para escolher um par de multiplicadores de Lagrange, α_p e α_q, de $\{\alpha_i\}_{i=1}^{n}$ a fim de otimizar conjuntamente;
   (b) Se não for capaz de encontrar multiplicadores a otimizar; então sair do laço principal;
   (c) Calcular μ, tal que $\mu \leftarrow \frac{E_q - E_p}{k(\mathbf{x}_p, \mathbf{x}_p) - 2k(\mathbf{x}_p, \mathbf{x}_q) + k(\mathbf{x}_q, \mathbf{x}_q)}$;
   (d) Atualizar $\alpha_q^{new} \leftarrow \alpha_q + y_q \mu$;
   (e) Verificar os limites aplicados a α_q;
   (f) Atualizar $\alpha_p^{new} \leftarrow \alpha_p - y_p \mu$;

6. Atualizar b tal que

   (a) $b_p \leftarrow E_p + y_p(\alpha_p^{new} - \alpha_p)k(\mathbf{x}_p, \mathbf{x}_p) + y_q(\alpha_q^{new} - \alpha_q)k(\mathbf{x}_q, \mathbf{x}_q) + b$

   (b) $b_q \leftarrow E_q + y_p(\alpha_p^{new} - \alpha_p)k(\mathbf{x}_p, \mathbf{x}_p) + y_q(\alpha_q^{new} - \alpha_q)k(\mathbf{x}_q, \mathbf{x}_q) + b$

   (c) $b = (b_p + b_q)/2$;

O algoritmo acima possui diversas simplificações, sendo que o algoritmo implementado para avaliação é descrito mais completamente no trabalho de [18]. Neste trabalho, Platt apresenta muito mais propriedades do algoritmo SMO, tanto no que se refere à sua eficiência quanto à estabilidade.

5.3 Solução baseada no Kernel Adatron

Adatron [19] é um método de aprendizado online que utiliza apenas informação de primeira ordem (gradiente) da função-custo, e que possui convergência garantida em relação à solução ótima, com uma taxa exponencial. *Kernel Adatron* é uma extensão do método Adatron que faz uso do truque do *kernel*. A aplicação do Kernel Adatron a SVM é primeiro descrita no trabalho de [20]. Outro trabalho correlato relacionado ao desenvolvimento do Kernel Adatron é o trabalho de [21]. Uma representação do *Kernel* Adatron em uma topologia similar a de redes neurais pode ser visualizada na Figura (7).

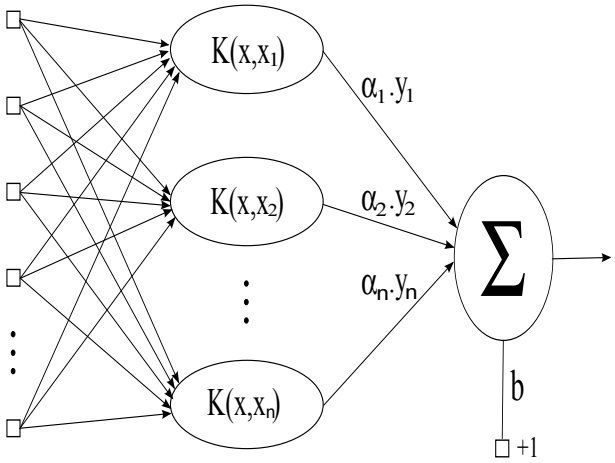


Figura 7: *Kernel* Adatron representado de forma similar às redes neurais artificiais.

A formulação do problema baseada no Kernel Adatron, pode ser descrita com base em um classificador SVM com margem flexível, como descrita na Equação (58), ou seja

$$\max L(\boldsymbol{\alpha}) = \max \left\{ \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j d_i d_j K(\mathbf{x}_i, \mathbf{x}_j) \right\} \tag{66}$$

$$s.a. \ \sum_{i=1}^{n} \alpha_i d_i = 0$$

$$s.a. \ 0 \leq \alpha_i \leq C \ \ para \ \ i = 1, \ldots, n.$$

Considerando apenas informação de primeira ordem, pode-se determinar o gradiente da função-custo apresentada na Equação (66), em relação a cada multiplicador de Lagrange, ou seja

$$\frac{\partial L}{\partial \alpha_i} = 1 - d_i \sum_{i=1}^{n} \alpha_i d_i K(\mathbf{x}_i, \mathbf{x}_j)$$

e

$$\frac{\partial L}{\partial \alpha_i} = 1 - d_i \left(\sum_{i=1}^{n} \alpha_i d_i K(\mathbf{x}_i, \mathbf{x}_j) \right). \tag{67}$$

A partir deste resultado, pode-se reescrever a Equação (67) de modo que o gradiente da função-custo $\frac{\partial L}{\partial \alpha_i}$ pode ser expresso em termos da função discriminante sem viés apresentada na Equação (59), tal que

$$\frac{\partial L}{\partial \alpha_i} = 1 - d_i f(\mathbf{x}). \tag{68}$$

Logo, o conjunto de multiplicadores de Lagrange $\{\alpha_i\}_{i=1}^n$ pode ser atualizado com base na seguinte expressão:

$$\begin{aligned}\Delta\alpha_i &= \alpha_i(t+1) - \alpha_i(t)\\ \alpha_i(t+1) &= \alpha_i(t) + \Delta\alpha_i\\ \alpha_i(t+1) &= \alpha_i(t) + \mu\frac{\partial L}{\partial \alpha_i}\\ \alpha_i(t+1) &= \alpha_i(t) + \mu(1 - d_i f(\mathbf{x}))\end{aligned} \tag{69}$$

A fim de garantir que os multiplicadores de Lagrange $\{\alpha_i\}_{i=1}^n$ tenham valores de acordo com as restrições do problema dual, a atualização de cada variável α_i deve ser feita considerando as equações:

$$\begin{array}{ll}\alpha_i(t+1) = \alpha_i(t) + \Delta\alpha_i, & se\ \ 0 \leq\ \ \alpha_i(t) + \Delta\alpha_i\ \ \leq\ \ C\\ \alpha_i(t+1) = 0, & se\ \ \alpha_i(t) + \Delta\alpha_i\ \ <\ \ 0\\ \alpha_i(t+1) = C, & se\ \ \alpha_i(t) + \Delta\alpha_i\ \ >\ \ C\end{array}$$

O critério de parada mais usualmente utilizado é o número de épocas de treinamento. Vale ressaltar que ao final de cada época deve ser estimado um valor para o viés. Uma expressão para esta estimativa é dada por

$$b = \frac{\max f^-(\mathbf{x}_i) + \min f^+(\mathbf{x}_i)}{2} \tag{70}$$

Outro critério de parada que pode ser utilizado em conjunto com o número de épocas de treinamento [3], refere-se à verificação da margem que pode ser expressa como:

$$\kappa = \frac{\min f^+(\mathbf{x}_i) - \max f^-(\mathbf{x}_i)}{2}, \tag{71}$$

em que ocorre a parada quando $\kappa < \epsilon$, em que ϵ é um número positivo.

6 Fundamentos Teóricos do Classificador LSSVM

Classificadores LSSVM [22] também são capazes de resolver problemas de classificação de padrões e problemas de aproximação de funções[4]. Em sua formulação original, classificadores LSSVM são projetados para resolver problemas binários, mas também podem ser combinados para resolver problemas multiclasses [23].

Classificadores LSSVM apresentam duas modificações importantes em relação aos classificadores SVM. Ambos estão presentes na formulação do problema primal de otimização. A primeira mudança está na restrição, a qual apresenta-se para o classificador LSSVM como uma igualdade[5]. A segunda mudança refere-se à incorporação na função-custo da soma das variáveis de folga ao quadrado, $\sum_{i=1}^{l}\xi_i^2$, ponderada por uma constante de regularização γ. Esta constante tem o mesmo significado que a constante C apresentada no problema do classificador SVM. Como conseqüência desta reformulação, o problema de otimização pode ser resolvido de uma maneira mais simples, a partir de um sistema de equações lineares, mais precisamente de um sistema Karush-Khun-Tucker (KKT) [24].

Do exposto, o problema de otimização para o classificador LSSVM, apresenta-se na seguinte forma:

$$\begin{aligned}\min \tau(\mathbf{w}, \boldsymbol{\xi}) &= \frac{1}{2}\mathbf{w}^T\mathbf{w} + \gamma\frac{1}{2}\sum_{i=1}^{n}\xi_i^2\\ s.a.\ \ d_i[(\mathbf{w}^T\mathbf{x}_i) + b] &= 1 - \xi_i,\ \ i = 1, \ldots, n\end{aligned} \tag{72}$$

em que $\tau(\mathbf{w}, \boldsymbol{\xi})$ representa a função-custo que deve ser minimizada. Pode-se perceber que as variáveis de folga $\{\xi_i\}_{i=1}^n$ podem assumir valores negativos, situação diferente da que ocorre no processo de aprendizagem do classificador SVM.

A simplificação do problema a ser resolvido pelo classificador LSSVM em relação ao classificador SVM é um ponto forte daquele classificador. No entanto, há algumas limitações que emergem desta reformulação. Uma delas refere-se à perda da natureza esparsa na solução do problema. Em outras palavras, os multiplicadores de Lagrange $\{\alpha_i\}_{i=1}^n$ obtidos no processo de aprendizagem do classificador LSSVM são não-nulos[6]. Desta forma, após o processo de aprendizagem, é necessário armazenar todos os padrões de treinamento, bem como os multiplicadores de Lagrange associados para fins de composição da função discriminante. Esta situação é particularmente indesejável quando a base de dados for muito grande.

[3] O que ocorrer primeiro, por exemplo.

[4] Porém esta trabalho trata de problemas de classificação de padrões, e assim sendo a atenção maior é dada para classificadores LSSVM que são obtidas para resolver tais problemas.

[5] Enquanto nos classificadores SVM a restrição do problema é de desigualdade, veja a Equação (18).

[6] Eventualmente um valor pode assumir valor nulo.

A função lagrangeana $L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha})$ para o problema de otimização primal apresenta-se então da seguinte maneira:

$$L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + \gamma\frac{1}{2}\sum_{i=1}^{n}\xi_i^2 - \sum_{i=1}^{n}\alpha_i(d_i(\mathbf{x}_i^T\mathbf{w} + b) - 1 + \xi_i), \tag{73}$$

em que os valores dos elementos pertences ao conjunto $\{\alpha_i\}_{i=1}^n$ são quase sempre não-nulos.

As condições para fins de otimização, de forma similar ao que ocorre com o problema do classificador SVM são dadas por

$$\begin{aligned}
\frac{\partial L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha})}{\partial \mathbf{w}} = 0 \quad &\Rightarrow \quad \mathbf{w} = \sum_{i=1}^{n}\alpha_i d_i \mathbf{x}_i \\
\frac{\partial L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha})}{\partial b} = 0 \quad &\Rightarrow \quad \sum_{i=1}^{n}\alpha_i d_i = 0 \\
\frac{\partial L(\mathbf{w}, b, \xi, \alpha)}{\partial \alpha_i} = 0 \quad &\Rightarrow \quad d_i(\mathbf{x}_i^T\mathbf{w} + b) - 1 + \xi_i = 0 \\
\frac{\partial L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha})}{\partial \xi_i} = 0 \quad &\Rightarrow \quad \alpha_i = \gamma\xi_i
\end{aligned} \tag{74}$$

Por conseguinte, pode-se formular a partir das condições acima descritas um sistema de equações lineares, $\mathbf{Ax} = \mathbf{B}$, a fim de representar o problema dos classificadores LSSVM, a saber:

$$\begin{bmatrix} 0 & \mathbf{d}^T \\ \mathbf{d} & \Omega + \gamma^{-1}\mathbf{I} \end{bmatrix} \begin{bmatrix} b \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{1} \end{bmatrix}$$

em que $\Omega_{i,j} = y_i y_j \mathbf{x}_i{}^T\mathbf{x}_j$, $\mathbf{d} = [d_1 \;\; d_2 \;\; \ldots \;\; d_n]^T$, $\boldsymbol{\alpha} = [\alpha_1 \;\; \alpha_2 \;\; \ldots \;\; \alpha_n]^T$ e $\mathbf{1} = [1 \;\; 1 \;\; \ldots \;\; 1]^T$, tal que o tamanho do vetor $\mathbf{1}$ é igual a n. E ainda, ao se utilizar o truque do *Kernel* pode-se redefinir $\Omega_{i,j}$ para $\Omega_{i,j} = y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$. Assim, a saída do classificador LSSVM pode também ser calculada por

$$f(\mathbf{x}) = sign\left(\sum_{i=1}^{l}\alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) + b\right). \tag{75}$$

7 Obtenção dos Parâmetros Ótimos para o classificador LSSVM

7.1 Solução Baseada na Matriz Inversa

A solução do classificador LSSVM, que consiste na obtenção de b e $\boldsymbol{\alpha}$ contida no vetor $\mathbf{x}^T = [b \;\; \boldsymbol{\alpha}^T]$, pode ser obtida resolvendo-se o sistema

$$\begin{aligned}
\mathbf{Ax} &= \mathbf{B} \\
\mathbf{A}^{-1}\mathbf{Ax} &= \mathbf{A}^{-1}\mathbf{B} \\
\mathbf{x} &= \mathbf{A}^{-1}\mathbf{B},
\end{aligned}$$

ou seja

$$\mathbf{x} = \begin{bmatrix} 0 & \mathbf{d}^T \\ \mathbf{d} & \Omega + \gamma^{-1}\mathbf{I} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \mathbf{1} \end{bmatrix},$$

bastando apenas que a matriz $\mathbf{A}$ seja não-singular.

7.2 Solução Baseada na Pseudo Inversa

Como apresentado na subseção anterior, o método mais simples de resolução é baseado no cálculo da inversa da matriz $\mathbf{A}$. Porém, no caso de uma matriz não-quadrada, faz-se necessário o uso do método da Pseudo Inversa para obtenção da solução do sistema [25,26]. Neste caso, a solução obtida equivale a considerar cada uma das colunas excluídas como possuindo multiplicador de Lagrange associado com valor zero [27]. A remoção de linhas, porém, equivale à remoção das restrições associadas ao problema de otimização. Assim, a obtenção do vetor solução $\mathbf{x}$ para uma matriz não-quadrada $\mathbf{A}$ é dada por

$$\mathbf{x} = \mathbf{A}^*\mathbf{B}, \tag{76}$$

em que

$$\mathbf{A}^* = \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T. \tag{77}$$

7.3 Solução Baseada no Método de Levenberg-Marquardt (Proposta 1)

Nesta subseção é apresentado um novo método para obtenção dos parâmetros ótimos do classificador LSSVM baseado no método de Levenberg-Marquardt (LM-LSSVM). O método Levenberg-Marquardt é um método iterativo que utiliza treinamento em lote e consiste em um aperfeiçoamento do método Gauss-Newton, que por sua vez é uma variante do método de Newton. O método de Newton utiliza informação da derivada parcial de segunda ordem da função-custo para iterativamente obter o ponto $\mathbf{x}$ que minimiza (ou maximiza) tal função. Neste contexto, além da informação de gradiente ∇, utilizada em métodos de primeira ordem, utiliza-se também de informação sobre a curvatura da superfície do erro [28].

Através da expansão de $\nabla L(\mathbf{x})$ em uma série de Taylor em torno do ponto $\mathbf{x}_0$, obtém-se

$$\nabla L(\mathbf{x}) = \nabla L(\mathbf{x}_0) + (\mathbf{x} - \mathbf{x}_0)^T \nabla L^2(\mathbf{x}_0) + \dots \tag{78}$$

Sabendo que o objetivo é encontrar o ponto ótimo (mínimo ou máximo), então faz-se

$$\nabla L(\mathbf{x}) = 0. \tag{79}$$

E aproximando $\nabla L(\mathbf{x})$ pelos termos até segunda ordem e desprezando os restantes[7], na Equação (78), bem como considerando $(\mathbf{x} - \mathbf{x}_0) = \Delta\mathbf{x}$, tem-se:

$$\begin{aligned}
\nabla L(\mathbf{x}_0) + (\mathbf{x} - \mathbf{x}_0)^T \nabla L^2(\mathbf{x}_0) &= 0 \\
(\mathbf{x} - \mathbf{x}_0)^T \nabla L^2(\mathbf{x}_0) &= -\nabla L(\mathbf{x}_0) \\
(\mathbf{x} - \mathbf{x}_0)^T [\nabla L^2(\mathbf{x}_0)][\nabla L^2(\mathbf{x}_0)]^{-1} &= -\nabla L(\mathbf{x}_0)[\nabla L^2(\mathbf{x}_0)]^{-1} \\
(\mathbf{x} - \mathbf{x}_0)^T &= -\nabla L(\mathbf{x}_0)[\nabla L^2(\mathbf{x}_0)]^{-1} \\
(\mathbf{x} - \mathbf{x}_0) &= -[\nabla L^2(\mathbf{x}_0)]^{-1} \nabla L(\mathbf{x}_0) \\
\Delta\mathbf{x} &= -[\nabla L^2(\mathbf{x}_0)]^{-1} \nabla L(\mathbf{x}_0).
\end{aligned} \tag{80}$$

De uma forma mais geral, pode-se considerar:

$$\Delta\mathbf{x} = -[\nabla L^2(\mathbf{x})]^{-1} \nabla L(\mathbf{x}). \tag{81}$$

Neste método a função-custo $L(\mathbf{x})$ é definida com base nos erros quadráticos, ou seja

$$L(\mathbf{x}) = \tfrac{1}{2} \sum_{i=1}^{n} e_i^2(\mathbf{x}) = \tfrac{1}{2} \mathbf{e}^T(\mathbf{x}) \mathbf{e}(\mathbf{x}) \tag{82}$$

em que $e_i(\mathbf{x})$ representa o erro da i-ésima entrada, enquanto $\mathbf{e}(\mathbf{x})$ representa o vetor de erro. O gradiente $\nabla L(\mathbf{x})$ e a hessiana $\nabla L^2(\mathbf{x})$ podem ser expressos com base na matriz de derivadas parciais da função-custo $\mathbf{L}(\mathbf{x})$, chamada de matriz jacobiana $\mathbf{J}(\mathbf{x})$. Assim, tem-se que

$$\nabla L(\mathbf{x}) = \mathbf{J}(\mathbf{x}) \mathbf{e}(\mathbf{x}) \tag{83}$$

$$\nabla L^2(\mathbf{x}) = \mathbf{J}^T(\mathbf{x}) \mathbf{J}(\mathbf{x}) + S(\mathbf{x}) \tag{84}$$

em que

$$S(\mathbf{x}) = \sum_{i=1}^{l} e_i(\mathbf{x}) \nabla^2 e_i(\mathbf{x}). \tag{85}$$

O cálculo da matriz hessiana aproximada pode ser extremamente complexo. Para contornar este problema, foram propostos métodos que utilizam aproximações, tais como Gauss-Newton e Levenberg-Marquardt, sendo comumente denominados por métodos Quase-Newton. Para o método de Gauss-Newton assume-se $S(\mathbf{x}) \approx 0$, e por isto a regra de atualização de Newton

$$\Delta\mathbf{x} = -[\mathbf{J}^T(\mathbf{x}) \mathbf{J}(\mathbf{x}) + S(\mathbf{x})]^{-1} \mathbf{J}(\mathbf{x}) \mathbf{e}(\mathbf{x}) \tag{86}$$

passa a ser dada por

$$\Delta\mathbf{x} = -[\mathbf{J}^T(\mathbf{x}) \mathbf{J}(\mathbf{x})]^{-1} \mathbf{J}(\mathbf{x}) \mathbf{e}(\mathbf{x}). \tag{87}$$

O problema com a Equação (87) está na obtenção da matriz hessiana aproximada $[\mathbf{J}^T(\mathbf{x}) \mathbf{J}(\mathbf{x})]$, pois esta pode não possuir inversa. Para contornar este problema, [29] propôs a adição da parcela μI à esta matriz, tal que μ é um escalar e I é a matriz identidade. Esta alteração resulta na seguinte regra de atualização:

$$\Delta\mathbf{x} = -[\mathbf{J}^T(\mathbf{x}) \mathbf{J}(\mathbf{x}) + \mu\mathbf{I}]^{-1} \mathbf{J}(\mathbf{x}) \mathbf{e}(\mathbf{x}) \tag{88}$$

Vale ressaltar que a matriz hessiana aproximada $[\mathbf{J}^T(\mathbf{x}) \mathbf{J}(\mathbf{x}) + \mu\mathbf{I}]$ sempre possui inversa [28].

[7] Nesta situação considera-se uma aproximação quadrática.

Após uma atualização, se o valor da função-custo diminuir[8], μ deve ser diminuído com o intuito de reduzir a influência do gradiente descendente. Caso contrário, quando o valor da função-custo aumenta, seguir a direção do gradiente descendente é a melhor escolha e, por isso, o valor de μ deve ser aumentado. No entanto, quando μ torna-se muito grande, a informação dada pela matriz hessiana aproximada não é útil no cálculo da atualização de $\mathbf{x}$. Para contornar esta situação adversa, [30] propôs substituir a matriz identidade pela matriz diagonal da matriz hessiana aproximada. Esta alteração resulta na seguinte regra de atualização:

$$\Delta\mathbf{x} = -[\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x}) + \mu\text{diag}[\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x})]]^{-1}\mathbf{J}(\mathbf{x})\mathbf{e}(\mathbf{x}). \tag{89}$$

Uma vez feita as considerações anteriores, o sistema de equações lineares resultante para o classificador LSSVM é dado por

$$\mathbf{A}\mathbf{x} = \mathbf{B}, \tag{90}$$

ou

$$\begin{bmatrix} 0 & \mathbf{d}^T \\ \mathbf{d} & \Omega + \gamma^{-1}\mathbf{I} \end{bmatrix} \begin{bmatrix} b \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{1} \end{bmatrix},$$

em que $\Omega_{i,j} = y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$, tal que $i, j = 1, \ldots, n$. Neste contexto, a matriz $\mathbf{B}$ pode ser considerada a saída desejada do sistema, e as matrizes $\mathbf{A}$ e $\mathbf{B}$ podem ser visualizadas como segue:

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_0 \\ \mathbf{a}_1 \\ \vdots \\ \mathbf{a}_{n+1} \end{bmatrix} \quad \text{e} \quad \mathbf{B} = \begin{bmatrix} d_0 \\ d_1 \\ \vdots \\ d_{n+1} \end{bmatrix}$$

em que $\mathbf{a}_i$ é um vetor que corresponde à i-ésima linha da matriz $\mathbf{A}$ e d_i representa a i-ésima saída escalar desejada. Ao considerar uma inicialização aleatória de $\mathbf{x}$, pode-se considerar o sistema

$$\mathbf{A}\mathbf{x} = \hat{\mathbf{Y}}(\mathbf{x}), \tag{91}$$

em que $\hat{\mathbf{Y}}$ é uma estimativa de $\mathbf{B}$. Nesta situação, pode-se calcular a função de erro da seguinte forma:

$$\begin{aligned} \mathbf{e}(\mathbf{x}) &= \mathbf{B} - \hat{\mathbf{Y}}(\mathbf{x}), \\ \mathbf{e}(\mathbf{x}) &= \mathbf{B} - \mathbf{A}\mathbf{x}. \end{aligned} \tag{92}$$

Para o método Levenberg-Maquardt deve-se considerar um lote de entradas aplicadas ao vetor $\mathbf{x}$ e, por conseqüência, um lote de erros associados. Assim, pode-se considerar cada vetor $\mathbf{a}_i$ como uma entrada para o vetor $\mathbf{x}$ e $\mathbf{y}_i$ como a sua saída associada.

A matriz Jacobiana $\mathbf{J}(\mathbf{x})$ pode ser calculada com base nas derivada parciais da função de erro, tal que

$$\mathbf{J}(\mathbf{x}) = \frac{\partial\mathbf{e}(\mathbf{x})}{\partial\mathbf{x}} = -\mathbf{A}. \tag{93}$$

E assim, a atualização do vetor $\mathbf{x}$ que contém os multiplicadores de Lagrange e o viés pode ser calculada de acordo com a seguinte expressão:

$$\begin{aligned} \mathbf{x}_{t+1} &= \mathbf{x}_t - \left[\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x}) + \mu\ \text{diag}(\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x}))\right]^{-1}\mathbf{J}^T(\mathbf{x})\mathbf{e}(\mathbf{x}), \\ &= \mathbf{x}_t + \left[\mathbf{A}^T\mathbf{A} + \mu\ \text{diag}(\mathbf{A}^T\mathbf{A})\right]^{-1}\mathbf{A}^T\mathbf{e}(\mathbf{x}), \end{aligned} \tag{94}$$

em que t representa a iteração do método e a matriz $[\mu\ \text{diag}(\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x}))]$ descreve um termo de regularização. Como dito anteriormente, a matriz $[\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x}) + \mu\ \text{diag}(\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x}))]$ é não singular [28]. O método Levenberg-Marquardt pode ser resumido no seguinte pseudo-código:

PASSO 1 - Atribuir valores de forma aleatória para o vetor $\mathbf{x}_t$, tal que $t = 0$;

PASSO 2 - Calcular o novo vetor $\mathbf{x}_{t+1}$, com base da Equação (94);

PASSO 3 - Avaliar o erro quadrático médio;

- **PASSO 3.1** - Se o erro quadrático médio aumentar então desfazer a atualização do vetor $\mathbf{x}_t$, bem como reduzir o valor de μ e retornar ao [PASSO 2];
- **PASSO 3.2** - Caso contrário, atualizar o vetor $\mathbf{x}_t$ e retornar ao [PASSO 2];

PASSO 4 - Avaliar a convergência;

- **PASSO 4.1** - Se o algoritmo convergir então finalizar e retornar o vetor $\mathbf{x}_t$;
- **PASSO 4.2** - Caso contrário, retornar ao [PASSO 2].

[8] Em um problema de minimização.

8 Simulações Computacionais

O objetivo deste primeiro grupo de experimentos computacionais é avaliar o desempenho dos classificadores SVM e LSSVM quando aplicados ao problema de diagnóstico de patologias da coluna vertebral. O conjunto de dados correspondente apresenta originalmente 3 classes, a saber: normal, hérnia de disco e espondilolistese. Mais detalhes sobre este conjunto de dados são apresentados nos Apêndices **??** e em [31]. Todavia, neste trabalho trata-se tanto do problema original com 3 classes, quanto de um problema com 2 classes, em que se agregam os padrões pertencentes às classes hérnia de disco e espondilolistese. Assim, o problema com 2 classes apresenta as classes normal (a mesma que no conjunto original) e não-normal (com patologia). O problema com 2 classes é denominado PCV-2C (Patologias da Coluna Vertebral com 2 classes), enquanto o problema com 3 classes é chamado PCV-3C (Patologias da Coluna Vertebral com 3 classes). A Tabela 2 sumariza as quantidades de padrões por classe tanto para o problema PCV-3C quanto para o problema PCV-2C.

Problema	Classe	Quantidade
PCV-3C	Normal	100
	Espondilolistese	150
	Hérnia de Disco	60
PCV-2C	Normal	100
	Não-normal	210

Tabela 2: Número de padrões por classe nos problemas PCV-3C e PCV-2C.

Nos experimentos realizados, o conjunto de dados com 310 padrões é dividido em dois conjuntos disjuntos. Nesta divisão, 80% dos dados devem compor o conjunto de treinamento, enquanto os outros 20% são utilizados para compor o conjunto de teste. O processo de separação em conjuntos de treinamento e teste é realizado com seleção de amostras de forma aleatória. Este processo é repetido 50 vezes (rodadas) para fins de avaliação da variabilidade da medida. Os classificadores SVM e LSSVM são avaliados com os *kernels*: linear, RBF e KMOD. O desempenho de cada um dos classificadores é estimado sobre o conjunto de teste. Os atributos são previamente normalizados, de tal maneira que a média tenha valor igual 0 e a variância seja igual a 1.

Para cada uma das 50 rodadas de separação em conjuntos de treinamento e teste busca-se a melhor parametrização dos classificadores através da realização de uma busca em grade (*grid search*). Esta busca é realizada no conjunto de treinamento e consiste na execução de validação cruzada de 5 partes (*5-fold cross validation*), conduzida sobre diversas combinações dos valores dos parâmetros. A busca em grade descrita ocorre em duas etapas. O esquema representando este processo de separação e otimização é apresentado na Figura 8.

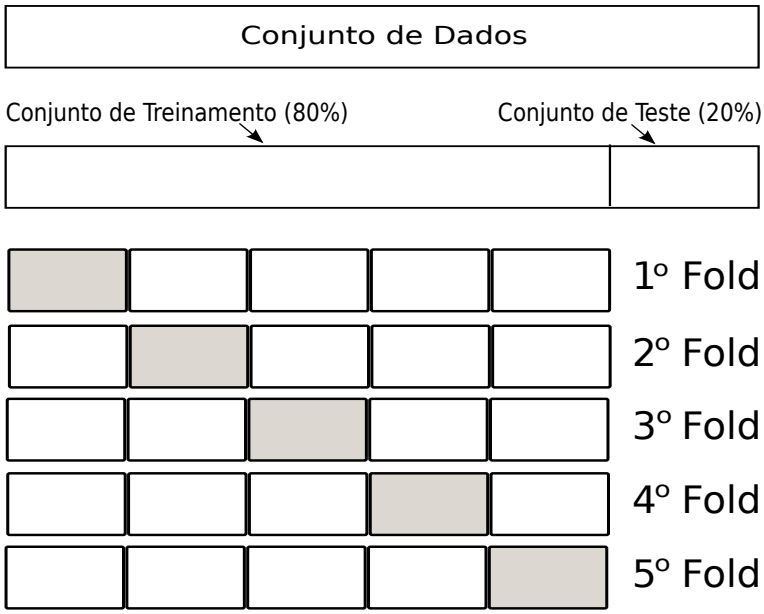


Figura 8: Esquema descrevendo um processo de separação do conjunto de dados entre conjunto de treinamento e teste. O conjunto de treinamento é ainda submetido a uma etapa de validação cruzada de 5 partes para obtenção dos parâmetros ótimos dos classificadores SVM e LSSVM.

A nomenclatura para os classificadores SVM e LSSVM, mais especificamente, utiliza a seguinte seqüência de termos classificador/algoritmo de treinamento/*kernel* e, portanto, um classificador SVM, treinado pelo algoritmo SMO com *kernel* linear, é referenciado por SMO/-LIN. Similarmente, os classificadores SVM, treinados pelo SMO com os *kernels* RBF e KMOD,

são referenciados por SVM/SMO/RBF e SVM/SMO/KMOD. Uma seqüência de termos também é aplicada para descrição dos classificadores treinados pelo algoritmo *Kernel* Adatron (ADA) e, por isto, tem-se as seguintes referências SVM/ADA/LIN, SVM/ADA/RBF e SVM/-ADA/KMOD. As diversas combinações de classificadores, métodos de treinamento e *kernel* utilizadas são apresentados na Tabela 3.

Nomenclatura	Classificador	Método Treinamento	*Kernel*
SVM/ADA/LIN	SVM	*K.* Adatron	linear
SVM/ADA/RBF	SVM	*K.* Adatron	RBF
SVM/ADA/KMOD	SVM	*K.* Adatron	KMOD
SVM/SMO/LIN	SVM	SMO	linear
SVM/SMO/RBF	SVM	SMO	RBF
SVM/SMO/KMOD	SVM	SMO	KMOD
LSSVM/MI/LIN	LSSVM	Matriz Inversa	linear
LSSVM/MI/RBF	LSSVM	Matriz Inversa	RBF
LSSVM/MI/KMOD	LSSVM	Matriz Inversa	KMOD
LSSVM/LM/LIN	LSSVM	Levenberg-Marquardt	linear
LSSVM/LM/RBF	LSSVM	Levenberg-Marquardt	RBF
LSSVM/LM/KMOD	LSSVM	Levenberg-Marquardt	KMOD

Tabela 3: Nomenclatura para os classificadores SVM e LSSVM.

Os códigos necessários para obtenção dos resultados descritos neste trabalho foram desenvolvidos em linguagem Java™. Destaca-se também que o ambiente de desenvolvimento integrado (*Integrated Development Environment* - IDE) Eclipse foi utilizado para desenvolvimento.

8.1 Resultados para o Problema Binário (PCV-2C)

Na Tabela 4 são apresentados os resultados obtidos para os classificadores SVM quando aplicados ao problema PCV-2C. Nesta tabela, mostram-se os resultados do desempenho do classificador SVM em função do algoritmo de treinamento (SMO ou Kernel Adatron) e do tipo de kernel (linear, RBF e KMOD), para uma tolerância de $0,1$. O desempenho de cada classificador é avaliado pela taxa de acerto médio no teste (acurácia), pelo desvio padrão e pelo número médio de vetores-suporte (# médio VS).

Ao se analisar a Tabela 4, pode-se perceber que o classificador SVM/SMO/KMOD obteve o melhor desempenho de classificação ($86,4\%$), seguido do classificador SVM/SMO/RBF com desempenho igual a $85,6\%$.

Classificador	*Kernel*	Acurácia	Desvio Padrão	# Médio VS
SVM/ADA	linear	$83,8$	$5,4$	$105,6$
SVM/SMO	linear	$84,2$	$4,0$	$81,2$
SVM/ADA	RBF	$84,5$	4.6	$126,0$
SVM/SMO	RBF	$85,6$	$3,7$	$120,8$
SVM/ADA	KMOD	$84,0$	$4,4$	$163,3$
SVM/SMO	KMOD	$86,4$	$4,2$	$152,0$

Tabela 4: Resultados dos classificadores SVM para o problema binário da coluna vertebral.

Na Figura 9 são apresentados os diagramas de caixa (*boxplots*) que destacam os valores mínimo e máximo, além dos percentis 25%, 50% e 75% percentil, dos valores da acurácia nas 50 rodadas de treinamento e teste, para cada um dos classificadores descritos na Tabela 4. O valor médio da acurácia para cada um dos classificadores é simbolizado por um ponto em forma de diamante no interior da caixa.

Na análise dos diagramas de caixa apresentados na Figura 9, percebe-se que o classificador com maior dispersão das taxas de acerto é o SVM/ADA/LIN; enquanto o classificador SVM/SMO/KMOD apresenta a parte central da sua caixa bastante elevada em relação aos outros classificadores. Percebe-se ainda que todos os classificadores apresentam um valor máximo de acurácia acima de 90%. O classificador SVM/SMO/KMOD apresenta maior valor médio e maior mediana (percentil 50%), seguido de perto pelo classificador SVM/SMO/RBF que, no entanto, apresenta uma dispersão ligeiramente menor que a do classificador SVM/SMO/KMOD.

Uma curva típica que descreve o processo de otimização realizado para busca do melhor conjunto de parâmetros em uma determinada rodada é apresentada na Figura 10. Esta curva descreve a acurácia em função dos parâmetros σ e C para o classificador SVM/SMO/RBF. O melhor conjunto de parâmetros encontrado para este classificador consistem em $\sigma = 3,0$ e $C = 21,0$, resultando em uma acurácia de $84,8\%$.

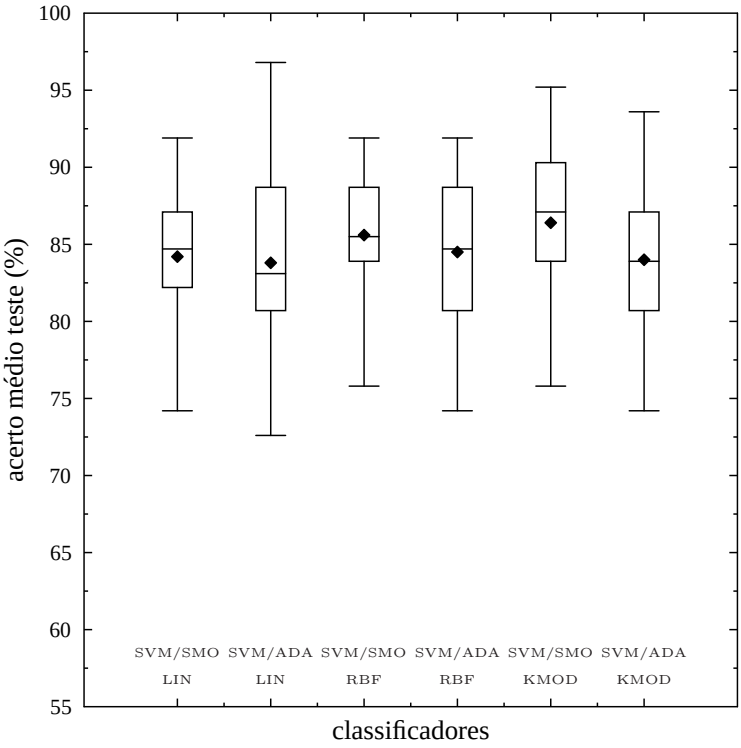


Figura 9: Diagramas de caixa contendo os valores obtidos nas 50 rodadas referentes ao problema binário com uso de classificadores SVM.

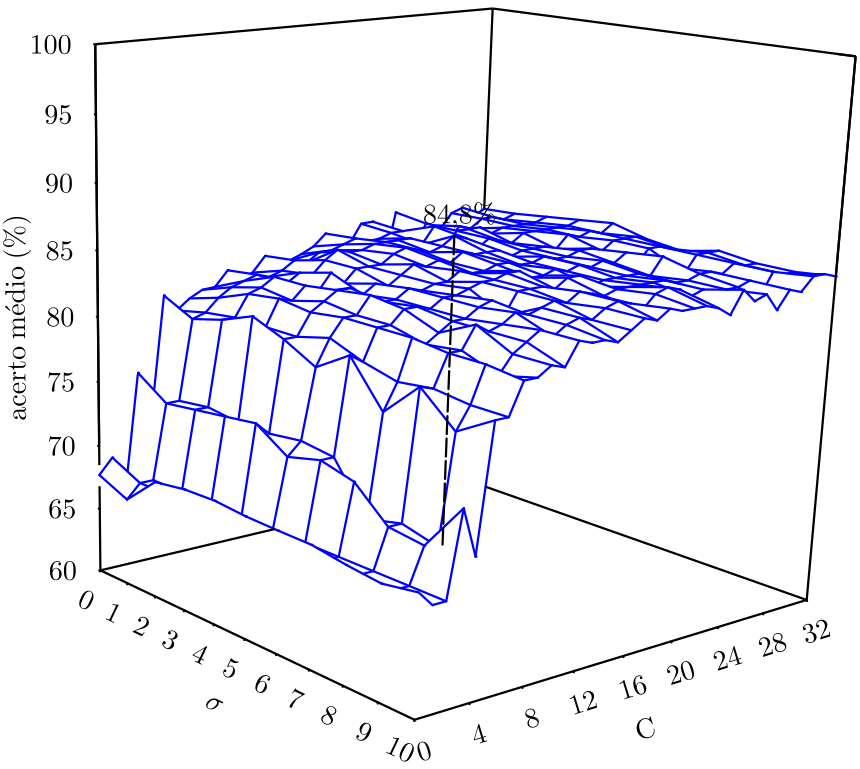


Figura 10: Superfície resultante do processo de busca em grade, via validação cruzada de 5 partes, pelo melhor conjunto de parâmetros em uma das rodadas para o classificador SMO/RBF.

De forma similar, na Tabela 5 são apresentados os resultados obtidos para o classificador LSSVM em função do método de treinamento (matriz inversa ou Levenberg-Maquardt) e do tipo de *kernel* (linear, RBF ou KMOD), quando aplicados ao problema PCV-2C. Uma busca em grade também é realizada em cada rodada para determinação dos melhores valores para γ e para os parâmetros do kernel. Para o classificador LSSVM treinado pelo método de Levenberg-Maquardt, o valor de μ é $0,001$ e o número máximo de iterações é 20. Ao se analisar os resultados, verifica-se que o algoritmo LSSVM/LM/RBF[9] apresenta a maior taxa de classificação dentre todos os classificadores LSSVM avaliados. Nota-se ainda, que o desempenho dos classificadores com *kernel* RBF e KMOD possuem valores bastante próximos. O número médio de vetores-suporte (# Médio VS), igual ao tamanho do conjunto de treinamento, confirma a falta de esparsidade da solução gerada pelo classificador LSSVM.

Classificador	*Kernel*	Acurácia	Desvio Padrão	# Médio VS
LSSVM/MI	linear	$80,4$	$5,2$	248
LSSVM/LM	linear	$81,1$	$5,2$	248
LSSVM/MI	RBF	$84,4$	$4,0$	248
LSSVM/LM	RBF	$85,0$	$4,2$	248
LSSVM/MI	KMOD	$84,3$	$4,8$	248
LSSVM/LM	KMOD	$84,3$	$4,9$	248

Tabela 5: Resultados dos classificadores LSSVM para o problema binário da coluna vertebral.

Na Figura 11 são mostrados os diagramas de caixa associados aos valores da acurácia para as 50 rodadas de treinamento e teste, para cada classificador avaliado na Tabela 5. Pela análise destes diagramas, também nota-se que há um desempenho semelhante entre os classificadores LSSVM com *kernels* RBF e KMOD. Percebe-se também que apenas estes classificadores atingem eventualmente taxas maiores que 90%. Nota-se ainda que o classificador que é treinado pelo algoritmo LM apresenta desempenho médio igual ou melhor que o classificador treinando pela matriz inversa. No entanto, a dispersão daquele classificador apresenta-se igual ou aumentada. Além disto, o classificador LSSVM/LM/RBF apresenta a parte central de sua caixa mais elevada que a dos outros classificadores, o que demonstra sua maior capacidade de generalização (o que pode ser confirmado pelo valor obtido para a acurácia do classificador).

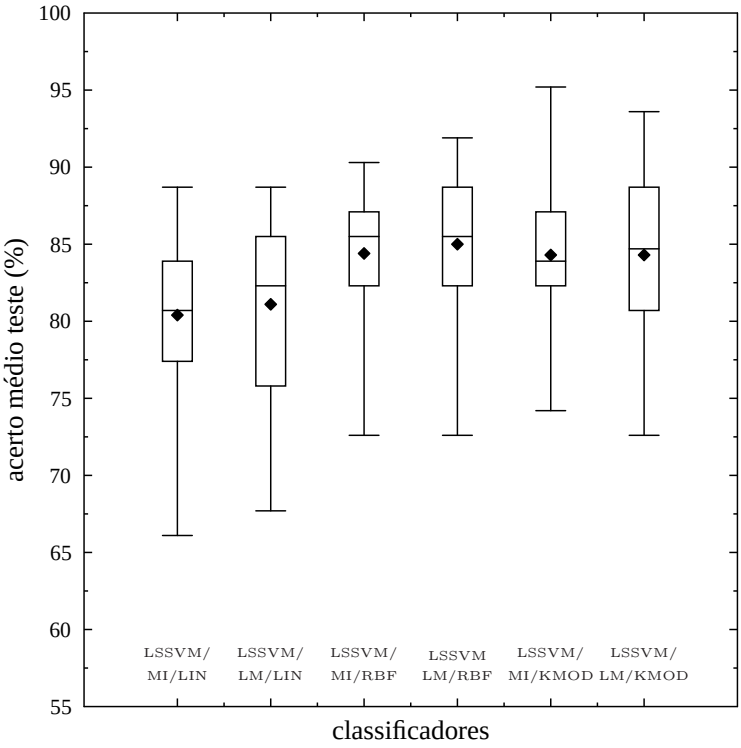


Figura 11: Diagrama de caixa contendo os valores da acurácia obtidos nas 50 rodadas referentes à aplicação do classificador LSSVM ao problema binário.

A Figura 12 contém as curvas ROC (*Receiver Operating Characteristic*) obtidas para os classificadores LSSVM/LM/RBF, SVM/SMO/KMOD e SVM/SMO/RBF. Mais detalhes sobre curvas ROC podem ser obtidos em [32]. Ao se analisar os valores

[9] O termo LSSVM/LM denota o classificador LSSVM treinado pelo algoritmo Levenberg-Maquadt, conforme proposto na Subseção 7.3.

das áreas sob as curvas (*Area Under Curve - AUC*) pode-se notar que os desempenhos dos classificadores LSSVM/LM/RBF e SVM/SMO/RBF são bastante similares, enquanto o classificador SVM/SMO/KMOD apresenta o melhor desempenho.

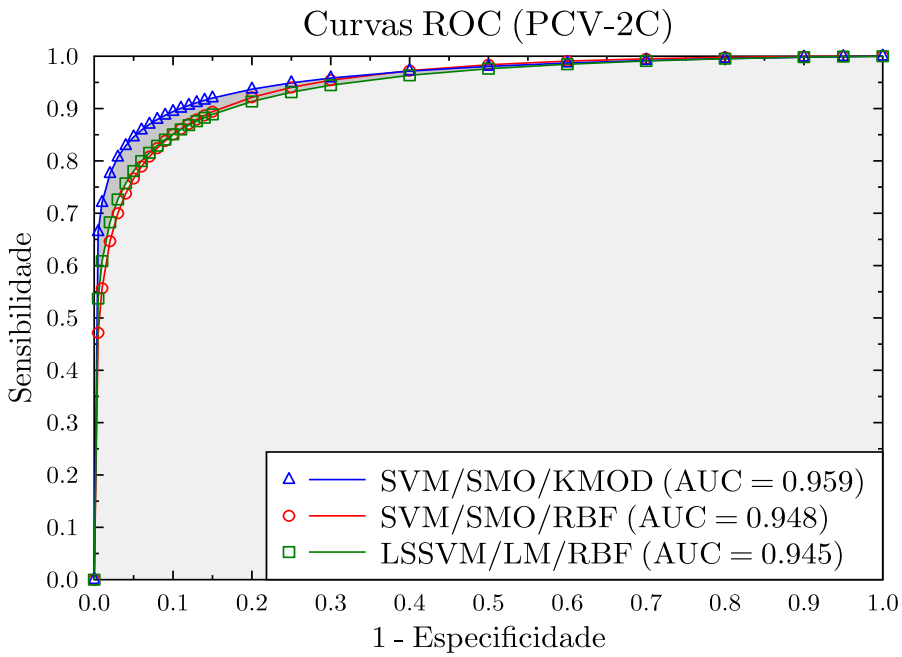


Figura 12: Curvas ROC para o problema da coluna vertebral com 2 classes, para os classificadores LSSVM/LM/RBF, SVM/SMO/KMOD e SVM/SMO/RBF, bem como os valores AUC correspondentes.

Na Figura 13, mostra-se um gráfico da acurácia em função do limiar de decisão (*thresold*) contendo três curvas, uma curva para cada um dos classificadores avaliados na Figura 12. Pode-se perceber que a curva do classificador SVM/SMO/KMOD posiciona-se quase que totalmente sobre as outras duas curvas. Ou seja, mesmo com limiares diferentes[10], o classificador SVM/SMO/KMOD possui, majoritariamente, desempenho de classificação superior ao dos demais. Este fato reforça a afirmação da superioridade do classificador SVM/SMO/KMOD em relação aos classificadores SVM/SMO/RBF e LSSVM/LM/RBF quando aplicados ao problema PCV-2C.

Finalmente, na Figura 14, é apresentada a superfície de decisão obtida para o classificador SVM/SMO/KMOD.

[10]O valor típico para o limiar é zero.

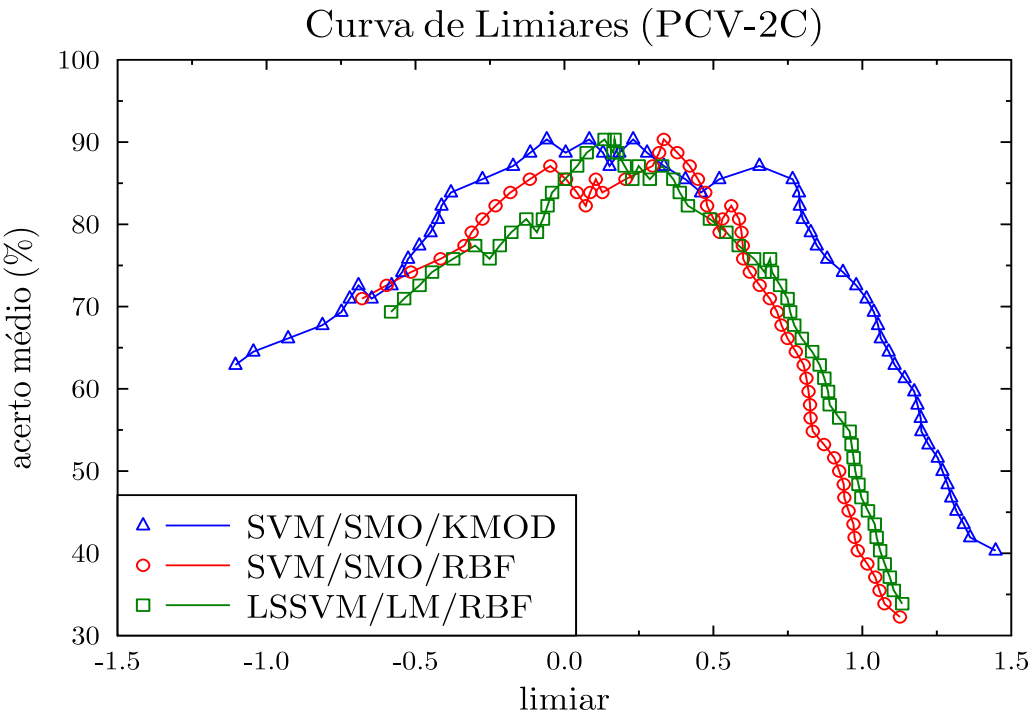


Figura 13: Gráfico com o desempenho dos classificadores LSSVM/LM/RBF, SVM/SMO/KMOD e SVM/SMO/RBF em função do limiar de decisão.

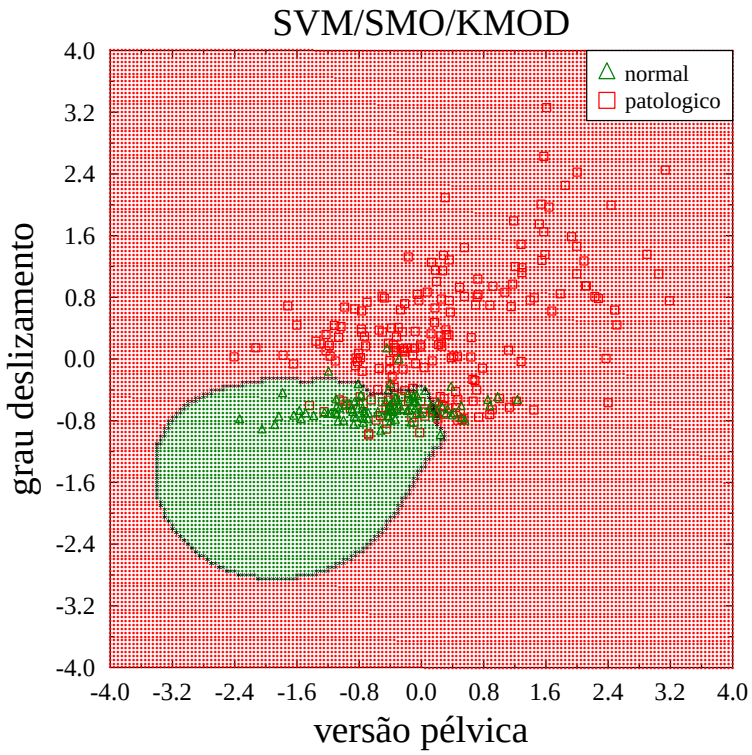


Figura 14: Superfície de decisão obtida a partir do classificador SVM/SMO/KMOD para o problema binário da coluna vertebral.

8.2 Resultados para Problema com 3 Classes (PCV-3C)

Nesta subseção tem-se a descrição dos resultados obtidos para o problema da coluna vertebral com 3 classes (PCV-3C). Neste caso adota-se a combinação de classificadores pelo método um-contra-todos [9]. A classificação usando o método um-contra-todos é realizada com base na agregação de 3 classificadores, os quais são capazes de resolver problemas binários derivados do problema original. Assim, cada classificador multiclasse agrega três classificadores SVM binários em que cada um resolve o problema de uma das classes contra as outras duas (normal contra hérnia de disco e espondilolistese, hérnia de disco contra

normal e espondilolistese e espondilolistese contra hérnia de disco e normal). A classe de um padrão é determinada de acordo com a maior saída dentre os classificadores agregados.

Para o problema PCV-3C, também é executado um processo de busca em grade a fim de se obter o melhor conjunto de parâmetros em cada uma das 50 rodadas de treinamento e teste. Vale ressaltar que neste processo de busca utiliza-se um mesmo conjunto de parâmetros para todos os três classificadores que compõem um determinado agregado que realiza a classificação multiclasse. Ou seja, na validação cruzada de 5 partes, cada ponto da grade (contendo os parâmetros) é utilizado para configurar os três classificadores.

Na Tabela 6 são mostrados os resultados obtidos para os classificadores SVM/SMO e SVM/ADA com *kernels* linear, RBF e KMOD, quando aplicados ao problema PCV-3C e para um tolerância de $0,1$. Nesta tabela são mostrados o classificador avaliado, o tipo de *kernel* utilizado, bem como a acurácia e o desvio padrão.

Infere-se, a partir da análise da Tabela 6, que o classificador SVM/SMO/KMOD apresenta melhor desempenho de classificação, com valor igual a $86,0\%$. No tocante ao diagrama de caixa, apresentado na Figura 15, pode-se notar que a parte central do classificador SVM/SMO/KMOD encontra-se mais elevada e compacta que as partes centrais dos outros classificadores, bem como apresenta maior desempenho médio o que demonstra a robustez de seu algoritmo de aprendizagem. Nota-se ainda que o classificador SVM/SMO/LIN é o único que consegue atingir desempenho superior à 95%, no entanto, este possui a maior dispersão entre os classificadores avaliados.

Classificador	*Kernel*	Acurácia	Desvio Padrão
SVM/SMO	Linear	$85,1$	$4,6$
SVM/ADA	Linear	$85,3$	$4,5$
SVM/SMO	RBF	$85,3$	$3,4$
SVM/ADA	RBF	$83,9$	$3,9$
SVM/SMO	KMOD	$86,0$	$4,0$
SVM/ADA	KMOD	$84,1$	$4,2$

Tabela 6: Resultados do classificador SVM para o problema da coluna vertebral com 3 classes.

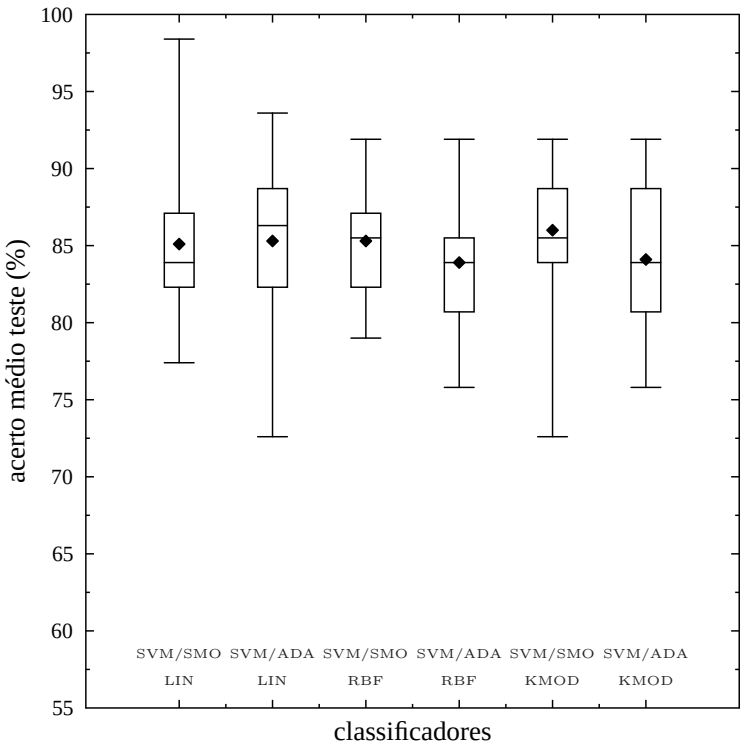


Figura 15: Diagramas de caixa com os resultados obtidos para os classificadores SVM quando aplicados ao problema PCV-3C.

A Tabela 7 traz os resultados obtidos para o problema PCV-3C no que se refere aos classificadores LSSVM. De um modo geral, os resultados apresentados nesta tabela são bem inferiores aos descritos na Tabela 6. Uma justificativa para esta situação pode residir no fato de que o ajuste dos parâmetros dos classificadores LSSVM é bastante sensível a pequenas variações. Nestes classificadores, muitas vezes, um ajuste fino faz-se necessário. Ou seja, como há vários classificadores agregados para a execução da classificação em um problema multiclasse, um conjunto de parâmetros que é adequado para um classificador binário, pode

não ser tão adequado para os outros dois classificadores.

Classificador	*Kernel*	Acurácia	Desvio Padrão
LSSVM/MI	Linear	$80,8$	$5,1$
LSSVM/LM	Linear	$80,7$	$4,7$
LSSVM/MI	RBF	$81,1$	$3,8$
LSSVM/LM	RBF	$81,9$	$5,6$
LSSVM/MI	KMOD	$82,7$	$4,1$
LSSVM/LM	KMOD	$82,5$	$3,5$

Tabela 7: Resultados dos classificadores LSSVM para o problema da coluna vertebral com 3 classes.

Na Figura 16 são mostrados os diagramas de caixa referentes aos classificadores e resultados apresentados na Tabela 7. Percebe-se que os desempenhos médios obtidos para os classificadores LSSVM/MI e LSSVM/LM são inferiores aos obtidos para os classificadores SVM/SMO e SVM/ADA. Os melhores resultados entre os classificadores LSSVM/MI e LSSVM/LM são os obtidos para os classificadores LSSVM/MI/KMOD e LSSVM/LM/KMOD. Percebe-se ainda que o classificador LS-SVM/LM/RBF apresenta o maior acerto e a maior dispersão dentre os classificadores avaliados.

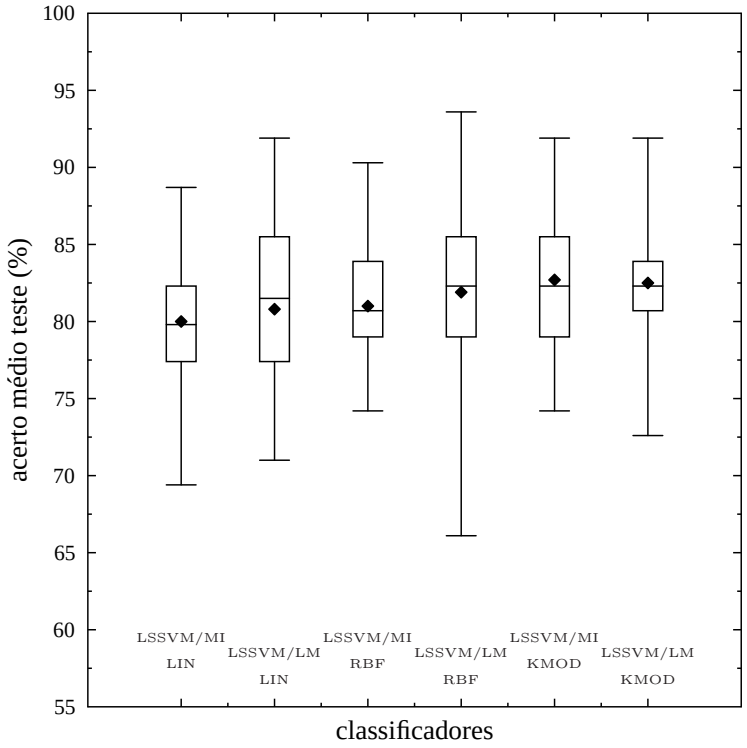


Figura 16: Diagramas de caixa com os resultados obtidos para os classificadores LSSVM para o problema PCV-3C.

Para fins de comparação, na Tabela 8 são apresentados diversos resultados obtidos para a aplicação dos classificadores k-NN, *Naive Bayes*, GRNN, MLP e SVM/SMO/KMOD ao problema PCV-3C. Detalhes sobre os classificadores k-NN, *Naive Bayes*, GRNN, MLP podem ser obtidos em [31]. Os parâmetros dos classificadores são os seguintes: k-NN ($k = 7$), GRNN ($\sigma = 2,0$), MLP (6,12,3)[11] com função de ativação sigmóide logística, treinada por 1000 épocas e usando taxa de aprendizagem igual a 0,05, SVM/SMO/KMOD com parâmetros de *kernel* $\sigma = 2,5$, $\gamma = 8,5$ e parâmetro de regularização $C = 2,5$, e a margem de tolerância com valor igual a $0,1$.

Na Tabela 8 analisa-se a evolução do desempenho do classificador em função do tamanho do conjunto de dados usado para treinamento dos classificadores. Ou seja, calcula-se a acurácia e o desvio padrão no conjunto de teste quando se utiliza 25% do conjunto de dados para treinamento e 75% para teste. Em seguida, faz-se o mesmo para 40% e 60% dos dados nos conjuntos de treinamento e teste. Verifica-se também o desempenho de cada classificador para as proporções 60%-40% e 80%-20% para os conjuntos de treinamento-teste. Para cada uma destas combinações são realizadas 50 rodadas.

Os gráficos relacionados aos resultados contidos na Tabela 8 são vistos na Figura 17. Nota-se que os classificadores SVM/SMO/KMOD e MLP apresentam maior acurácia que os classificadores k-NN, *Naive Bayes* e GRNN. O classificador MLP

[11] Utiliza-se a notação compacta MLP(p,q,m), em que p é a dimensão do vetor de entrada, q é o número de neurônios ocultos e m é o número de neurônios de saída.

Classificador	**Tamanho do Conjunto de Treinamento**			
	25%	40%	60%	80%
k-NN	$76,0 \pm 2,9$	$77,3 \pm 2,7$	$77,8 \pm 3,1$	$78,3 \pm 4,3$
Naive Bayes	$78.8 \pm 2,6$	$78.9 \pm 2,5$	$80,2 \pm 3,1$	$80,3 \pm 4,7$
GRNN	$71,9 \pm 4,1$	$73,9 \pm 3,1$	$75,0 \pm 4,2$	$75,0 \pm 5,3$
MLP	$82,9 \pm 2,7$	$82,8 \pm 2,4$	$83,8 \pm 3,0$	$83,7 \pm 4,4$
SVM/SMO/KMOD	$80,0 \pm 2,8$	$82,1 \pm 2,4$	$83,7 \pm 2,5$	$85,0 \pm 4,6$

Tabela 8: Resultados obtidos para diversos classificadores aplicados ao problema PCV-3C. Pode-se observar os acertos médios obtidos para diferentes tamanhos do conjunto de treinamento.

apresenta maior desempenho médio que o classificador SVM/SMO/KMOD na faixa de 25%-40%, para 60% do conjunto de treinamento as acurácias são semelhantes, para 80% do conjunto de dados o classificador SVM/SMO/KMOD apresenta melhor desempenho.

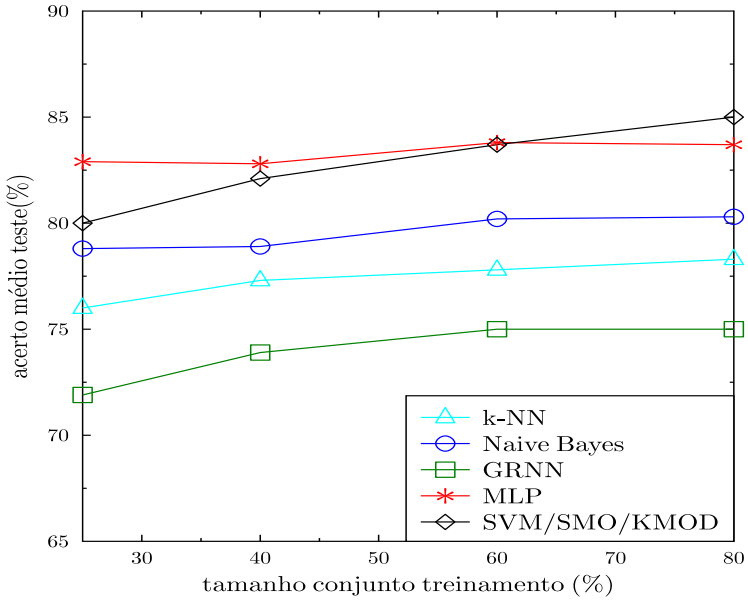


Figura 17: Acurácia de diversos classificadores em função do tamanho do conjunto de treinamento.

Na Figura 18, pode-se visualizar as superfícies de decisão geradas pelos classificadores MLP e SVM/SMO/KMOD para o par de atributos com maior capacidade de discriminação. Nota-se, ao se analisar a figura, que há uma significativa sobreposição entre os dados pertencentes às classes normal e hérnia de disco.

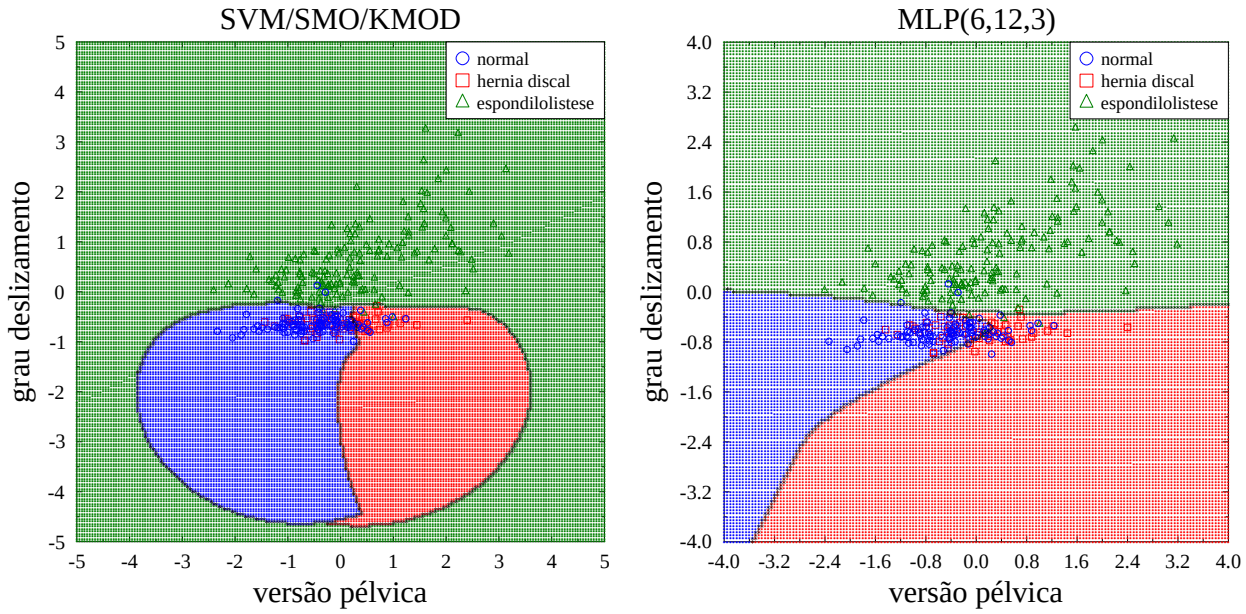


Figura 18: (a) Superfície de decisão do classificador SVM/SMO/KMOD[$\gamma = 8,5; \sigma = 2,0$ e $C = 2,5$]. (b) Superfície de decisão da rede MLP(6,12,3) treinada por 1000 épocas com taxa de aprendizado igual a $0,05$.

9 Conclusão

Neste artigo foram avaliados os desempenhos de classificadores do tipo SVM e LSSVM quando aplicados ao problema de diagnóstico de patologias da coluna vertebral, tanto para a versão com duas classes, quanto para a versão com 3 classes. Uma nova técnica para treinamento de classificadores LSSVM é proposta, com base no método Levenberg-Maquardt. Esta proposta apresentou resultados similares aos dos classificadores baseados nos algoritmos SMO e Adatron para o problema binário (PCV-2C).

De uma forma geral, pode-se considerar o classificador SVM/SMO/KMOD como sendo o melhor classificador para o problema da coluna PCV-2C, pois o mesmo obteve melhor resultado em termos de acurácia, análise de diagramas de caixa e análise de curvas ROC. O classificador SVM/SMO/KMOD quando aplicado ao problema PCV-3C também apresenta-se mais adequado do que os classificadores k-NN, *Naive Bayes*, GRNN e MLP, bem como apresenta-se melhor do que a grande maioria das máquinas de vetores-suporte avaliadas. Logo, o classificador SVM/SMO/KMOD deverá compor o módulo de diagnóstico da plataforma SINPATCO.

Em situações em que se exige um melhor desempenho em termos de custo computacional pode-se considerar o uso do classificador SVM/SMO/LIN, pois este possui um pouco mais do que 80 vetores-suporte em média, cerca de $2/3$ da quantidade obtida para os outros classificadores (que possuem RBF ou KMOD) quando se considera o problema PCV-2C.

Os resultados obtidos para o conjunto PCV-3C pelo classificador LSSVM/MI ou pelo LSSVM/LM não foram tão satisfatórios quanto os obtidos pelo classificador SVM/SMO e SVM/ADA. Isto pode ser justificado pela necessidade de um ajuste fino dos parâmetros dos classificadores LSSVM agregados quando se objetiva a resolução de um problema multiclasse. Ou seja, um determinado conjunto de parâmetros que se faz adequado para um classificador LSSVM binário pode não ser tão adequado para os outros dois classificadores. Percebe-se que os classificadores LSSVM são bastante sensíveis a pequenas variações dos parâmetros de treinamento, diferentemente do observado para os classificadores SVM.

REFERÊNCIAS

[1] V. Vapnik and A. Lerner. “Pattern Recognition using Generalized Portrait Method”. *Automation and Remote Control*, vol. 24, 1963.

[2] V. N. Vapnik and A. Y. Chervonenkis. “A note on one class of perceptrons”. *Automation and Remote Control*, vol. 25, 1964.

[3] B. E. Boser, I. Guyon and V. Vapnik. “A training algorithm for optimal margin classifiers”. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, pp. 144–152. ACM Press, 1992.

[4] C. Cortes and V. Vapnik. “Support Vector Networks”. *MLEARN*, vol. 20, pp. 273–297, 1995.

[5] R. Burbidge and B. Buxton. “B.F.: An introduction to support vector machines for data mining”. In *Keynote Papers, Young OR12, University of Nottingham, Operational Research Society, Operational Research Society*, pp. 3–15, 2001.

[6] Burges and Christopher. "A Tutorial on Support Vector Machines for Pattern Recognition". *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[7] A. Smola and B. Schölkopf. "A Tutorial on Support Vector Regression". Technical Report NeuroCOLT2 Technical Report Series, NC2-TR-1998-030, Royal Holloway College, University of London, UK, October 1998.

[8] K. Crammer and Y. Singer. "On the Algorithmic Implementation of Multiclass Kernel-based Vector Machines". *IEEE Transactions on Latin America*, vol. 2, no. 12, pp. 265–292, Dec. 2001.

[9] K. Duan and S. S. Keerthi. "Which is the best multiclass SVM method? An empirical study". In *Proceedings of the Sixth International Workshop on Multiple Classifier Systems*, pp. 278–285, 2005.

[10] J. C. Platt, N. Cristianini and J. Shawe-taylor. "Large Margin DAGs for Multiclass Classification". In *Advances in Neural Information Processing Systems*, pp. 547–553. MIT Press, 2000.

[11] G. B. T. Dietterich. "Solving multiclass problem via error-correcting output code". *Journal of Artificial Intelligence Research*, vol. 2, pp. 263–286, 1995.

[12] V. N. Vapnik. *Estimation of Dependences Based on Empirical Data*. Springer Series in Statistics. 1982.

[13] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag., New York, 1995.

[14] J. Mercer. "Functions of positive and negative types and their connection with the theory of integral equations". In *Transactions of the London Philosophical Society*, volume A of *209*, 1909.

[15] J. Valyon. "Extended LS-SVM For System Modeling". Ph.D. thesis, Budapest University of Technology and Economics, 2007.

[16] N. E. Ayat, M. Cheriet and C. Y. Suen. "Kmod - a two parameter SVM kernel for pattern recognition". In *In Proceedings of the 16th International Conference on Pattern Recognition*, volume 3, pp. 331–334, 2002.

[17] L. Kaufman. "Solving the quadratic programming problem arising in support vector classification". In *Advances in kernel methods*, pp. 147–167. MIT Press, Cambridge, MA, USA, 1999.

[18] J. C. Platt. "Fast training of support vector machines using sequential minimal optimization". In *Advances in Kernel Methods - Support Vector Learning*. MIT Press, Cambridge, MA, USA, 1999.

[19] J. K. Anlauf and M. Biehl. "The AdaTron: An Adaptive Perceptron Algorithm". *EPL (Europhysics Letters)*, vol. 10, no. 7, pp. 687, 1989.

[20] C. Campbel and N. Cristianini. "Simple Learning Algorithms for Training Support Vector Machines". Technical report, Technical Report, Dept. of Engineering Mathematics, University of Bristol, UK, 1998.

[21] T. Frieß, N. Cristianini and C. Campbell. "The Kernel-Adatron Algorithm: a Fast and Simple Learning Procedure for Support Vector Machines". In *Machine Learning: Proceedings of the Fifteenth International Conference*. Morgan Kaufmann Publishers, 1998.

[22] J. A. K. Suykens and J. Vandewalle. "Least squares support vector machine classifiers". *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, 1999.

[23] J. A. K. Suykens and J. Vandewalle. "Multiclass least squares support vector machines". In *IJCNN'99 International Joint Conference on Neural Networks*, Washington, DC, 1999.

[24] R. Fletcher. *Practical Methods of Optimization*. John Wiley & Sons, New York, second edition, 1987.

[25] E. H. Moore. "On the Reciprocal of the General Algebraic Matrix". *Bull. Amer. Math. Soc.*, vol. 26, pp. 394–395, 1920.

[26] R. Penrose. "A Generalized Inverse for Matrices". *Proceedings of the Cambridge Philosophical Society*, vol. 51, pp. 406–413, 1955.

[27] Y. Lee and O. Mangasarian. "RSVM: Reduced support vector machines". In *SIAM International Conference on Data Mining*, pp. 00–07, 2001.

[28] A. Ranganathan. "The Levenberg-Marquardt Algorithm", 2004.

[29] K. Levenberg. "A Method for the Solution of Certain Problems in Least Squares". *Quart. Appl. Math.*, vol. 2, pp. 164–168, 1944.

[30] D. Marquardt. "An Algorithm for Least-Squares Estimation of Nonlinear Parameters". *SIAM J. Appl. Math.*, vol. 11, pp. 431–441, 1963.

[31] A. R. Rocha-Neto. "SINPATCO - Sistema Iinteligente para o Diagnóstico de Patologias da Coluna Vertebral". Master's thesis, Universidade Federal do Ceará, 2006.

[32] R. Prati, G. Batista and M. Monard. "Evaluating Classifiers Using ROC Curves". *IEEE Latin America Transactions*, vol. 6, no. 2, pp. 215 –222, june 2008.