



UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA DE TELEINFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE TELEINFORMÁTICA

AJALMAR RÊGO DA ROCHA NETO

**SINPATCO II: NOVAS ESTRATÉGIAS DE APRENDIZADO DE MÁQUINA PARA
CLASSIFICAÇÃO DE PATOLOGIAS DA COLUNA VERTEBRAL**

Fortaleza - Ceará
13 de Setembro de 2011

AJALMAR RÊGO DA ROCHA NETO

**SINPATCO II: NOVAS ESTRATÉGIAS DE APRENDIZADO DE MÁQUINA PARA
CLASSIFICAÇÃO DE PATOLOGIAS DA COLUNA VERTEBRAL**

Tese submetida ao Programa de Pós-Graduação em Engenharia de Teleinformática, da Universidade Federal do Ceará, como requisito parcial para obtenção do grau de Doutor em Engenharia de Teleinformática.

Área de Concentração: Sinais e Sistemas.

Orientador: Prof. Dr. Guilherme de Alencar Barreto

Fortaleza - Ceará
13 de Setembro de 2011

Este trabalho é dedicado à minha mãe, **Angéla Maria Soares de Moura**, pelo exemplo de vida, dedicação aos filhos e por ter mostrado à família que podemos vencer os obstáculos da vida. Dedico também à minha esposa, **Cláudia Roberta Oliveira de Farias Rocha**, por ter me acompanhado em grande parte desta trajetória, bem como dedico às minhas irmãs, **Atslands Rêgo da Rocha e Andressa Rêgo da Rocha**, por sempre estarmos juntos, em qualquer situação. Por fim, agradeço à **Maria dos Remédios** (em memória), pelo apoio e incentivo aos meus estudos.

Agradecimentos

Este momento é especial pois me permite agradecer aos colaboradores deste trabalho. Foram horas de estudo e dedicação e durante todo esse tempo muitas pessoas amigas e outras até então desconhecidas colaboraram para que eu pudesse chegar neste momento. Não poderia deixar de agradecer a estas pessoas e pedir perdão por alguém que eu tenha esquecido de mencionar. Gostaria de agradecer:

- a Deus que me orienta em todos os momentos;
- aos meus pais pela educação, pelos esforços para me oferecer o melhor e pelo amor incondicional;
- aos meus irmãos que sempre me apóiam;
- ao professor Guilherme Barreto por ser essencial na elaboração deste trabalho;
- ao médico ortopedista Henrique da Mota pela grande contribuição e auxílio na construção e correção deste trabalho;
- ao professor Jaime dos Santos Cardoso pela orientação durante a realização do doutorado sanduíche;
- ao doutorando Ricardo Sousa pela relação de amizade e trabalho conjunto durante a realização do doutorado sanduíche;
- aos professores Paulo César Cortez, Juvêncio Santos Nobre, Carlos Eduardo Pedreira, Antônio de Pádua Braga pelas sugestões e correções;
- aos demais professores da UFC, secretários da pós-graduação e a todos os funcionários do departamento pela colaboração durante o decorrer do doutorado;
- a todos que de alguma forma contribuíram para a realização deste trabalho.

Definir é limitar.

Oscar Wilde

Resumo

Esta tese tem por objetivo principal avaliar o estado da arte em algoritmos de classificação de padrões, bem como propor novas estratégias, que conduzam a uma melhoria do desempenho do módulo de diagnóstico do Sistema Inteligente para Diagnóstico de Patologias da Coluna Vertebral (SINPATCO). A plataforma SINPATCO é um ambiente computacional voltado para a classificação de pacientes em três categorias: normal, com hérnia de disco ou com espondilolistese. Pode-se ainda configurar a plataforma SINPATCO para fundir as duas classes de patologias (hérnia e espondilolistese) em uma única classe, abordando a tarefa como um problema de classificação binária. As diversas estratégias de classificação discutidas nesta tese, sejam clássicas ou propostas, são avaliadas com o objetivo último de fazerem parte do processo de tomada de decisão da segunda geração da plataforma SINPATCO, dando origem à plataforma SINPATCO II.

Dentre as estratégias consideradas estado da arte em classificação de padrões, são de particular interesse para esta tese as seguintes abordagens: (i) classificação baseada em máquinas de vetores-suporte (SVM) e *Least Squares SVM* (LSSVM), (ii) classificação com opção de rejeição, e (iii) classificação baseada em comitês. Para cada uma destas três estratégias, além de avaliar a aplicação de métodos clássicos ao problema de interesse para esta tese, são introduzidas cinco (5) novas propostas com o intuito de contribuir também para o desenvolvimento teórico da área de aprendizado de máquinas, de um modo geral.

Com relação à abordagem baseada em classificadores SVM e LSSVM, são apresentadas duas novas propostas, chamadas de Propostas 1 e 2. A Proposta 1 corresponde a um novo algoritmo de treinamento do classificador LSSVM baseado no método Levenberg-Marquardt, já a Proposta 2 envolve o uso do mapa auto-organizável de Kohonen na geração de conjuntos reduzidos (*reduced sets*) para treinamento de classificadores SVM e LSSVM. Uma generalização da Proposta 2 é apresentada, a fim de poder ser usada com qualquer algoritmo de quantização vetorial, tais como *K*-médias, kernel *K*-médias e *Growing Neural Gas*. Com relação à abordagem baseada em classificação com opção de rejeição, são introduzidas também duas novas propostas, chamadas de Propostas 3 e 4, que usam o mapa de Kohonen como classificador (supervisionado) de padrões. A Proposta 3 envolve o uso de apenas um mapa de Kohonen, enquanto a Proposta 4 utiliza dois mapas. Finalmente, com relação à abordagem por comitê de classificadores, introduz-se uma nova idéia, chamada de Proposta 5, que consiste no uso de classificadores SVM, preferencialmente em comitês homogêneos, com um kernel não-usual, conhecido como KMOD (*Kernel With Moderate Decreasing*). Uma série abrangente de experimentos computacionais com vários classificadores clássicos, arranjados em comitês homogêneos/heterogêneos, e usando diferentes funções kernels (no caso do classificador SVM), atestam o desempenho superior da Proposta 5.

Palavras-Chave: *Patologias da Coluna Vertebral, Máquinas de Vetores-Suporte, Conjuntos Reduzidos em Máquinas de Vetores-Suporte, Opposite Maps, Classificação com Opção de Rejeição e Comitês de Classificadores.*

Abstract

This thesis aims at evaluating the state of the art in pattern classification algorithms, as well as proposing novel ones, that result in improvements for the diagnostic module of the System for Intelligent Diagnosis of Pathologies of the Vertebral Column (SINPATCO). The SINPATCO platform is a computational environment engaged in the classification of orthopaedic patients into three categories: normal patient, with disk hernia, or with spondylolisthesis. It is possible to configure the platform in order to merge the two classes of pathologies (i.e. disk hernia and spondylolisthesis) into a single class, redefining the task as a binary classification problem. The several classification strategies discussed in this thesis, be they classic or novel ones, are evaluated with the ultimate goal of becoming part of the process decision making of the second generation of the SINPATCO platform, to be called SINPATCO II.

Among the state-of-the-art classification strategies, the following ones are of particular interest for this thesis: (i) classification based on support vector machines (SVM) and least-squares SVM, (ii) classification with rejection option, and (iii) classification based on ensembles of classifiers. For each one of them, in addition to the application of standard classification methods to the problem of interest, five new classification strategies are introduced in order to also contribute the theoretic development of the Machine Learning field as a whole.

With respect to the SVM/LSSVM-based approach, two new proposals are presented, to be called Proposals 1 and 2. The Proposal 1 corresponds to a new learning algorithm for LSSVM classifiers based on the Levenberg-Marquardt method, while the Proposal 2 involves the use of the Kohonen map to generate reduced sets for training SVM/LSSVM classifiers. A generalized framework for the Proposal 2 is then introduced in order to allow the use of any vector quantization algorithm, such as the K -means, kernel K -means and the Growing Neural Gas. With respect to the approach based on rejection option, two new strategies are also proposed, to be called Proposals 3 and 4, both of them using the Kohonen map as a (supervised) pattern classifier. The Proposal 3 involves the use of a single Kohonen map, while the Proposal 4 uses two maps. Finally, with respect to the ensemble-based approach, a new idea is introduced, to be called Proposal 5, which consists in the use SVM classifiers, preferably in a homogeneous ensemble, with a non-usual kernel known as KMOD (*Kernel With Moderate Decreasing*). Comprehensive computer experiments with several standard classifiers arranged homogeneous/heterogeneous ensembles and with different kernel functions (for SVM classifiers) attest the superior performance of the Proposal 5.

Keywords: *Pathologies of the Vertebral Column, Support Vector Machines, Reduced Set, Opposite Maps, Classification with Rejection Option and Ensembles.*

Sumário

Lista de Figuras	xii
Lista de Tabelas	xvii
Lista de Símbolos	xviii
1 Introdução	1
1.1 Motivação para a Tese	2
1.2 Descrição do Problema	4
1.2.1 Atributos Biomecânicos	6
1.3 Objetivos Geral e Específicos	7
1.3.1 Objetivo Geral	7
1.3.2 Objetivos Específicos	7
1.4 Produção Científica	8
1.5 Estrutura da Tese	9
1.5.1 Metodologia de Organização	9
1.5.2 Organização da Tese	10
2 Classificadores SVM e LSSVM	12
2.1 Introdução	12
2.2 Definições e Conceitos Preliminares	13
2.3 Fundamentos Teóricos do Classificador SVM	19
2.3.1 Classificador SVM com Margem Rígida	19

2.3.2	Classificador SVM com Margem Flexível	21
2.4	O Truque do <i>Kernel</i>	25
2.4.1	Kernel with MODerate decreasing (KMOD)	28
2.5	Obtenção dos Parâmetros Ótimos para o Classificador SVM	29
2.5.1	Solução baseada em Programação Quadrática	29
2.5.2	Solução baseada no Algoritmo SMO	30
2.5.3	Solução baseada no Kernel Adatron	31
2.6	Fundamentos Teóricos do Classificador LSSVM	33
2.7	Obtenção dos Parâmetros Ótimos para o classificador LSSVM	35
2.7.1	Solução Baseada na Matriz Inversa	35
2.7.2	Solução Baseada na Pseudo Inversa	35
2.7.3	Solução Baseada no Método de Levenberg-Marquardt (Proposta 1) . .	36
2.8	Simulações Computacionais	39
2.8.1	Resultados para o Problema Binário (PCV-2C)	41
2.8.2	Resultados para Problema com 3 Classes (PCV-3C)	47
2.9	Conclusão	52
3	Técnicas para Obtenção de Conjuntos Reduzidos em SVM e LSSVM	53
3.1	Introdução	53
3.2	Análise dos Vetores-Suporte	55
3.2.1	Vetores-Suporte em Classificadores SVM	55
3.2.2	Vetores-Suporte em Classificadores LSSVM	56
3.3	Métodos para obtenção de Conjuntos Reduzidos	58
3.3.1	<i>Reduced Set Method</i> (RSM)	59
3.3.2	<i>Reduced SVM</i> (RSVM)	60
3.3.3	<i>Pruning</i> LSSVM	60
3.3.4	IP-LSSVM	61

3.3.5	GNG-SVM	62
3.4	Opposite Maps (Proposta 2)	63
3.4.1	<i>Opposite Maps:</i> Passo-a-Passo	65
3.4.2	Classificador OM-SVM	69
3.4.3	Classificador OM-LSSVM	69
3.5	<i>Generalized Opposite Maps</i>	69
3.6	Simulações Computacionais	71
3.6.1	Resultados para Conjuntos de Dados Artificiais	71
3.6.2	Resultados para o problema PCV-2C	75
3.7	Conclusão	80
4	Classificação com Opção de Rejeição	82
4.1	Introdução	82
4.2	Fundamentação Teórica	83
4.2.1	Regra de Decisão Ótima	84
4.3	Abordagens para Classificação Binária com Opção de Rejeição	88
4.3.1	Um Classificador Padrão	88
4.3.2	Dois Classificadores Independentes	88
4.3.3	Classificador com Opção de Rejeição Embutida	91
4.4	Novas Propostas para Classificação com Opção de Rejeição	96
4.4.1	Classificador SOM Padrão (Proposta 3)	97
4.4.2	Dois Classificadores SOM Independentes (Proposta 4)	98
4.5	Simulações Computacionais	99
4.5.1	Resultados para classificadores SVM	99
4.5.2	Resultados das Estratégias que se baseiam em redes MLP e SOM	103
4.5.3	Análise Comparativa das Estratégias de Rejeição	106
4.6	Conclusão	107

5 Comitês de Classificadores	109
5.1 Introdução	109
5.2 Fundamentação Teórica	111
5.2.1 Ambigüidade Decomposicional	112
5.2.2 Decomposição Viés/Variância/Covariância	112
5.2.3 Correlação do Erro de Classificação	113
5.3 Projeto de Comitês	114
5.3.1 Geração de Componentes-Base	114
5.3.2 Seleção de Componentes-Base	117
5.3.3 Combinação de Componentes	118
5.4 Simulações Computacionais	119
5.4.1 Classificadores Considerados Isoladamente	121
5.4.2 Comitês de Classificadores (Proposta 5)	123
5.5 Conclusão	129
6 Conclusões e Trabalhos Futuros	130
6.1 Conclusões	130
6.2 Resumo das Contribuições Científicas	131
6.3 Trabalhos Futuros	132
Apêndice A – Análise Preliminar dos Dados	133
A.1 Diagramas de Caixa dos Atributos Biomecânicos	133
A.2 Gráficos de Dispersão e Curvas de Nível	134
A.3 Análise de Componentes Principais (PCA)	135
A.4 Análise de Generalização por Combinação dos Atributos	136
A.5 Mapeamento Quadrático	139
Apêndice B – Rede GNG, Algoritmos K-Médias e Kernel K-Médias	141

B.1	K-Médias	141
B.2	Kernel K-Médias	142
B.3	<i>Growing Neural Gas</i> (GNG)	143
Referências Bibliográficas		146

Lista de Figuras

1.1	Histórico de desenvolvimento das plataformas SINPATCO I e II.	5
1.2	Descrição dos atributos biomecânicos.	7
2.1	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.	14
2.2	Um problema linearmente separável e hiperplanos solução do problema.	15
2.3	Interpretação geométrica da distância de um padrão \mathbf{x} ao hiperplano ótimo. . .	18
2.4	Exemplo de problema binário em que um hiperplano não resolve de forma satisfatória, mesmo considerando a possibilidade de alguns erros.	25
2.5	Espaço de entrada no \mathbb{R}^2 , Espaço de Características no \mathbb{R}^3 e Espaço <i>Kernel</i> . . .	27
2.6	Comportamento dos <i>kernels</i> RBF e KMOD.	29
2.7	<i>Kernel</i> Adatron representado de forma similar às redes neurais artificiais. . .	31
2.8	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.	41
2.9	Diagramas de caixa contendo os valores obtidos nas 50 rodadas referentes ao problema binário com uso de classificadores SVM.	43
2.10	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 SVM/SMO/RBF.	43
2.11	Diagrama de caixa contendo os valores da acurácia obtidos nas 50 rodadas referentes à aplicação do classificador LSSVM ao problema binário.	45

2.12	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.	45
2.13	Gráfico com o desempenho dos classificadores LSSVM/LM/RBF, SVM/SMO/KMOD e SVM/SMO/RBF em função do limiar de decisão.	46
2.14	Superfície de decisão obtida a partir do classificador SVM/SMO/KMOD para o problema binário da coluna vertebral.	47
2.15	Diagramas de caixa com os resultados obtidos para os classificadores SVM quando aplicados ao problema PCV-3C.	49
2.16	Diagramas de caixa com os resultados obtidos para os classificadores LSSVM para o problema PCV-3C.	50
2.17	Acurácia de diversos classificadores em função do tamanho do conjunto de treinamento.	51
2.18	(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.	51
3.1	Interpretação geométrica das variáveis de folga utilizadas em classificadores SVM. Valores na faixa $0 < \xi < 1$ indicam exemplos na margem de separação de sua classe e valores na faixa $\xi > 1$ indicam exemplos incorretamente classificados.	55
3.2	Hiperplanos construídos por um classificador LSSVM no espaço de características e regiões correspondentes.	57
3.3	Separação do conjunto de dados $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^l$ em dois subconjuntos: $\mathcal{D}^{(1)} = \{(\mathbf{x}_i, y_i) y_i = +1\}$ (quadrados) e $\mathcal{D}^{(2)} = \{(\mathbf{x}_i, y_i) y_i = -1\}$ (círculos). Equivale ao Passo 1 do método OM.	65
3.4	Treinamento da rede SOM usando o subconjunto $\mathcal{D}^{(1)}$ (SOM-1). Equivale ao Passo 2.a.	65
3.5	Treinamento da rede SOM usando o subconjunto $\mathcal{D}^{(2)}$ (SOM-2). Equivale ao Passo 2.b.	66
3.6	Realização de poda de todos os neurônios mortos nas redes SOM-1 e SOM-2. Como resultado tem-se as redes PSOM-1 e PSOM-2. Equivale ao Passo 3. . . .	66
3.7	Apresentação do conjunto $\mathcal{D}^{(1)}$ à rede PSOM-2. Equivale ao Passo 4.1a. . . .	66

3.8	Busca dos neurônios vencedores para o conjunto de dados $\mathcal{D}^{(2)}$ na rede PSOM-2. Equivale ao Passo 4.1b.	67
3.9	Para cada neurônio ativado no Passo 4.1a , encontrar seus K pontos mais próximos. Refere-se a este conjunto como $\mathcal{X}^{(1)}$. Equivale ao Passo 6.1.	67
3.10	Apresentação do conjunto $\mathcal{D}^{(2)}$ à rede PSOM-1. Equivale ao Passo 4.2a.	67
3.11	Busca dos neurônios vencedores para o conjunto $\mathcal{D}^{(1)}$ na rede PSOM-1. Equivale ao Passo 4.2b.	68
3.12	Para cada neurônio ativado no Passo 4.2a , encontrar seus K pontos mais próximos. Refere-se a este conjunto como $\mathcal{X}^{(2)}$. Equivale ao Passo 6.2.	68
3.13	O conjunto reduzido de padrões é dado por $\mathcal{X}_{rs} = \mathcal{X}^{(1)} \cup \mathcal{X}^{(2)}$. Passo Final.	68
3.14	(a) Problema artificial linearmente separável. (b) Resultado do método OM para o problema Artificial I.	71
3.15	(a) Problema artificial não-linearmente separável. (b) Resultado do método OM para o problema Artificial II.	72
3.16	(a) Problema artificial linearmente separável. (b) Resultado do método OM para o problema Artificial III.	73
3.17	(a) Problema artificial não-linearmente separável, em que os dados de uma classe estão envoltos pelos dados da outra classe. (b) Resultado do método OM para o problema Artificial IV.	73
3.18	(a) Superfície de decisão e número de vetores-suporte para a SVM padrão treinada com o algoritmo SMO. (b) Superfície de decisão e quantidade de vetores-suporte para o classificador OM-SVM.	74
3.19	(a) Superfície de decisão e os vetores-suporte para o classificador SVM padrão com <i>kernel</i> RBF. (b) Superfície de decisão e os vetores-suporte obtidos do classificador GOM-SVM/K ² M/RBF.	75
3.20	Evolução do número médio de vetores-suporte (série com quadrados) e do acerto médio (série com triângulos) em função do número de protótipos por rede GNG, no classificador GNG-SVM, para o problema PCV-2C.	77
3.21	Gráfico comparativo entre os classificadores SVM e OM-SVM para o problema PCV-2C quando os classificadores são treinados com diferentes percentuais do conjunto de treinamento.	78

3.22	Curva ROC obtida a partir dos classificadores SVM e OM-SVM para o problema da coluna vertebral (PCV-2C).	79
3.23	Curvas ROC para os classificadores SVM/RBF e GOM-SVM/K ² M/RBF para o problema PCV-2C.	80
4.1	Ilustração das regiões de rejeição e aceitação da regra ótima de reconhecimento.	86
4.2	Ilustração da curva de compromisso entre o erro e a rejeição.	86
4.3	Exemplo ilustrativo de uma curva hipotética de compromisso entre o erro e a rejeição.	87
4.4	Regiões de decisão obtidas pela utilização de um único classificador para um problema binário.	89
4.5	Regiões de decisão obtidas pela utilização de dois classificadores independentes para um problema binário.	90
4.6	Exemplo ilustrativo da função $h(\xi_i, \varepsilon)$.	92
4.7	(a) Problema artificial com 3 classes em que são apresentados os padrões no espaço \mathbb{R}^2 obedecendo o conceito de ordem. (b) Problema binário artificial.	94
4.8	Processo de replicação de dados de um problema binário artificial em que os padrões são expandidos para o espaço \mathbb{R}^3 .	94
4.9	Hiperplano de separação entre as classes.	95
4.10	Hiperplanos de separação entre as classes no espaço original.	95
4.11	(a) Curva A-R quando se utiliza 5% dos dados para treinamento (b) Curva A-R quando se utiliza 25% dos dados de treinamento.	100
4.12	(a) Curva A-R quando se utiliza 40% dos dados para treinamento. (b) Curva A-R quando se utiliza 60% dos dados para treinamento.	101
4.13	(a)Curva A-R quando se utiliza 80% dos dados para treinamento. (b) Resultados para diversas técnicas de aprendizado de máquina.	101
4.14	(a) Curva A-R obtida para 20% do total de dados no conjunto de treinamento. (b) Curva A-R obtida para 40% do total de dados no conjunto de treinamento. Estes resultados foram obtidos para o conjunto Sintético I.	104

4.15 (a) Curva A-R obtida para 25% do total de dados no conjunto de treinamento.	105
(b) Curva A-R obtida para 40% do total de dados no conjunto treinamento.	
Estes resultados foram obtidos para o problema PCV-2C.	105
4.16 (a) Curva A-R obtida para 60% do total de dados no conjunto de treinamento.	
(b) Curva A-R obtida para 80% do total de dados para treinamento. O problema	
avaliado é o PCV-2C.	105
4.17 (a) Curvas A-R para estratégias de rejeição treinadas com 60% do total de da-	
dos. (b) Curvas A-R para estratégias de rejeição treinadas com 80% do total de	
dados. O problema avaliado é o PCV-2C.	107
5.1 Curvas das taxas de acerto médio em função de P% para comitês homogêneos	
de $L = 5$ classificadores.	125
5.2 Diagrama de caixas (boxplots) das taxas de acerto de classificação para os co-	
mitês homogêneos ($P = 100\%$).	125
5.3 Gráfico do desempenho de classificação em função de P% para os melhores	
comitês heterogêneos com $L = 5$ classificadores-base.	127
5.4 Gráfico do desempenho de classificação em função de P% para os comitês ho-	
mogêneos e o melhor comitê heterogêneo, formado por $L = 5$ classificadores-base.	128
5.5 Gráfico do desempenho de classificação em função de P% para os comitês ho-	
mogêneos e os classificadores individuais.	129
A.1 (E) Diagrama de Caixa do atributo incidência pélvica. (D) Diagrama de caixa	
do atributo versão pélvica.	134
A.2 (E) Diagrama de Caixa do atributo ângulo de lordose. (D) Diagrama de caixa	
do atributo declive sacral.	134
A.3 (E) Diagrama de Caixa do atributo raio pélvico. (D) Diagrama de caixa do	
atributo grau de deslizamento.	135
A.4 (E) Gráficos de dispersão e curvas de nível dos atributos raio pélvico e grau de	
deslizamento. (D) Gráficos de dispersão e curvas de nível dos atributos versão	
pélvica e grau de deslizamento.	135
A.5 (E) Gráficos de dispersão e curvas de nível dos atributos ângulo de lordose e	
grau de deslizamento. (D) Gráficos de dispersão e curvas de nível dos atributos	
declive sacral e grau de deslizamento.	136

Lista de Tabelas

2.1	Listagem de importantes <i>kernels</i>	28
2.2	Número de padrões por classe nos problemas PCV-3C e PCV-2C.	40
2.3	Nomenclatura para os classificadores SVM e LSSVM.	41
2.4	Resultados dos classificadores SVM para o problema binário da coluna vertebral.	42
2.5	Resultados dos classificadores LSSVM para o problema binário da coluna vertebral.	44
2.6	Resultados do classificador SVM para o problema da coluna vertebral com 3 classes.	48
2.7	Resultados dos classificadores LSSVM para o problema da coluna vertebral com 3 classes.	49
2.8	Resultados obtidos para diversos classificadores aplicados ao problema PCV-3C. Pode-se observar os acertos médios obtidos para diferentes tamanhos do conjunto de treinamento.	50
3.1	Métodos (OM e GOM) e classificadores propostos nesta capítulo.	70
3.2	Parâmetros de treinamento da rede SOM para os problemas Artificial I, Artificial II, Artificial III e Artificial IV.	72
3.3	Parâmetros dos classificadores OM-SVM e GOM-SVM/K ² M.	74
3.4	Resultados para os classificadores SVM, OM-SVM, GOM-SVM/KM, GOM-SVM/GNG e GNG-SVM para o conjunto PCV-2.	76
3.5	Resultados para os classificadores LSSVM, OM-LSSVM, GOM-LSSVM/KM, GOM-LSSVM/GNG e GNG-LSSVM para o conjunto PCV-2C.	76
3.6	Resultados para os classificadores SVM/RBF e GOM-SVM/K ² M/RBF.	79
3.7	Resultados para os classificadores LSSVM/RBF e GOM-LSSVM/K ² M/RBF.	79
4.1	Exemplo da matriz de custos.	96

4.2	Resultados para os classificadores rejoSVM, SVM-1C, SVM-2C quando treinados com 40% e 80% do total de dados do problema PCV-2C.	102
4.3	Resultados para os classificadores SVM/SMO/KMOD, libSVM/Linear, oSVM, MLP e GRNN quando treinados com 40% e 80% do total de dados do problema PCV-2C.	102
5.1	Número de padrões do conjunto S por percentagem de <i>outliers</i>	120
5.2	explanação da matriz de confusão.	121
5.3	Resultados para os classificadores individuais SVM, MLP e GRNN.	122
5.4	Melhor matriz de confusão - Classificador SVM.	122
5.5	Melhor matriz de confusão - Classificador MLP.	123
5.6	Melhor matriz de confusão - Classificador GRNN.	123
5.7	Resultados para comitês homogêneos de $L = 5$ classificadores.	124
5.8	Matriz de confusão - Comitê C-SVM.	126
5.9	Matriz de confusão - Comitê C-MLP.	126
5.10	Matriz de confusão - Comitê C-GRNN.	126
A.1	Autovetores para cada um dos atributos biomecânicos.	136
A.2	Autovalores associados a cada um dos autovetores apresentados anteriormente.	136
A.3	Percentual de informação contida em cada uma das componentes principais.	136
A.4	Combinação dos atributos tomados seis-a-seis. Ou seja, utilizando todos os atributos.	137
A.5	Combinação dos atributos tomados cinco-a-cinco.	137
A.6	Combinação dos atributos tomados quatro-a-quatro.	138
A.7	Combinação dos atributos tomados três-a-três.	138
A.8	Combinação dos atributos tomados dois-a-dois.	139
A.9	Resumo dos resultados obtidos pela combinação de atributos.	139
A.10	Resultados obtidos para diversas configurações da rede MLP.	140

Lista de Símbolos

x	Vetor de atributos
x ^(s)	Vetor-suporte
<i>y</i>	Saída do classificador
<i>y_k</i>	Saída ponderada do classificador
<i>l</i>	Número de classes
<i>d</i>	Saída desejada
<i>d</i> ^(vs)	Saída do vetor-suporte
(x , <i>y</i>)	Padrão ou amostra
<i>n</i>	Tamanho do conjunto de treinamento
\bar{n}	Tamanho do conjunto de teste
<i>n</i>	Tamanho do conjunto de treinamento
w	Vetor de pesos do hiperplano
<i>b</i>	Viés do hiperplano
w_o	Vetor de pesos ótimo do hiperplano
<i>b_o</i>	Viés ótimo do hiperplano
sign(.)	Função sinal.
<i>r</i> ^(s)	Distância do vetor-suporte ao hiperplano ótimo
<i>ρ</i>	Margem de separação
<i>α</i>	Vetor de multiplicadores de Lagrange
<i>α_i</i>	Multiplicador de Lagrange

β	Vetor de multiplicadores de Lagrange
β_i	Multiplicador de Lagrange
C	Parâmetro de regularização do classificador SVM
γ	Parâmetro de regularização do classificador LSSVM
ξ	Vetor de variáveis de folga
ξ_i	Variável de folga
n	Número de padrões de treinamento
n_{vs}	Número de vetores-suporte
$\phi(.)$	Vetor característico
$K(.,.)$	Função kernel
N	Dimensão do vetor de entrada
M	Dimensão do vetor de características
\mathbb{R}	Conjunto dos números Reais
max	Função Máximo
min	Função Mínimo
J	Matriz jacobiana
t_d	Estatística do teste-t pareado
C_{+1}	Classe positiva
C_{-1}	Classe positiva
h_{+1}	Hiperplano da Classe C_{+1}
h_{-1}	Hiperplano da Classe C_{-1}
h_{+1}	Hiperplano da Classe C_{+1}
h_{-1}	Hiperplano da Classe C_{-1}
t	Limiar de rejeição

W_r	Custo de rejeição
W_c	Custo de acertar
W_e	Custo de errar
$\mathbf{h}(\cdot, \cdot, \cdot, \cdot)$	Função de decaimento exponencial da vizinhança
$\eta(\cdot)$	Função decaimento exponencial da aprendizagem
p_k	Probabilidade a priori da Classe C_k
$E(t)$	Função de erro
$R(t)$	Função de rejeição
\mathbf{A}_{rs}	Matriz não-quadrada reduzida
$\mathcal{D}^{(1)}$	Conjunto de treinamento da Classe C_{+1}
$\mathcal{D}^{(2)}$	Conjunto de treinamento da Classe C_{-1}
\mathcal{X}_{rs}	Conjunto de treinamento reduzida
$\mathcal{X}^{(1)}$	Conjunto de treinamento reduzido da Classe C_{+1}
$\mathcal{X}^{(2)}$	Conjunto de treinamento reduzido da Classe C_{-1}
δ	Regra ótima de decisão de Chow
δ	Gradiente
δ^2	Hessiana
$C(\mathbf{x})$	Índice da classe mais votada
E_{total}	Erro total
E_{bayes}	Erro de Bayes
E_{add}^{ind}	Erro adicionado pelo componentes-base do comitê
E_{add}^{end}	Erro adicionado pelo comitê
E_{total}	Erro total
m	Quantidade de componentes-base do comitê

1 *Introdução*

Aárea de pesquisa em Aprendizado de Máquina (AM) tem por objetivo maior desenvolver ferramentas computacionais eficientes capazes de fornecer, senão a melhor, pelo menos soluções satisfatórias, a problemas complexos de classificação de padrões, em especial aos de natureza não-linear, incerta e de elevada dimensionalidade. Como exemplo de propriedades desejáveis em tais ferramentas, podem-se citar a capacidade de construir fronteiras de decisão arbitrárias fazendo pouca ou nenhuma suposição acerca da distribuição dos dados disponíveis, e a capacidade de generalizar o conhecimento para novos dados (WEBB, 2002).

Atualmente verifica-se a disseminação do uso de estratégias de AM em diversos tipos de aplicações e áreas do conhecimento, tais como Medicina e Biologia (SIERMALA et al., 2008; VOLYANSKYY et al., 2009; TZALLAS et al., 2009; GHORAI et al., 2010; GAVRISHCHAKA et al., 2010; RAHMAN et al., 2011; BI et al., 2011). Uma explicação para a disseminação do uso dessas ferramentas de AM reside na capacidade limitada de diagnóstico humano sob condições adversas, tais como estresse, fadiga e pouco conhecimento técnico. Sob estas condições, sistemas computacionais de diagnóstico costumam apresentar desempenho melhor que os de especialistas humanos (BRAUSE, 1999).

Em Reggia (1993), Papik et al. (1998), Brause (2001), Ramesh et al. (2004) e Cios et al. (2007) são apresentadas várias aplicações de estratégias de AM em diversas áreas da Medicina, tais como cardiologia, análise de eletrocardiogramas, gastroenterologia, pneumologia, oncologia, neurologia, análise de electroencefalogramas, otorrinolaringologia, ginecologia e obstetrícia, oftalmologia, radiologia, patologia, citologia, genética, bioquímica, dentre outras. Em Robert et al. (2004) é feito um levantamento do número de artigos (mais de 800) publicados envolvendo a utilização de Redes Neurais Artificiais (RNAs) em Medicina e Biologia nos anos 2000 e 2001 em mais de 40 países. Além de RNAs , vale ressaltar que outras técnicas de AM, tais como Máquinas de Vetores-Suporte (SVM) e Comitês de Classificadores, têm contribuído bastante para a área de classificação e diagnóstico médico (ZHOU; JIANG, 2003; MANGIAMELI et al., 2004; LI et al., 2008; XIANG et al., 2009; GUIMERA-TOMAS et al., 2010; DOBROWOLSKI et al., 2010).

1.1 Motivação para a Tese

Embora o uso de técnicas de AM já esteja bastante difundido em Medicina Diagnóstica, de um modo geral, a aplicação dessas técnicas em Traumato-Ortopedia é bastante escassa na literatura especializada. Os poucos trabalhos correlatos são descritos a seguir.

Em Ohno-Machado & Rowland (1999), os autores descrevem a utilização de RNAs na inferência e prognóstico em Ortopedia, relacionados a danos na medula espinhal. Em Grigsby et al. (1994) é apresentada uma aplicação de redes neurais na predição do tempo de reabilitação e do uso de recursos hospitalares no tratamento de pacientes ortopédicos. Além destes trabalhos, Antani et al. (2003) mostram a utilização de RNAs na recuperação de imagens da coluna vertebral com base em conteúdo e Cherukuri et al. (2004) reportam um sistema de classificação de indivíduos em normais ou anormais para a patologia osteófita. Já no trabalho de Bounds & Lloyd (1998) é descrita uma aplicação da rede MLP no diagnóstico de dor lombar. Por fim, aos leitores interessados recomenda-se a leitura de Schollhorn (2004), que apresentada uma revisão de trabalhos envolvendo a aplicação de estratégias de AM na área de Biomecânica Clínica.

Até meados de 2005, a escassez de trabalhos voltados ao diagnóstico automático de patologias da coluna vertebral se devia em grande parte à ausência de atributos numéricos que descrevessem quantitativamente as patologias de interesse para o campo da Ortopedia, de modo a gerar um conjunto de dados adequado para o projeto de classificadores de padrões.

No entanto, há poucos anos, um grupo de especialistas em Ortopedia e Biomecânica definiu um conjunto de descritores (atributos) biomecânicos relacionados a dores e deformidades da coluna vertebral (FIÉRE; DA MOTA, 2001; LABELLE et al., 2005; BERTHONNAUD et al., 2005). Em seguida, este grupo realizou medidas dos atributos definidos em vários pacientes, formando assim um conjunto de dados com casos clínicos relevantes para o estudo. Este conjunto de atributos biomecânicos que estão associados a dores e deformidades da coluna vertebral foi então utilizado para projetar classificadores e, como resultado, foi proposto a plataforma SINPATCO (Sistema INteligente para diagnóstico de PATologias da COnuna vertebral), doravante chamado de SINPATCO I, voltada para classificação semi-automática de patologias da coluna vertebral (ROCHA-NETO, 2006).

Os atributos mencionados no parágrafo anterior foram pioneiramente utilizados na plataforma SINPATCO I. Esta plataforma possui três módulos, a saber: interface gráfica, modulo de diagnóstico e de explanação. O módulo de interface gráfica permite uma interação amigável com o especialista médico. O módulo de diagnóstico possui os seguintes classificadores: (*K Nearest Neighbors*) e NB (*Naive Bayes*), assim como as redes neurais MLP (*Multilayer*

Perceptron), GRNN (Generalized Regression Neural Network) (SPECHT, 1990) e SOM (Self-Organizing Map). O módulo de extração de conhecimento é responsável pela extração de regras a partir dos classificadores treinados, a fim de elucidar para o médico ortopedista como o classificador chega ao diagnóstico final.

O módulo de diagnóstico é composto por uma unidade de pré-processamento, responsável pela normalização dos dados e filtragem de amostras discrepantes (*outliers*), pelos classificadores de padrões supracitados que são treinados com os casos clínicos previamente rotulados. Os classificadores do módulo de diagnóstico são avaliados quanto à capacidade de categorizar corretamente os casos clínicos em uma das seguintes classes: Normal, Hérnia de Disco e Espondilolistese.

Em sistemas de auxílio ao diagnóstico baseados em técnicas de AM a qualidade da informação produzida é extremamente dependente da qualidade dos dados (número de casos clínicos disponíveis para projeto do classificador, presença de ruído e outliers, número de atributos, atributos ausentes, etc.) e da escolha de um classificador que informe de modo confiável a presença ou ausência de certa patologia a partir do treinamento e validação sobre o conjunto de dados previamente coletado.

No entanto, as implicações negativas de um diagnóstico incorreto têm motivado buscas por estratégias de AM sempre mais eficientes, tais como SVMs (VAPNIK, 1998, 2000). Grosso modo, SVMs são classificadores com apenas uma camada oculta de processadores simples, submetidas a treinamento supervisionado, cujos princípios de funcionamento são oriundos da Teoria do Aprendizado Estatístico. O seu treinamento consiste basicamente na resolução de um problema de programação quadrática, o que equivale à tarefa de minimização simultânea do erro empírico (relacionado ao erro no treinamento) e do erro estrutural (relacionado à complexidade do modelo). Uma variante da SVM é a *Least Squares SVM* (SUYKENS; VANDEWALLE, 1999a), que reduz o problema de programação quadrático das SVMs a um sistema linear.

Na plataforma SINPATCO I vários classificadores são treinados com estes dados, sendo escolhido para uso aquele com melhor desempenho (i.e. menos erros). Pesquisas recentes sugerem que uma estratégia alternativa à seleção do melhor classificador individual consiste em empregar um modelo baseado na combinação de classificadores (comitês de classificadores ou ensembles) (DIETTERICH, 2000; TAN; GILBERT, 2003; MERKWIRTH et al., 2004; MANGIAMELI et al., 2004; DIETTERICH, 2002; MOSKOVITCH et al., 2008; MENAHEM et al., 2009; ROKACH, 2010).

Em sua forma mais simples, comitês são formados por arranjos paralelos de vários classificadores treinados, em geral, de forma independente. Os classificadores individuais podem ser

de um mesmo tipo (e.g. MLPs de uma camada oculta) ou de tipos diferentes (MLPs e SVMs). No primeiro caso tem-se um comitê homogêneo, enquanto no segundo, um comitê heterogêneo. Neste contexto, para que um comitê possua um desempenho superior a qualquer um de seus componentes individuais basta que estes tenham um bom desempenho isoladamente e que apresentem erros de classificação diferentes frente a um mesmo conjunto de dados, i. e., sejam diversos entre si (HANSEN; SALAMON, 1990).

Com o intuito de evitar implicações negativas de um diagnóstico incorreto, técnicas de classificação com opção de rejeição também têm sido utilizadas em Medicina Diagnóstica (RAMOSER et al., 2005; HANCZAR; DOUGHERTY, 2008; QUEVEDO et al., 2011; ROCHA-NETO et al., 2011). Como consequência, a automatização destas decisões podem conduzir a diversas previsões erradas, e portanto a elevação do erro de classificação. Neste tipo de sistema a classificação com opção de rejeição é inserida para salvaguardar contra erros excessivos e tomadas de decisão difíceis (CHOW, 1970).

As características encontradas em estratégias de classificação usando SVMs, opção de rejeição ou comitês de classificadores são bastante desejáveis e, portanto, a incorporação destas estratégias na plataforma SINPATCO pode permitir um aumento considerável na qualidade e no desempenho de classificação. Tais incorporações conduzem à plataforma SINPATCO II. O histórico da plataforma SINPATCO, tanto em sua primeira versão (resultado do mestrado) quanto em sua segunda (resultado do doutorado), é apresentado na Figura 1.1.

Na Figura 1.1, são apresentadas ainda informações acerca da cooperação internacional no âmbito do presente projeto de doutorado, realizada entre a Universidade Federal do Ceará (UFC), através do Programa de Pós-Graduação em Engenharia de Teleinformática (PPGETI), e a Faculdade de Engenharia da Universidade do Porto (FEUP). Esta cooperação permitiu que o autor desta tese realizasse estágio de doutorado-sanduíche em Portugal, que resultou no contato com um grupo de pesquisadores especialistas em classificação com opção de rejeição. Permitiu também a realização de doutorado-sanduíche no Brasil (Março/2011 à Maio/2011) no PPGETI, do doutorando português Ricardo Gamelas de Sousa. Diversos outros pontos considerados relevantes, como o prêmio recebido de melhor artigo no evento IWANN'2001 e a futura cooperação, em virtude deste prêmio, com a Universidade de Granada/Espanha são listados na Figura 1.1.

O problema-alvo desta tese de doutorado concentra-se na área de Ortopedia, e mais especificamente no diagnóstico de patologias da coluna vertebral. Neste sentido, uma breve descrição sobre a coluna vertebral, as patologias hérnia de disco e espondilolistese, bem como sobre os atributos biomecânicos associados são apresentados a seguir.

Mar/2004	→ Início do mestrado.
Jan/2005	→ Acesso ao conjunto de dados de patologias da coluna vertebral.
Jun/2005	→ Implementação dos classificadores MLP, SOM, GRNN, Naive Bayes e kNN.
Mar/2006	→ Desenvolvimento SINPATCO I.
Abr/2006	→ Fim do mestrado.
Ago/2007	→ Início do doutorado.
Set/2007	→ Implementação do classificador SVM/Adatron.
Out/2008	→ Implementação de Comitês de Classificadores (GRNN, MLP e SVM).
Jul/2009	→ Implementação do classificador SVM/SMO.
Fev/2010	→ Aprovação de projeto para realização de doutorado sanduíche no âmbito do prog. CNPq/Univ. do Porto/590008/2009-9.
Ago/2010	→ Idade à Portugal realizar doutorado sanduíche na Faculdade de Engenharia da Universidade do Porto/INESC.
Out/2010	→ Classificação com opção de rejeição (SVM-1C, SVM-2C, MLP-1C, MLP-2C, Fumera e rejoSVM).
Dez/2010	→ Implementação do método Opposite Maps (Proposta).
Dez/2010	→ Implementação de estratégias para obtenção de conjuntos reduzidos em classificadores SVM e LSSVM (Propostas).
Jan/2011	→ Retorno do doutorado sanduíche.
Mar/2011	→ Realização de pesquisa conjunta com pesquisador português no Brasil.
Jul/2009	→ Implementação de Classificadores com opção de rejeição, SOM-1C e SOM-2C (Propostas).
Jun/2011	→ Prêmio de melhor artigo IWANN'2011 (método Opposite Maps).
Ago/2011	→ Planejamento de cooperação com grupo da Univ. de Granada/Espanha em Dez/2011, em virtude do prêmio no IWANN'2011.
Jul/2011	→ Desenvolvimento SINPATCO II.
Set/2011	→ Fim do doutorado.

Figura 1.1: Histórico de desenvolvimento das plataformas SINPATCO I e II.

1.2 Descrição do Problema

A coluna é um sistema composto por um conjunto de vértebras, discos intervertebrais, nervos, músculos, medula e ligamentos. As principais funções da coluna vertebral são as seguintes: eixo de suporte do corpo humano; protetor ósseo da medula espinhal e das raízes nervosas; e eixo de movimentação do corpo, possibilitando o movimento nos três planos: frontal, sagital e transversal (HALL, 2000).

Esse complexo sistema está sujeito a disfunções que causam dor nas costas, das mais variadas intensidades. Hérnia de disco e espondilolistese são exemplos de patologias da coluna vertebral que causam dores intensas. A hérnia de disco surge como resultado de diversos pequenos traumas na coluna que vão, com o passar do tempo, lesando as estruturas do disco intervertebral, ou pode acontecer como consequência de um trauma severo sobre a coluna. Na hérnia de disco o núcleo do disco intervertebral migra de seu local, no centro do disco para a periferia, em direção ao canal medular ou nos espaços por onde saem as raízes nervosas, levando à compressão destas raízes.

Espondilolistese ocorre quando uma das 33 vértebras da coluna vertebral desliza adiante em

relação às outras. Este deslizamento quando verificado ocorre, geralmente, em direção à base da espinha na região lombar, ocasionando dor ou sintomatologia de irritação de raiz nervosa. O mecanismo que ocasiona esse tipo de lesão não é bem conhecido, mas existem teorias que sugerem algumas possíveis causas: fratura por fadiga conjugada a um defeito hereditário ou predisposição, fratura ocorrida durante o parto, trauma, deslocamento de uma vértebra sobre a outra secundária à lordose lombar, fraqueza dos ligamentos e estruturas fasceais da região envolvida ou má formação das facetas articulares.

Para treinar classificadores capazes de discriminar patologias da coluna vertebral faz-se necessário definir um conjunto de atributos que possam ser medidos para cada paciente analisado pelo ortopedista. Estes atributos são usados para descrever quantitativamente cada paciente (i.e. caso clínico), com o objetivo de relacionar os sintomas apresentados pelo paciente com o arranjo estrutural (biomecânico) da coluna vertebral. Os atributos de interesse para esta tese de doutorado são descritos a seguir.

1.2.1 Atributos Biomecânicos

A base de dados utilizada neste trabalho contém dados extraídos de 310 pacientes, a partir de radiografias panorâmicas sagitais em formato de 30×90 cm. Destes, 100 indivíduos são voluntários que não possuem patologias na coluna, doravante chamados de normais. Os dados restantes são obtidos a partir de radiografias de pacientes operados de hérnias de disco (60 indivíduos) ou espondilolistese (150 indivíduos). Cada um dos 310 pacientes é descrito por seis atributos biomecânicos: ângulo de incidência pélvica, ângulo de versão pélvica, declive sacral, ângulo de lordose, raio pélvico e grau de deslizamento. A correlação destes atributos com patologias comuns da coluna vertebral (e.g. hérnia de disco e espondilolistese) foi originalmente proposta na referência (BERTHONNAUD et al., 2005).

O ângulo de incidência pélvica (*pelvic incidence*, PI) é o ângulo subtendido pelo segmento de reta \overline{ao} , que vai do centro da cabeça femoral (ponto *o*) ao ponto médio *a* da placa sacral terminal, e uma reta perpendicular ao centro da placa sacral \overline{bc} no ponto *a* (ver Figura 1.2(a)). A placa terminal sacral é definida pelo segmento de reta \overline{bc} entre o canto superior posterior do sacrum e a ponta anterior da placa terminal S1 no promontório sacral. Para o caso em que as cabeças femorais não são sobrepostas, o centro de cada cabeça femoral é marcado, e um segmento de reta deve ligar os centros das cabeças femorais.

O ângulo de versão pélvica (*pelvic tilt*, PT), conforme indicado na Figura 1.2(b), é descrito como o ângulo subtendido por uma reta de referência vertical (VRL), originada do ponto *o* (centro da cabeça do fêmur) e o segmento de reta \overline{ao} . Deve-se enfatizar que esta afirmação está

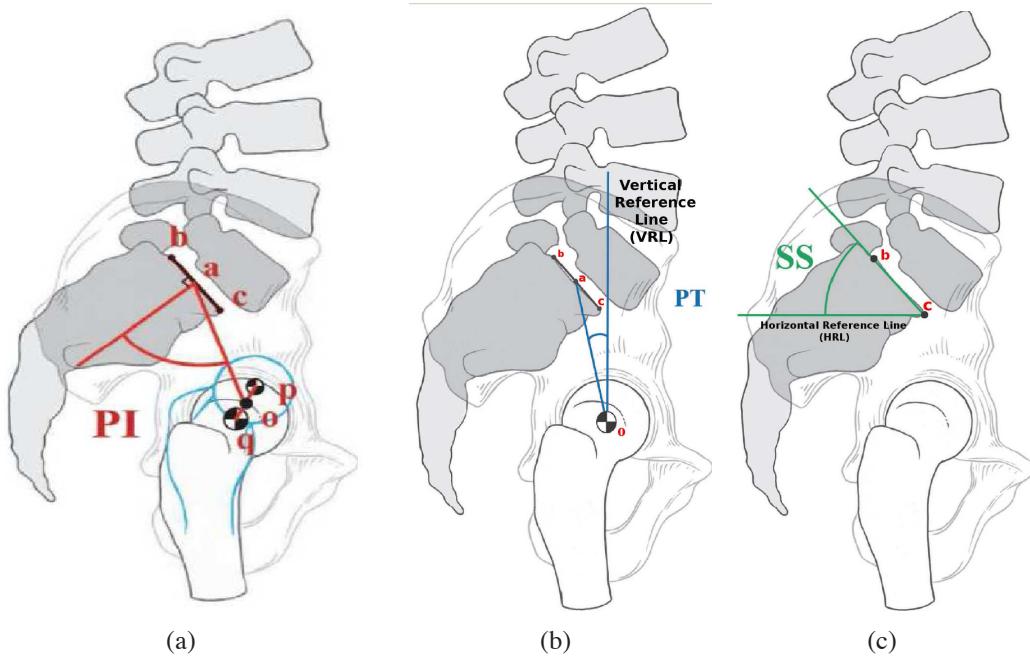


Figura 1.2: Descrição dos atributos biomecânicos.

correta somente quando o verdadeiro eixo hipotético está em frente ao ponto médio da placa sacral terminal.

O declive sacral (*sacral slope*, SS) é definido como o ângulo subtendido por uma linha de referência horizontal (HRL) e pela reta-suporte da placa terminal sacral, conforme ilustrado na Figura 1.2(c).

O ângulo de lordose é o maior ângulo sagital entre o platô superior do sacro e o platô superior da vértebra lombar ou torácica limite. O raio pélvico (segmento de reta \overline{ao}) é a distância do centro do eixo bicoxofemural ao centro do platô sacral. Este segmento de reta é ilustrado na Figura 1.2(b). Por último, o grau de deslizamento é o grau percentual de deslizamento entre o platô inferior da quinta vértebra lombar e o sacro.

1.3 Objetivos Geral e Específicos

1.3.1 Objetivo Geral

O trabalho reportado nesta tese comprehende uma segunda fase no desenvolvimento da plataforma SINPATCO, fase esta cujo o objetivo é aumentar a acurácia dos classificadores responsáveis pelo diagnóstico de patologias da coluna vertebral.

1.3.2 Objetivos Específicos

Em um sentido mais específico, esta tese tem as seguintes metas particulares:

- Avaliar o desempenho de classificadores SVM e LSSVM utilizando diferentes tipos de *kernel*;
- Apresentar um novo método para treinamento do classificador LSSVM com base no algoritmo de Levenberg-Marquardt;
- Apresentar um novo método para obtenção de conjuntos reduzidos em SVMs e LSSVMs;
- Apresentar novas técnicas de classificação com opção de rejeição com base na rede SOM;
- Avaliar o desempenho de comitês de classificadores homogêneos e heterogêneos quando aplicados ao diagnóstico de patologias da coluna vertebral a partir de classificadores-base do tipo SVM, MLP e GRNN;
- Adicionar à plataforma SINPATCO modelos que baseiam-se em comitês de classificadores, bem como modelos que realizam classificação com opção de rejeição.

1.4 Produção Científica

Durante o transcorrer deste projeto de doutorado, foram publicados os seguintes trabalhos científicos:

- **ROCHA NETO, A. R. ; SOUSA, R. ; BARRETO, G. A. ; CARDOSO, J. S.** . Diagnostic of Pathology on the Vertebral Column with Embedded Reject Option. In: 5th Iberian Conference on Pattern Recognition and Image Analysis, 2011, Las Palmas de Gran Canaria. Proceedings of the IbPRIA'2011 - **Lecture Notes in Computer Science**. Berlin Heidelberg : Springer-Verlag, 2011. v. 6669. p. 588-595.
- **ROCHA NETO, A. R. ; BARRETO, G. A.** . A Novel Heuristic for Building Reduced-Set SVMs Using the Self-Organizing Map (Best Paper of Youg Research). In: 11th International Work Conference on Artificial Neural Networks (IWANN'2011), 2011, Torremolinos. Proceedings of the IWANN'2011 - **Lecture Notes on Computer Science**. Heidelberg : Springer-Verlag, 2011. v. 6691. p. 97-104.
- **ROCHA NETO, A. R. ; BARRETO, G. A.** . On the Application of Ensembles of Classifiers to the Diagnosis of Pathologies of the Vertebral Column: A Comparative Analysis. **IEEE Latin America Transactions**, v. 7, p. 487-496, 2009.

- **ROCHA NETO, A. R.** ; BARRETO, G. A. . Aplicação de Máquinas de Vetor Suporte ao Diagnóstico de Patologias da Coluna Vertebral: Um Estudo Comparativo. In: Anais do Workshop on Computational Intelligence - SBRN 2008, 2008, Salvador/BA. **II Workshop on Computational Intelligence (SBRN 2008 - WCI)**, 2008.
- **ROCHA NETO, A. R.** ; BARRETO, G. A. ; CORTEZ, P. C. ; DA MOTA, H. . Aplicação de KNN e SOM no Auxílio ao Diagnóstico de Patologias da Coluna Vertebral. In: XXI Congresso Brasileiro de Engenharia Biomédica, 2008, Salvador/BA. Anais do **Congresso Brasileiro de Engenharia Biomédica**, 2008.
- **ROCHA NETO, A. R.** ; BARRETO, G. A. . Comitês de Classificadores para Diagnóstico Automático de Patologias da Coluna Vertebral: Um Estudo Comparativo. In: **VIII Congresso Brasileiro de Redes Neurais**, 2007, Florianópolis/SC. Anais do VIII CBRN, 2007.

1.5 Estrutura da Tese

1.5.1 Metodologia de Organização

Esta tese é subdivida em capítulos com base nas diversas estratégias de classificação de padrões de interesse. Os capítulos são organizados de forma a serem o mais auto-contidos possível em termos de conteúdo. Esta organização permite descrever mais especificamente a origem, as motivações e a fundamentação teórica para cada uma destas estratégias de aprendizado. Além disto, a organização proposta visa permitir que as contribuições teóricas desta tese sejam mais adequadamente apresentadas e inseridas no contexto de outras abordagens contidas na literatura.

Em virtude desta organização que também busca facilitar a leitura e o entendimento desta tese, ao final de cada capítulo são apresentadas as simulações computacionais, contendo informação sobre a metodologia de experimentação e os diversos resultados obtidos. No final de cada capítulo, são descritas ainda as conclusões relacionadas à estratégia de aprendizado abordada.

A forma como esta tese está organizada permite que o leitor interessado em uma determinada estratégia se restrinja a um determinado capítulo de interesse. Os detalhes contidos em cada capítulo são descritos de forma resumida na subseção a seguir.

1.5.2 Organização da Tese

O restante deste trabalho é formado por cinco capítulos que descrevem os diversos conceitos relacionados aos classificadores SVM e LSSVM, as técnicas para obtenção de conjuntos reduzidos, as estratégias para classificação com opção de rejeição e os comitês de classificadores.

Mais especificamente, o Capítulo 2 traz uma descrição da teoria de classificadores SVM e LSSVM. A teoria subjacente destes classificadores é apresentada em um nível de detalhe adequado para permitir o entendimento dos capítulos posteriores que apresentam classificadores SVM derivados dos convencionais, ou seja, que apresentam alterações na formulação do problema primal. Além disto, neste capítulo, uma proposta de solução para o classificador LSSVM baseada no método Levenberg-Marquardt é apresentada. Por fim, são apresentados os resultados da aplicação dos classificadores SVM e LSSVM ao problema de diagnóstico de patologias da coluna vertebral.

No Capítulo 3 são apresentados diversos conceitos relacionados à obtenção de conjunto reduzidos em classificadores SVM e LSSVM. Além disto, são propostos novos classificadores que apresentam custo computacional para avaliação de exemplos não-vistos reduzido em comparação com os classificadores SVM e LSSVM convencionais. Tais classificadores baseiam-se em um novo método proposto, denominado *Opposite Maps*, o qual pode ser aplicado tanto aos classificadores SVM quanto aos classificadores LSSVM.

O Capítulo 4 trata de classificação com opção de rejeição. São apresentadas as abordagens que baseiam-se em (i) um classificador padrão; (ii) em dois classificadores independentes; e (iii) em um classificador com opção de rejeição embutida. Ainda neste capítulo, são propostos dois novos métodos para fins de classificação com opção de rejeição baseados na Rede Auto-Organizável de Kohonen.

O Capítulo 5 descreve estratégias de classificação baseadas em Comitês de Classificadores. Descreve também as etapas necessárias para o projeto de comitês, tais como geração, seleção e combinação de componentes. São apresentados também os resultados obtidos para comitês homogêneos e heterogêneos, baseados em classificadores MLP, GRNN e SVM.

O Capítulo 6 são traz as considerações finais, comentários e análises dos resultados obtidos nesta tese. São listadas as propostas apresentadas por capítulo, bem como também são sugeridos trabalhos futuros relacionados com o tema abordado.

No Apêndice A são descritos outros conjuntos de resultados obtidos para o problema de diagnóstico de patologias da coluna vertebral. São apresentados os diagramas de caixa para os

atributos biomecânicos, os gráficos de dispersão dos atributos tomados dois-a-dois, análise de componentes principais e análise da generalização com base nas diversas combinações possíveis dos atributos. A partir dos resultados apresentados pode-se compreender mais completamente a relação entre os atributos e a complexidade do problema.

Por fim, o Apêndice B apresenta ainda os algoritmos de aprendizagem e diversas características da rede GNG, e dos algoritmos K-Médias e *Kernel K-Médias*.

2 Classificadores SVM e LSSVM

Este capítulo contém uma revisão bibliográfica sobre os classificadores SVM e LS-SVM. Neste capítulo 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 do *kernel*. No final do capítulo são apresentados ainda diversos resultados obtidos a partir da aplicação de SVM e LSSVM ao conjunto de dados da coluna vertebral, nas versões com 2 e 3 classes.

2.1 Introdução

O algoritmo *Generalized Portrait*, projetado para resolver problemas linearmente separáveis pode ser considerado o precursor dos classificadores SVM (VAPNIK; LERNER, 1963; VAPNIK; CHERVONENKIS, 1964). Posteriormente, classificadores SVM foram também denominados classificadores de margem ótima (*Optimal Margin Classifiers*, por Boser et al. (1992)); redes de vetores-suporte (*Support Vector Network*), por Cortes & Vapnik (1995); e então, a nomenclatura mais consolidada e difundida, máquinas de vetores-suporte (BURBIDGE; BUXTON, 2001).

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 capítulo. 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* (BURGES; CHRISTOPHER, 1998), enquanto a abordagem que aplica SVM a problemas de classificação de padrões é denominada *Support Vector Classification* (SMOLA; SCHÖLKOPF, 1998).

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 Perceptron - 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 (CRAMMER; SINGER, 2001), 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 (DUAN; KEERTHI, 2005), um-contra-todos (DUAN; KEERTHI, 2005), DAGSVM (PLATT et al., 2000) e por códigos corretores de erros (DIETTERICH, 1995) são as mais amplamente utilizadas.

2.2 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:

$$(\mathbf{x}_1, d_1), \dots, (\mathbf{x}_n, d_n) \in \mathbb{R}^N \times \{\pm 1\}, \quad (2.1)$$

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 estimativa da função, isto é, $f(\mathbf{x}_i) = d_i$. O conjunto de dados $(\mathbf{x}_1, d_1), \dots, (\mathbf{x}_n, d_n) \in \mathbb{R}^N \times \{\pm 1\}$ usado para estimar f é denominado conjunto de treinamento. Enquanto o conjunto de dados $(\tilde{\mathbf{x}}_1, \tilde{d}_1), \dots, (\tilde{\mathbf{x}}_n, \tilde{d}_n) \in \mathbb{R}^N \times \{\pm 1\}$ com os exemplos não utilizados na estimativa de f é denominado conjunto de teste.

Considere um problema linearmente separável, como mostrado na Figura 2.1. 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

$$\mathbf{w}^T \mathbf{x} + b = 0 , \quad (2.2)$$

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:

$$\begin{aligned} \mathbf{w}^T \mathbf{x}_i + b &\geq a \quad \text{para } d_i = +1 \\ \mathbf{w}^T \mathbf{x}_i + b &\leq a \quad \text{para } d_i = -1 \end{aligned} \quad (2.3)$$

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

No problema artificial apresentado na Figura 2.1 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 .

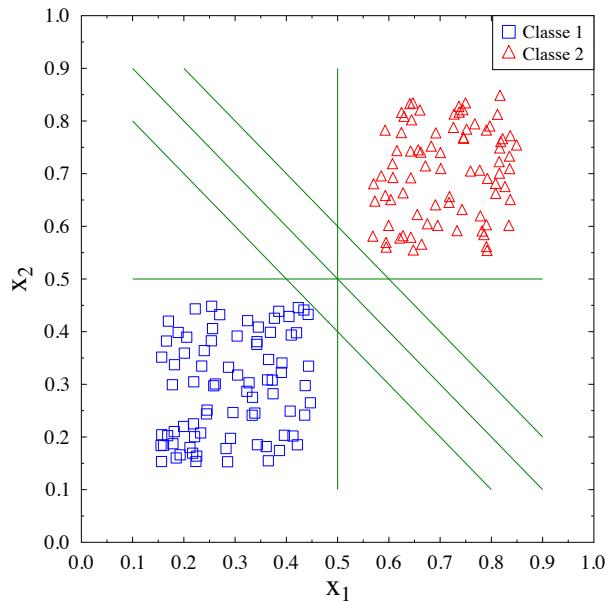


Figura 2.1: Os hiperplanos (retas) apresentados, descrevem algumas possíveis soluções para um problema linearmente separável. Duas classes hipotéticas são descritas por quadrados e triângulos, apresentando pontos gerados artificialmente.

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 2.2 ilustra o problema relacionado à escolha do melhor hiperplano de separação para um problema linearmente separável. Uma escolha como a realizada na Figura 2.2(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 2.2(c). Neste sentido, uma escolha mais adequada seria o hiperplano apresentado na Figura 2.2(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.

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

$$\mathbf{w}_o^T \mathbf{x} + b_o = 0. \quad (2.4)$$

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:

$$f(\mathbf{x}) = \mathbf{w}_o^T \mathbf{x} + b_o, \quad (2.5)$$

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

$$f(\mathbf{x}) = \text{sign}(\mathbf{w}_o^T \mathbf{x} + b_o). \quad (2.6)$$

tal que

$$f(\mathbf{x}) = \begin{cases} -1, & \text{se } \mathbf{w}_o^T \mathbf{x} + b_o < 0, \\ +1, & \text{se } \mathbf{w}_o^T \mathbf{x} + b_o \geq 0. \end{cases} \quad (2.7)$$

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 (2.3) pode ser reescrito como

$$\begin{aligned} \mathbf{w}_o^T \mathbf{x}_i + b_o &\geq +1 \quad \text{para } d_i = +1 \\ \mathbf{w}_o^T \mathbf{x}_i + b_o &\leq -1 \quad \text{para } d_i = -1 \end{aligned} \quad (2.8)$$

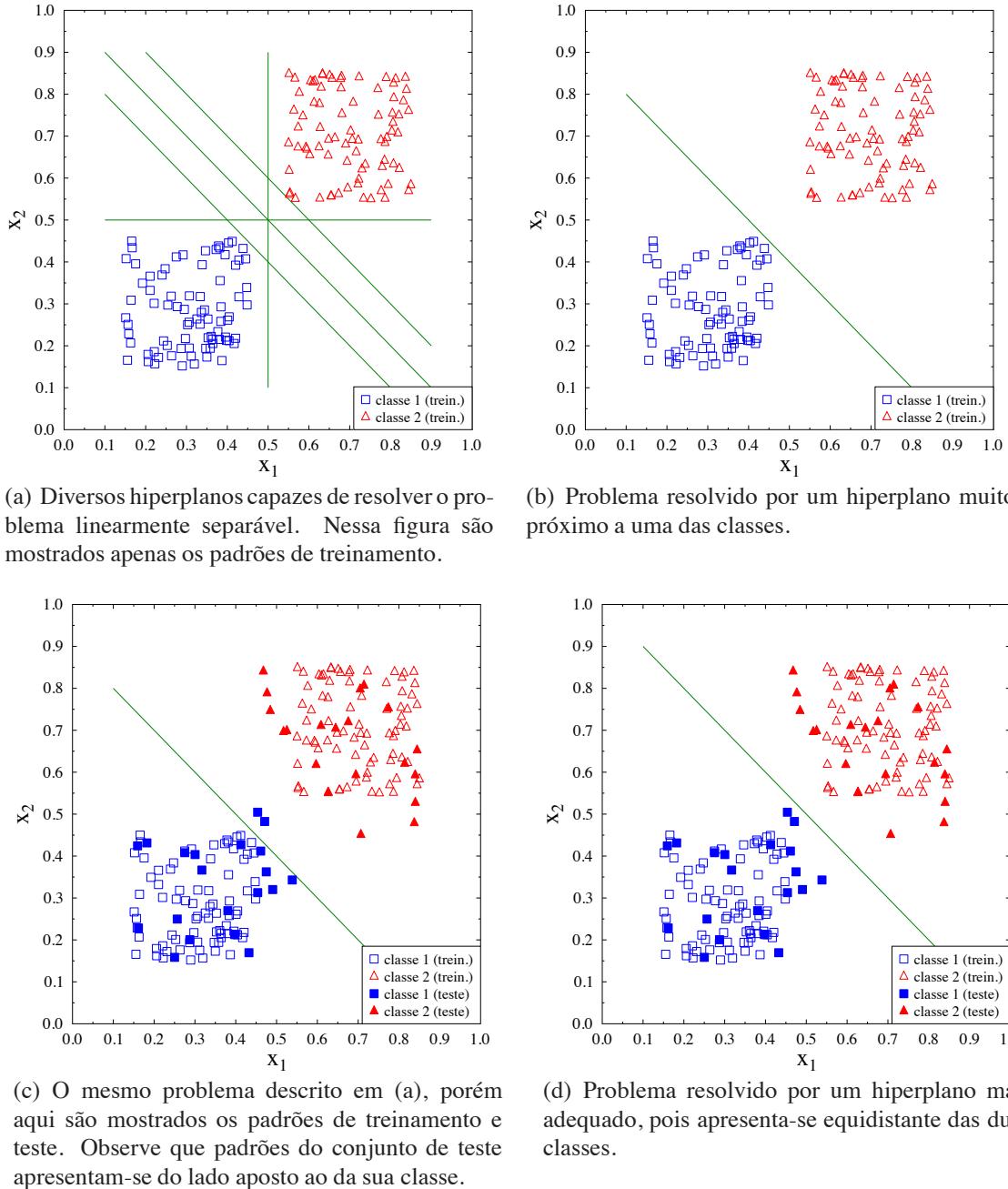


Figura 2.2: Um problema linearmente separável e hiperplanos solução do problema.

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 (2.8) com sinal de igualdade são denominados *vetores-suporte* (VS), ou seja

$$f(\mathbf{x}^{(s)}) = \begin{cases} -1, & \mathbf{w}_o^T \mathbf{x}^{(s)} + b_o = -1 \\ +1 & \mathbf{w}_o^T \mathbf{x}^{(s)} + b_o = +1. \end{cases} \quad (2.9)$$

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:

$$\mathbf{x} = \mathbf{x}_p + r \frac{\mathbf{w}_o}{\|\mathbf{w}_o\|}. \quad (2.10)$$

A distância r do ponto até o hiperplano pode ser obtida com base nas equações 2.5, 2.10 e no conhecimento do valor de f no ponto \mathbf{x}_p , $f(\mathbf{x}_p) = 0$, da seguinte forma:

$$\begin{aligned} f(\mathbf{x}) &= \mathbf{w}_o^T \mathbf{x} + b_o \\ f(\mathbf{x}) &= \mathbf{w}_o^T \left[\mathbf{x}_p + r \frac{\mathbf{w}_o}{\|\mathbf{w}_o\|} \right] + b_o \\ f(\mathbf{x}) &= \mathbf{w}_o^T \mathbf{x} + b_o + \mathbf{w}_o^T r \frac{\mathbf{w}_o}{\|\mathbf{w}_o\|} \\ f(\mathbf{x}) &= 0 + r \frac{\mathbf{w}_o^T \mathbf{w}_o}{\|\mathbf{w}_o\|} \\ f(\mathbf{x}) &= r \|\mathbf{w}_o\| \\ r &= \frac{f(\mathbf{x})}{\|\mathbf{w}_o\|} \end{aligned} \quad (2.11)$$

A distância $r^{(s)}$ dos VS ao hiperplano ótimo pode ser obtida a partir da Equação (2.11) e da Equação (2.9)¹, a saber

$$\begin{aligned} r^{(s)} &= \frac{f(\mathbf{x}^{(s)})}{\|\mathbf{w}_o\|} \\ r^{(s)} &= \frac{1}{\|\mathbf{w}_o\|} \end{aligned} \quad (2.12)$$

A interpretação geométrica da distância de um dado padrão \mathbf{x} ao hiperplano ótimo pode ser visualizada na Figura 2.3. 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 (2.12), tal que

$$\rho = 2r^{(s)} = \frac{2}{\|\mathbf{w}_o\|}. \quad (2.13)$$

A partir de Vapnik (1982, 1995) 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 (2.13) 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

¹A distância deve assumir valor positivo, logo o valor negativo da Equação 2.9 é desconsiderado.

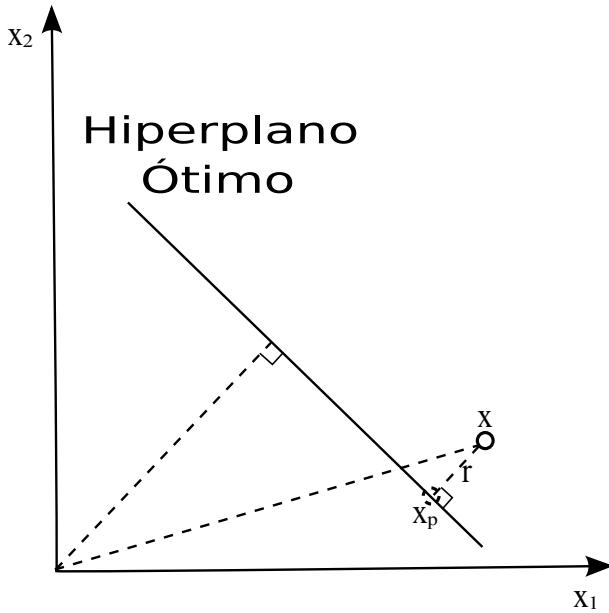


Figura 2.3: Interpretação geométrica da distância de um padrão \mathbf{x} ao hiperplano ótimo.

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 (2.13), com a maximização da margem de separação ρ :

$$\max \rho = \max \frac{2}{\|\mathbf{w}\|}, \quad (2.14)$$

que ainda pode ser apresentado como

$$\min \|\mathbf{w}\| \iff \min \sqrt{\mathbf{w}^T \mathbf{w}} \iff \min \frac{1}{2} \mathbf{w}^T \mathbf{w}, \quad (2.15)$$

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:

$$\tau(\mathbf{w}) = \frac{1}{2} \mathbf{w}^T \mathbf{w}. \quad (2.16)$$

Toda a discussão realizada até este ponto é válida tanto para classificadores SVM, quanto para classificadores LSSVM. Na Seção 2.3 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 2.6.

2.3 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

$$\tau(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{w}, \quad (2.17)$$

satisfazendo a restrição

$$d_i(\mathbf{w}^T \mathbf{x}_i + b) \geq +1, \quad (2.18)$$

a qual é obtida pela combinação das duas linhas da Equação (2.8).

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

$$\begin{aligned} \min \tau(\mathbf{w}) &= \min \frac{1}{2}\|\mathbf{w}\| = \min \frac{1}{2}\mathbf{w}^T\mathbf{w} \\ s.a. \quad d_i[(\mathbf{w}^T \mathbf{x}_i) + b] &\geq 1, \quad i = 1, \dots, n \end{aligned} \quad (2.19)$$

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

Conforme mencionado anteriormente o problema apresentado na Equação (2.19) 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.

2.3.1 Classificador SVM com Margem Rígida

O problema de otimização com restrição apresentado na Equação (2.19) 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:

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^n \alpha_i(d_i(\mathbf{x}_i^T \mathbf{w} + b) - 1), \quad (2.20)$$

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:

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^n \alpha_i d_i \mathbf{w}^T \mathbf{x}_i - b \sum_{i=1}^n \alpha_i d_i + \sum_{i=1}^n \alpha_i. \quad (2.21)$$

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

$$\frac{\partial L(\mathbf{w}, b, \alpha)}{\partial \mathbf{w}} = 0 \quad (2.22)$$

$$\frac{\partial L(\mathbf{w}, b, \alpha)}{\partial b} = 0 \quad (2.23)$$

que resultam em

$$\mathbf{w} = \sum_{i=1}^n \alpha_i d_i \mathbf{x}_i \quad (2.24)$$

e

$$\sum_{i=1}^n \alpha_i d_i = 0, \quad (2.25)$$

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 (2.21) é zero, devido ao resultado obtido na Equação (2.25). Desta forma, pode-se reescrever a Equação (2.21) da seguinte forma:

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2} \mathbf{w}^T \mathbf{w} - \sum_{i=1}^n \alpha_i d_i \mathbf{w}^T \mathbf{x}_i + \sum_{i=1}^n \alpha_i, \quad (2.26)$$

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 (2.21), quando aplicado o resultado obtido na Equação (2.24), equivale a $-\mathbf{w}^T \mathbf{w}$. Logo, pode-se mostrar que

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2} \mathbf{w}^T \mathbf{w} - \mathbf{w}^T \mathbf{w} + \sum_{i=1}^n \alpha_i, \quad (2.27)$$

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

ou ainda, substituindo a equação acima pelo resultado apresentado na Equação (2.24), obtém-se

$$L(\alpha) = -\sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T \mathbf{x}_j + \sum_{i=1}^n \alpha_i \quad (2.28)$$

ou

$$L(\alpha) = \sum_{i=1}^n \alpha_i - \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T \mathbf{x}_j. \quad (2.29)$$

A partir da análise da Equação (2.29) percebe-se que a mesma apresenta-se apenas em função dos multiplicadores de Lagrange $\{\alpha_i\}_{i=1}^n$.

O problema de otimização dual é formulado como

$$\begin{aligned} \max L(\alpha) &= \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T \mathbf{x}_j \\ \text{s.a. } & \sum_{i=1}^n \alpha_i d_i = 0 \\ \text{s.a. } & \alpha_i \geq 0 \text{ para } i = 1, \dots, n \end{aligned} \quad (2.30)$$

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

$$\mathbf{w}_o = \sum_{i=1}^N \alpha_i^o d_i \mathbf{x}_i \quad (2.31)$$

e

$$b_o = 1 - \mathbf{w}_o^T \mathbf{x}^{(s)}, \quad (2.32)$$

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 (2.5) para o hiperplano ótimo, pode ser reescrita para um função com base no problema dual, aplicando-se a Equação (2.31), resultando em

$$f(\mathbf{x}) = \sum_{i=1}^N \alpha_i^o d_i \mathbf{x}_i^T \mathbf{x} + b_o. \quad (2.33)$$

2.3.2 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

$$d_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i, i = 1, \dots, n, \quad (2.34)$$

em que $\xi_i \geq 0, i = 1, \dots, n$. 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 (CORTES; VAPNIK, 1995). 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

$$\begin{aligned} \min \tau(\mathbf{w}, \xi) &= \min \left\{ \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^n \xi_i \right\} \\ \text{s.a. } d_i[(\mathbf{w}^T \mathbf{x}_i) + b] &\geq 1 - \xi_i, \quad i = 1, \dots, n \\ \text{s.a. } \xi_i &\geq 0, i = 1, \dots, n. \end{aligned} \quad (2.35)$$

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 consequentemente 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 $L(\mathbf{w}, b, \xi, \alpha, \beta)$, a qual deve ser minimizada em relação a \mathbf{w} , b e ξ e maximizada em relação a α e β , sendo a mesma da seguinte

forma:

$$L(\mathbf{w}, b, \xi, \alpha, \beta) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^n \xi_i - \sum_{i=1}^n \alpha_i (d_i (\mathbf{x}_i^T \mathbf{w} + b) - 1 + \xi_i) - \sum_{i=1}^n \beta_i \xi_i, \quad (2.36)$$

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

$$L(\mathbf{w}, b, \xi, \alpha, \beta) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^n \xi_i - \sum_{i=1}^n \alpha_i d_i \mathbf{w}^T \mathbf{x}_i - b \sum_{i=1}^n \alpha_i d_i + \sum_{i=1}^n \alpha_i - \sum_{i=1}^n \alpha_i \xi_i - \sum_{i=1}^n \beta_i \xi_i. \quad (2.37)$$

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

$$\frac{\partial L(\mathbf{w}, b, \xi, \alpha)}{\partial \mathbf{w}} = 0 \quad (2.38)$$

$$\frac{\partial L(\mathbf{w}, b, \xi, \alpha)}{\partial b} = 0 \quad (2.39)$$

$$\frac{\partial L(\mathbf{w}, b, \xi, \alpha)}{\partial \xi} = 0 \quad (2.40)$$

A primeira condição, ao ser resolvida resulta em

$$\mathbf{w} = \sum_{i=1}^n \alpha_i d_i \mathbf{x}_i. \quad (2.41)$$

Enquanto a segunda, ao ser resolvida, resulta em

$$\sum_{i=1}^n \alpha_i d_i = 0. \quad (2.42)$$

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

$$C = \alpha_i + \beta_i. \quad (2.43)$$

Ao aplicar no segundo termo ($C \sum_{i=1}^n \xi_i$) da Equação (2.37) o resultado obtido na Equação (2.43), 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 (2.37) 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 (2.37) seja reescrita como:

$$L(\mathbf{w}, b, \xi, \alpha, \beta) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \sum_{i=1}^n \xi_i (\alpha_i + \beta_i) - \sum_{i=1}^n \alpha_i d_i \mathbf{w}^T \mathbf{x}_i - b \sum_{i=1}^n \alpha_i d_i + \sum_{i=1}^n \alpha_i - \sum_{i=1}^n \xi_i (\alpha_i + \beta_i). \quad (2.44)$$

Pode-se verificar que o segundo e o último termos da Equação (2.44) podem ser eliminados,

resultando em:

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2} \mathbf{w}^T \mathbf{w} - \sum_{i=1}^n \alpha_i d_i \mathbf{w}^T \mathbf{x}_i - b \sum_{i=1}^n \alpha_i d_i + \sum_{i=1}^n \alpha_i. \quad (2.45)$$

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 (2.45) e (2.21). 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 (2.45) é zero, devido ao resultado obtido na Equação (2.42). Então a Equação (2.45) pode ser reescrita da seguinte forma:

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2} \mathbf{w}^T \mathbf{w} - \sum_{i=1}^n \alpha_i d_i \mathbf{w}^T \mathbf{x}_i + \sum_{i=1}^n \alpha_i, \quad (2.46)$$

Nota-se também que ao aplicar o resultado obtido na Equação (2.41) ao segundo termo ($-\sum_{i=1}^n \alpha_i d_i \mathbf{w}^T \mathbf{x}_i$) da Equação (2.46), obtém-se $-\mathbf{w}^T \mathbf{w}$. Logo, tem-se que:

$$\begin{aligned} L(\mathbf{w}, b, \alpha) &= \frac{1}{2} \mathbf{w}^T \mathbf{w} - \mathbf{w}^T \mathbf{w} + \sum_{i=1}^n \alpha_i, \\ L(\mathbf{w}, b, \alpha) &= -\frac{1}{2} \mathbf{w}^T \mathbf{w} + \sum_{i=1}^n \alpha_i. \end{aligned} \quad (2.47)$$

Substituindo a equação acima pelo resultado apresentado na Equação (2.41), a fim de que a função não dependa de \mathbf{w} , obtém-se então

$$L(\alpha) = - \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T \mathbf{x}_j + \sum_{i=1}^n \alpha_i, \quad (2.48)$$

ou

$$L(\alpha) = \sum_{i=1}^n \alpha_i - \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T \mathbf{x}_j. \quad (2.49)$$

Por consequência, a Equação (2.49) 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

$$\begin{aligned} \max L(\alpha) &= \max \left\{ \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T \mathbf{x}_j \right\} \\ &\text{s.a. } \sum_{i=1}^n \alpha_i d_i = 0 \\ &\text{s.a. } 0 \leq \alpha_i \leq C \text{ para } i = 1, \dots, n \end{aligned} \quad (2.50)$$

Ao analisar a Equação (2.50) pode-se perceber apenas uma diferença em relação à Equação (2.30) 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 (2.43). 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 (2.4), que traz um problema de classificação binário hipotético, de natureza não-linearmente separável.

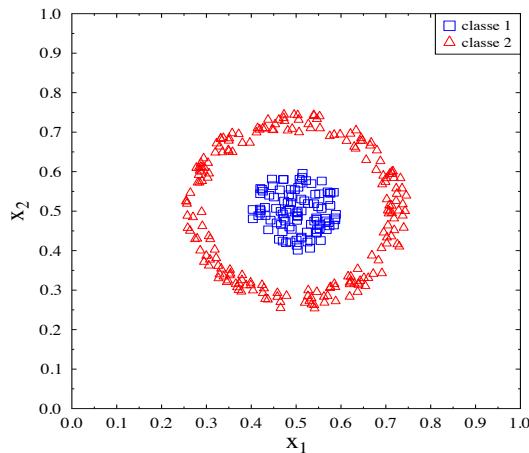


Figura 2.4: Exemplo de problema binário em que um hiperplano não resolve de forma satisfatória, mesmo considerando a possibilidade de alguns erros.

2.4 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 (2.4). 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 (MERCER, 1909), há um espaço de Hilbert H , um mapeamento $\phi : \mathbb{R}^N \rightarrow H$ e um

número $\beta_i > 0$, tal que

$$K(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^M \beta_i \tilde{\phi}_i(\mathbf{x}) \tilde{\phi}_i(\mathbf{y}), \quad (2.51)$$

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 $g(\cdot)$, tal que $g(\cdot) \neq 0$,

$$\int K(\mathbf{x}, \mathbf{y}) g(\mathbf{x}) g(\mathbf{y}) d\mathbf{x} d\mathbf{y} \geq 0. \quad (2.52)$$

Por conseguinte, ao se definir $\phi_i(\cdot) = \tilde{\phi}_i(\cdot) \sqrt{\beta_i}$ pode-se escrever a Equação (2.51) como

$$K(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^M \tilde{\phi}_i(\mathbf{x}) \sqrt{\beta_i} \tilde{\phi}_i(\mathbf{y}) \sqrt{\beta_i}, \quad (2.53)$$

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

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

$$K(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^M \phi_i(\mathbf{x}) \phi_i(\mathbf{y}) \quad (2.54)$$

$$K(\mathbf{x}, \mathbf{y}) = \langle \phi_i(\mathbf{x}), \phi_i(\mathbf{y}) \rangle \quad (2.55)$$

$$K(\mathbf{x}, \mathbf{y}) = \phi_i^T(\mathbf{x}) \phi_i(\mathbf{y}). \quad (2.56)$$

A Equação (2.56) 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 $\phi_i(\mathbf{x})$, desde que se conheça uma função que descreva o produto interno $\langle \phi_i(\mathbf{x}), \phi_i(\mathbf{y}) \rangle$. Exemplos destas funções (de *kernel*) são apresentadas na Tabela 2.1. 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 2.5. Esta figura é derivada de outra apresentada em Valyon (2007).

Nesse sentido, pode-se redefinir as Equações (2.30) e (2.50) referentes ao classificador SVM com margem rígida e ao com margem flexível para

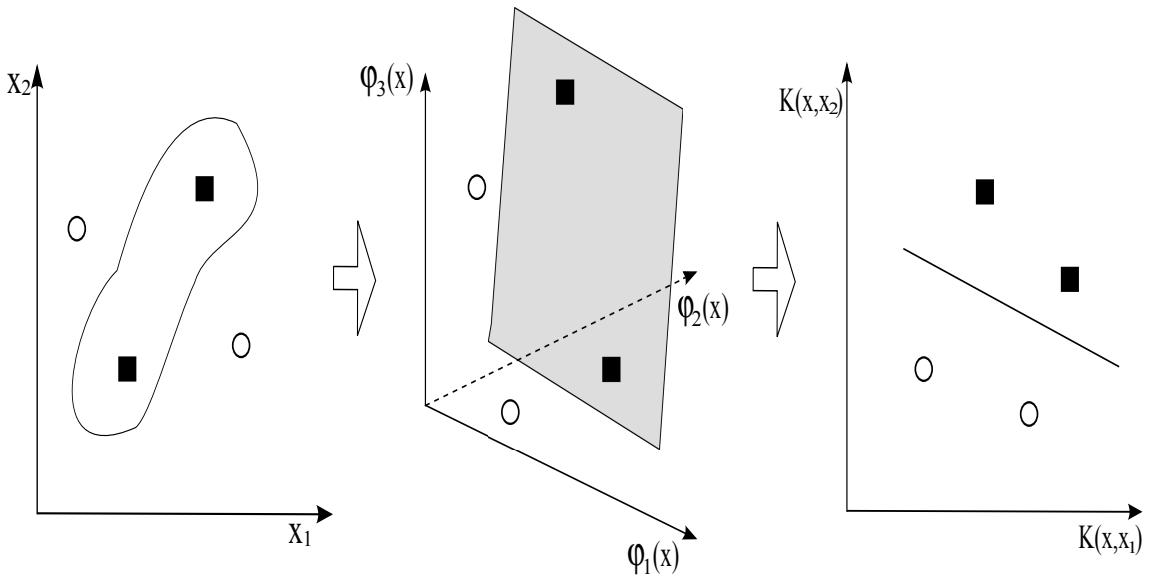


Figura 2.5: Espaço de entrada no \mathbb{R}^2 , Espaço de Características no \mathbb{R}^3 e Espaço Kernel.

$$\begin{aligned} \max L(\alpha) = \max & \left\{ \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j d_i d_j K(\mathbf{x}_i, \mathbf{x}_j) \right\} \\ & s.a. \quad \sum_{i=1}^n \alpha_i d_i = 0 \\ & s.a. \quad \alpha_i \geq 0 \quad para \quad i = 1, \dots, n \end{aligned} \quad (2.57)$$

e

$$\begin{aligned} \max L(\alpha) = \max & \left\{ \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j d_i d_j K(\mathbf{x}_i, \mathbf{x}_j) \right\} \\ & s.a. \quad \sum_{i=1}^n \alpha_i d_i = 0 \\ & s.a. \quad 0 \leq \alpha_i \leq C \quad para \quad i = 1, \dots, n. \end{aligned} \quad (2.58)$$

Similarmente, a Equação (2.33), que descreve a função discriminante no espaço de características, pode ser reescrita como

$$f(\mathbf{x}) = \sum_{i=1}^n \alpha_i^o d_i K(\mathbf{x}_i, \mathbf{x}) + b_o. \quad (2.59)$$

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

$$f(\mathbf{x}) = sign \left(\sum_{i=1}^n \alpha_i^o d_i K(\mathbf{x}_i, \mathbf{x}) + b_o \right), \quad (2.60)$$

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*) (AYAT et al., 2002). Estes *kernels* são apresentados na Tabela 2.1, enquanto o KMOD é detalhado na subseção seguinte.

Kernel	Descrição
Linear	$K(\mathbf{x}, \mathbf{x}_i) = \mathbf{x}_i^T \cdot \mathbf{x}$
Polinomial	$K(\mathbf{x}, \mathbf{x}_i) = (\mathbf{x}_i^T \cdot \mathbf{x} + 1)^d$, 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 2.1: Listagem de importantes *kernels*.

2.4.1 Kernel with MODerate decreasing (KMOD)

O *kernel* KMOD (AYAT et al., 2002) 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.

Embora o *kernel* KMOD não seja tão utilizado quanto os *kernels* descritos anteriormente, este apresenta-se bastante importante, pois, os melhores resultados para classificação de patologias da coluna vertebral foram obtidos a partir de sua utilização, tanto na avaliação de classificadores SVM e LSSVM, quanto na avaliação de comitês de classificadores (veja Capítulo 5).

O *kernel* KMOD é descrito por

$$K(\mathbf{x}, \mathbf{x}_i) = a \cdot \left[\exp\left\{ \frac{\gamma}{(||\mathbf{x} - \mathbf{x}_i||^2 + \sigma^2)} \right\} - 1 \right], \quad (2.61)$$

em que a é uma constante de normalização, tal que

$$a = \frac{1}{\exp\left(\frac{\gamma}{\sigma^2}\right) - 1}, \quad (2.62)$$

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

$$K(\mathbf{x}, \mathbf{x}_i) = K(||\mathbf{x} - \mathbf{x}_i||^2). \quad (2.63)$$

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 2.6. Para fins de comparação, nesta figura apresenta-se ainda o comportamento do *kernel* RBF.

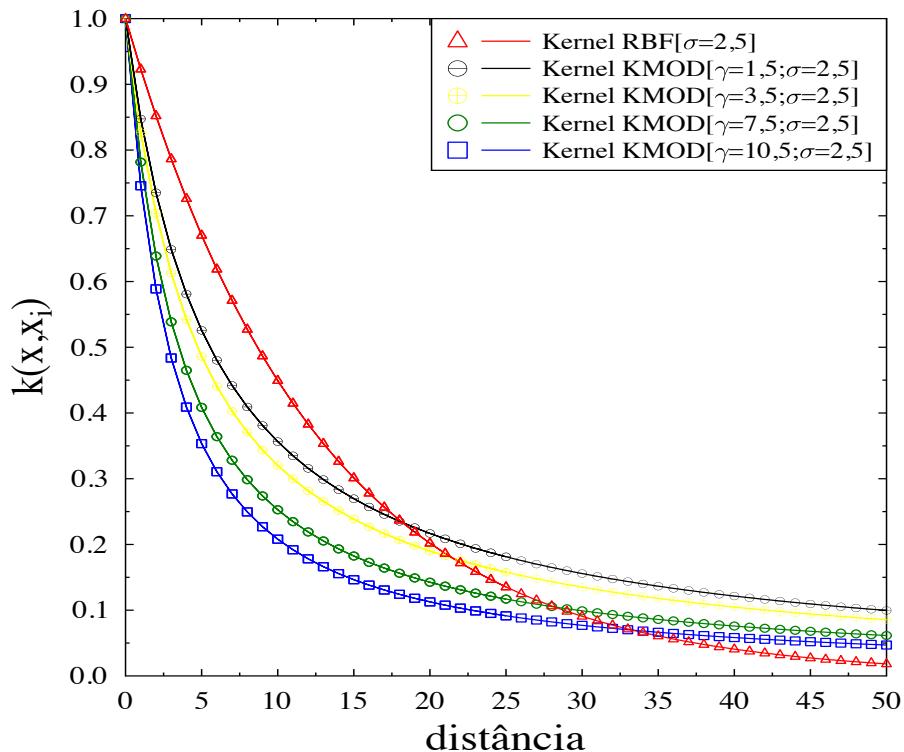


Figura 2.6: Comportamento dos *kernels* RBF e KMOD.

2.5 Obtenção dos Parâmetros Ótimos para o Classificador SVM

2.5.1 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 Octave².

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

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

na Equação (2.58) pode ser apresentado como

$$\min L(\alpha) = \frac{1}{2} \alpha^T Q \alpha - \alpha^T \mathbf{1} \quad (2.64)$$

sujeito a

$$\begin{aligned} \alpha^T \mathbf{d} &= 0 \\ 0 \leq \alpha_i &\leq C \quad i = 1 \dots n, \end{aligned} \quad (2.65)$$

em que Q é uma matriz $n \times n$, sendo seus elementos descritos por $q_{i,j} = d_i d_j K(\mathbf{x}_i, \mathbf{x}_j)$ tal que $i, j = 1 \dots n$. 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 (KAUFMAN, 1999). Em virtude disto, um método *online* pode ser útil, pois este não exige uma grande quantidade de memória.

2.5.2 Solução baseada no Algoritmo SMO

Sequential Minimal Optimization (SMO) é um algoritmo iterativo para solucionar o problema de otimização dual dos classificadores SVM (PLATT, 1999). O algoritmo seleciona um par de parâmetros, α_p e α_q , do conjunto de multiplicadores de Lagrange, $\{\alpha_i\}_{i=1}^l$, e optimiza 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 optimizar conjuntamente;
 - (b) Se não for capaz de encontrar multiplicadores a optimizar; 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. Nesta tese de doutorado, o algoritmo implementado é descrito mais completamente no trabalho de Platt (1999). Neste trabalho, Platt apresenta muito mais propriedades do algoritmo SMO, tanto no que se refere à sua eficiência quanto à estabilidade.

2.5.3 Solução baseada no Kernel Adatron

Adatron (ANLAUF; BIEHL, 1989) 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 primeiramente descrita no trabalho de Campbell & Cristianini (1998). Outro trabalho correlato relacionado ao desenvolvimento do Kernel Adatron é o trabalho de Frieß et al. (1998). Uma representação do *Kernel Adatron* em uma topologia similar a de redes neurais pode ser visualizada na Figura (2.7).

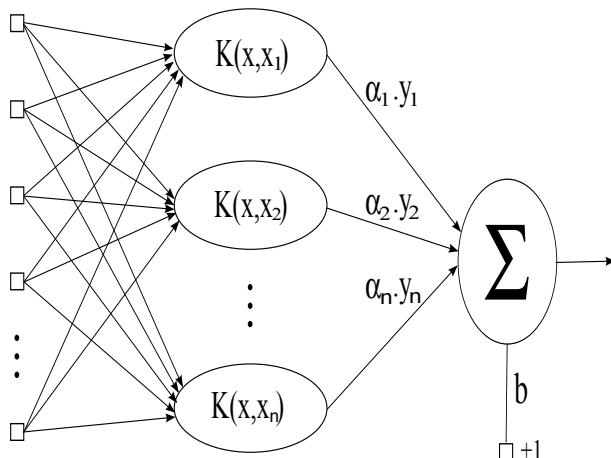


Figura 2.7: *Kernel Adatron* representado de forma similar às redes neurais artificiais.

A formulação do problema baseada no Kernel Adatron, pode ser descrita com base em um

classificador SVM com margem flexível, como descrita na Equação (2.58), ou seja

$$\begin{aligned} \max L(\alpha) = & \max \left\{ \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j d_i d_j K(\mathbf{x}_i, \mathbf{x}_j) \right\} \\ & s.a. \quad \sum_{i=1}^n \alpha_i d_i = 0 \\ & s.a. \quad 0 \leq \alpha_i \leq C \quad \text{para } i = 1, \dots, n. \end{aligned} \quad (2.66)$$

Considerando apenas informação de primeira ordem, pode-se determinar o gradiente da função-custo apresentada na Equação (2.66), 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

$$\frac{\partial L}{\partial \alpha_i} = 1 - d_i \left(\sum_{i=1}^n \alpha_i d_i K(\mathbf{x}_i, \mathbf{x}_j) \right). \quad (2.67)$$

A partir deste resultado, pode-se reescrever a Equação (2.67) 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 (2.59), tal que

$$\frac{\partial L}{\partial \alpha_i} = 1 - d_i f(\mathbf{x}). \quad (2.68)$$

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

$$\begin{aligned} \Delta \alpha_i &= \alpha_i(t+1) - \alpha_i(t) \\ \alpha_i(t+1) &= \alpha_i(t) + \Delta \alpha_i \\ \alpha_i(t+1) &= \alpha_i(t) + \mu \frac{\partial L}{\partial \alpha_i} \\ \alpha_i(t+1) &= \alpha_i(t) + \mu (1 - d_i f(\mathbf{x})) \end{aligned} \quad (2.69)$$

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:

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

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

$$b = \frac{\max f^-(\mathbf{x}_i) + \min f^+(\mathbf{x}_i)}{2} \quad (2.70)$$

Outro critério de parada que pode ser utilizado em conjunto com o número de épocas de treinamento³, refere-se à verificação da margem que pode ser expressa como:

$$\kappa = \frac{\min f^+(\mathbf{x}_i) - \max f^-(\mathbf{x}_i)}{2}, \quad (2.71)$$

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

2.6 Fundamentos Teóricos do Classificador LSSVM

Classificadores LSSVM (SUYKENS; VANDEWALLE, 1999a) também são capazes de resolver problemas de classificação de padrões e problemas de aproximação de funções⁴. Em sua formulação original, classificadores LSSVM são projetados para resolver problemas binários, mas também podem ser combinados para resolver problemas multiclasses (SUYKENS; VANDEWALLE, 1999c).

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 igualdade⁵. 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 consequência 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

³O que ocorrer primeiro, por exemplo.

⁴Porém esta tese 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.

⁵Enquanto nos classificadores SVM a restrição do problema é de desigualdade, veja a Equação (2.18).

sistema Karush-Khun-Tucker (KKT) (FLETCHER, 1987).

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

$$\begin{aligned} \min \tau(\mathbf{w}, \xi) &= \frac{1}{2} \mathbf{w}^T \mathbf{w} + \gamma \frac{1}{2} \sum_{i=1}^n \xi_i^2 \\ s.a. \quad d_i[(\mathbf{w}^T \mathbf{x}_i) + b] &= 1 - \xi_i, \quad i = 1, \dots, n \end{aligned} \quad (2.72)$$

em que $\tau(\mathbf{w}, \xi)$ 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-nulos⁶. 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.

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

$$L(\mathbf{w}, b, \xi, \alpha) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \gamma \frac{1}{2} \sum_{i=1}^n \xi_i^2 - \sum_{i=1}^n \alpha_i (d_i(\mathbf{x}_i^T \mathbf{w} + b) - 1 + \xi_i), \quad (2.73)$$

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

$$\begin{aligned} \frac{\partial L(\mathbf{w}, b, \xi, \alpha)}{\partial \mathbf{w}} = 0 &\Rightarrow \mathbf{w} = \sum_{i=1}^n \alpha_i d_i \mathbf{x}_i \\ \frac{\partial L(\mathbf{w}, b, \xi, \alpha)}{\partial b} = 0 &\Rightarrow \sum_{i=1}^n \alpha_i d_i = 0 \\ \frac{\partial L(\mathbf{w}, b, \xi, \alpha)}{\partial \alpha_i} = 0 &\Rightarrow d_i(\mathbf{x}_i^T \mathbf{w} + b) - 1 + \xi_i = 0 \\ \frac{\partial L(\mathbf{w}, b, \xi, \alpha)}{\partial \xi_i} = 0 &\Rightarrow \alpha_i = \gamma \xi_i \end{aligned} \quad (2.74)$$

⁶Eventualmente um valor pode assumir valor nulo.

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:

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

em que $\Omega_{i,j} = y_i y_j \mathbf{x}_i^T \mathbf{x}_j$, $\mathbf{d} = [d_1 \ d_2 \ \dots \ d_n]^T$, $\alpha = [\alpha_1 \ \alpha_2 \ \dots \ \alpha_n]^T$ e $\mathbf{1} = [1 \ 1 \ \dots \ 1]^T$, 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

$$f(\mathbf{x}) = \text{sign} \left(\sum_{i=1}^l \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) + b \right). \quad (2.75)$$

2.7 Obtenção dos Parâmetros Ótimos para o classificador LSSVM

2.7.1 Solução Baseada na Matriz Inversa

A solução do classificador LSSVM, que consiste na obtenção de b e α contida no vetor $\mathbf{x}^T = [b \ \alpha^T]$, 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

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

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

2.7.2 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 (MOORE, 1920; PENROSE, 1955). Neste caso, a solução obtida equivale a considerar cada uma das colunas excluídas como possuindo multiplicador de Lagrange associado com valor zero (LEE; MANGASARIAN, 2001a). 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

$$\mathbf{x} = \mathbf{A}^* \mathbf{B}, \quad (2.76)$$

em que

$$\mathbf{A}^* = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T. \quad (2.77)$$

2.7.3 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 (RANGANATHAN, 2004).

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

$$\nabla L(\mathbf{x}) = \nabla L(\mathbf{x}_0) + (\mathbf{x} - \mathbf{x}_0)^T \nabla L^2(\mathbf{x}_0) + \dots \quad (2.78)$$

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

$$\nabla L(\mathbf{x}) = 0. \quad (2.79)$$

E aproximando $\nabla L(\mathbf{x})$ pelos termos até segunda ordem e desprezando os restantes⁷, na Equação (2.78), bem como considerando $(\mathbf{x} - \mathbf{x}_0) = \Delta \mathbf{x}$, tem-se:

$$\begin{aligned} \nabla L(\mathbf{x}_0) + (\mathbf{x} - \mathbf{x}_0)^T \nabla L^2(\mathbf{x}_0) &= 0 \\ (\mathbf{x} - \mathbf{x}_0)^T \nabla L^2(\mathbf{x}_0) &= -\nabla L(\mathbf{x}_0) \\ (\mathbf{x} - \mathbf{x}_0)^T [\nabla L^2(\mathbf{x}_0)] [\nabla L^2(\mathbf{x}_0)]^{-1} &= -\nabla L(\mathbf{x}_0) [\nabla L^2(\mathbf{x}_0)]^{-1} \\ (\mathbf{x} - \mathbf{x}_0)^T &= -\nabla L(\mathbf{x}_0) [\nabla L^2(\mathbf{x}_0)]^{-1} \\ (\mathbf{x} - \mathbf{x}_0) &= -[\nabla L^2(\mathbf{x}_0)]^{-1} \nabla L(\mathbf{x}_0) \\ \Delta \mathbf{x} &= -[\nabla L^2(\mathbf{x}_0)]^{-1} \nabla L(\mathbf{x}_0). \end{aligned} \quad (2.80)$$

⁷Nesta situação considera-se uma aproximação quadrática.

De uma forma mais geral, pode-se considerar:

$$\Delta \mathbf{x} = -[\nabla L^2(\mathbf{x})]^{-1} \nabla L(\mathbf{x}). \quad (2.81)$$

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

$$L(\mathbf{x}) = \frac{1}{2} \sum_{i=1}^n e_i^2(\mathbf{x}) = \frac{1}{2} \mathbf{e}^T(\mathbf{x}) \mathbf{e}(\mathbf{x}) \quad (2.82)$$

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

$$\nabla L(\mathbf{x}) = \mathbf{J}(\mathbf{x}) \mathbf{e}(\mathbf{x}) \quad (2.83)$$

$$\nabla L^2(\mathbf{x}) = \mathbf{J}^T(\mathbf{x}) \mathbf{J}(\mathbf{x}) + S(\mathbf{x}) \quad (2.84)$$

em que

$$S(\mathbf{x}) = \sum_{i=1}^l e_i(\mathbf{x}) \nabla^2 e_i(\mathbf{x}). \quad (2.85)$$

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

$$\Delta \mathbf{x} = -[\mathbf{J}^T(\mathbf{x}) \mathbf{J}(\mathbf{x}) + S(\mathbf{x})]^{-1} \mathbf{J}(\mathbf{x}) \mathbf{e}(\mathbf{x}) \quad (2.86)$$

passa a ser dada por

$$\Delta \mathbf{x} = -[\mathbf{J}^T(\mathbf{x}) \mathbf{J}(\mathbf{x})]^{-1} \mathbf{J}(\mathbf{x}) \mathbf{e}(\mathbf{x}). \quad (2.87)$$

O problema com a Equação (2.87) 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, Levenberg (1944) 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:

$$\Delta \mathbf{x} = -[\mathbf{J}^T(\mathbf{x}) \mathbf{J}(\mathbf{x}) + \mu \mathbf{I}]^{-1} \mathbf{J}(\mathbf{x}) \mathbf{e}(\mathbf{x}) \quad (2.88)$$

Vale ressaltar que a matriz hessiana aproximada $[\mathbf{J}^T(\mathbf{x}) \mathbf{J}(\mathbf{x}) + \mu \mathbf{I}]$ sempre possui inversa (RANGANATHAN, 2004).

Após uma atualização, se o valor da função-custo diminuir⁸, 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, Marquardt (1963) propôs substituir a matriz identidade pela matriz diagonal da matriz hessiana aproximada. Esta alteração resulta na seguinte regra de atualização:

$$\Delta \mathbf{x} = -[\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x}) + \mu \text{diag}[\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x})]]^{-1}\mathbf{J}(\mathbf{x})\mathbf{e}(\mathbf{x}). \quad (2.89)$$

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

$$\mathbf{Ax} = \mathbf{B}, \quad (2.90)$$

ou

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

em que $\Omega_{i,j} = y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$, tal que $i, j = 1, \dots, n$. 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

$$\mathbf{Ax} = \hat{\mathbf{Y}}(\mathbf{x}), \quad (2.91)$$

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

$$\begin{aligned} \mathbf{e}(\mathbf{x}) &= \mathbf{B} - \hat{\mathbf{Y}}(\mathbf{x}), \\ \mathbf{e}(\mathbf{x}) &= \mathbf{B} - \mathbf{Ax}. \end{aligned} \quad (2.92)$$

Para o método Levenberg-Maquardt deve-se considerar um lote de entradas aplicadas ao vetor \mathbf{x} e, por consequência, um lote de erros associados. Assim, pode-se considerar cada vetor

⁸Em um problema de minimização.

\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 derivadas parciais da função de erro, tal que

$$\mathbf{J}(\mathbf{x}) = \frac{\partial \mathbf{e}(\mathbf{x})}{\partial \mathbf{x}} = -\mathbf{A}. \quad (2.93)$$

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:

$$\begin{aligned} \mathbf{x}_{t+1} &= \mathbf{x}_t - [\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x}) + \mu \operatorname{diag}(\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x}))]^{-1} \mathbf{J}^T(\mathbf{x})\mathbf{e}(\mathbf{x}), \\ &= \mathbf{x}_t + [\mathbf{A}^T\mathbf{A} + \mu \operatorname{diag}(\mathbf{A}^T\mathbf{A})]^{-1} \mathbf{A}^T\mathbf{e}(\mathbf{x}), \end{aligned} \quad (2.94)$$

em que t representa a iteração do método e a matriz $[\mu \operatorname{diag}(\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x}))]$ descreve um termo de regularização. Como dito anteriormente, a matriz $[\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x}) + \mu \operatorname{diag}(\mathbf{J}^T(\mathbf{x})\mathbf{J}(\mathbf{x}))]$ é não singular (RANGANATHAN, 2004). 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 (2.94);

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].

2.8 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 A e em (ROCHA-NETO, 2006). 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 2.2 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 2.2: 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 2.8.

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 SVM/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*

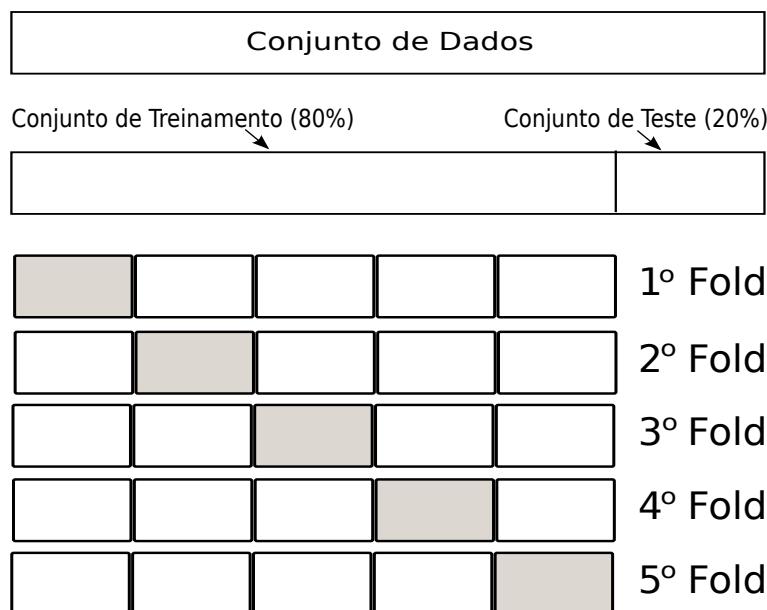


Figura 2.8: Esquema descrevendo um processo de separação do conjunto de dados entre conjunto de treinamento e teste. O conjunto de treinamento é ainda submetido a uma etapa de validação cruzada de 5 partes para obtenção dos parâmetros ótimos dos classificadores SVM e LSSVM.

utilizadas são apresentados na Tabela 2.3.

Nomenclatura	Classificador	Método Treinamento	Kernel
SVM/ADA/LIN	SVM	<i>K.</i> Adatron	linear
SVM/ADA/RBF	SVM	<i>K.</i> Adatron	RBF
SVM/ADA/KMOD	SVM	<i>K.</i> 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 2.3: Nomenclatura para os classificadores SVM e LSSVM.

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

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

Na Tabela 2.4 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 2.4, 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 2.4: Resultados dos classificadores SVM para o problema binário da coluna vertebral.

Na Figura 2.9 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 2.4. 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 2.9, 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 2.10. Esta curva descreve a acurácia em função dos parâmetros σ e C para o classificador SVM/SMO/RBF.

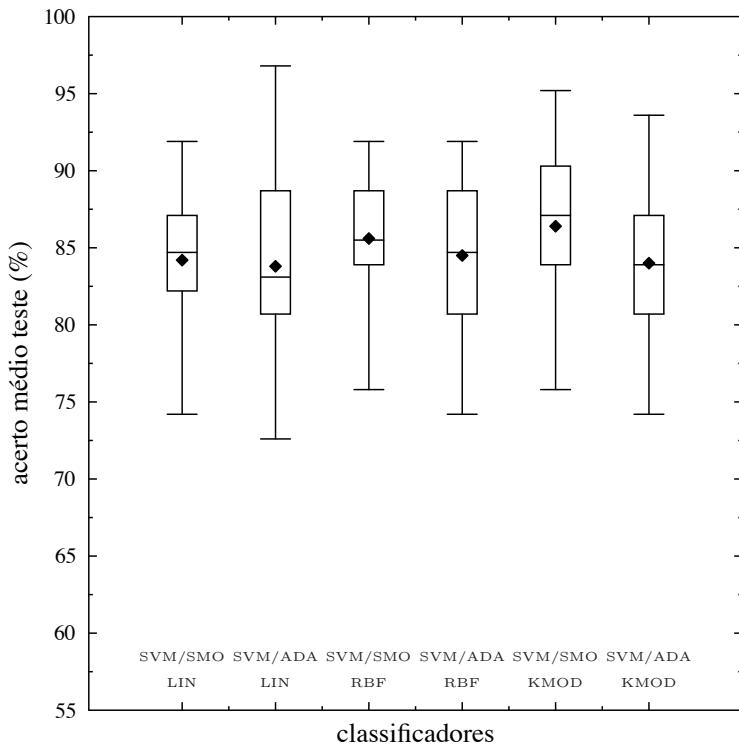


Figura 2.9: Diagramas de caixa contendo os valores obtidos nas 50 rodadas referentes ao problema binário com uso de classificadores SVM.

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%.

De forma similar, na Tabela 2.5 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/RBF⁹ 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.

Na Figura 2.11 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 2.5. Pela análise destes diagramas, também nota-se que há um desempenho semelhante entre os classi-

⁹O termo LSSVM/LM denota o classificador LSSVM treinado pelo algoritmo Levenberg-Maquardt, conforme proposto na Subseção 2.7.3.

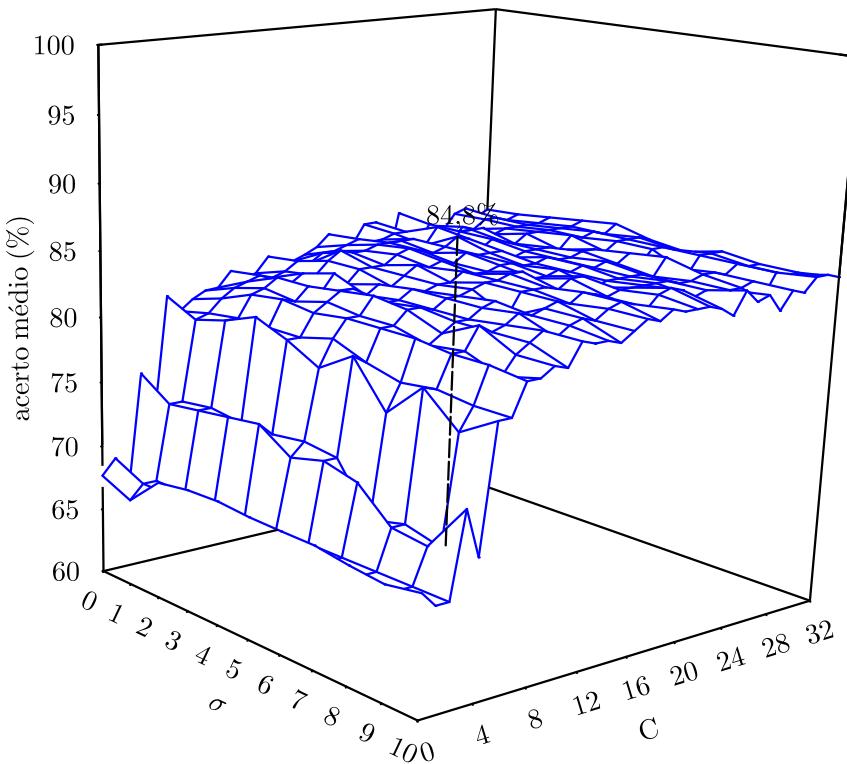


Figura 2.10: Superfície resultante do processo de busca em grade, via validação cruzada de 5 partes, pelo melhor conjunto de parâmetros em uma das rodadas para o classificador SVM/SMO/RBF.

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

ficadores 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).

A Figura 2.12 contém as curvas ROC (*Receiver Operating Characteristic*) obtidas para os classificadores LSSVM/LM/RBF, SVM/SMO/KMOD e SVM/SMO/RBF. Mais detalhes sobre

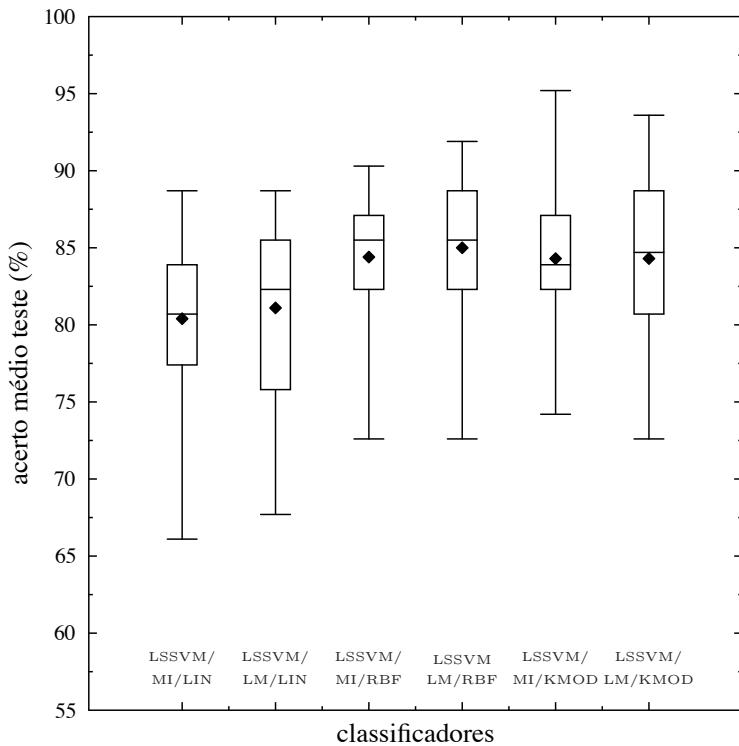


Figura 2.11: Diagrama de caixa contendo os valores da acurácia obtidos nas 50 rodadas referentes à aplicação do classificador LSSVM ao problema binário.

curvas ROC podem ser obtidos em Prati et al. (2008). Ao se analisar os valores 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.

Na Figura 2.13, mostra-se um gráfico da acurácia em função do limiar de decisão (*threshold*) contendo três curvas, uma curva para cada um dos classificadores avaliados na Figura 2.12. 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 diferentes¹⁰, 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 2.14, é apresentada a superfície de decisão obtida para o classificador SVM/SMO/KMOD.

¹⁰O valor típico para o limiar é zero.

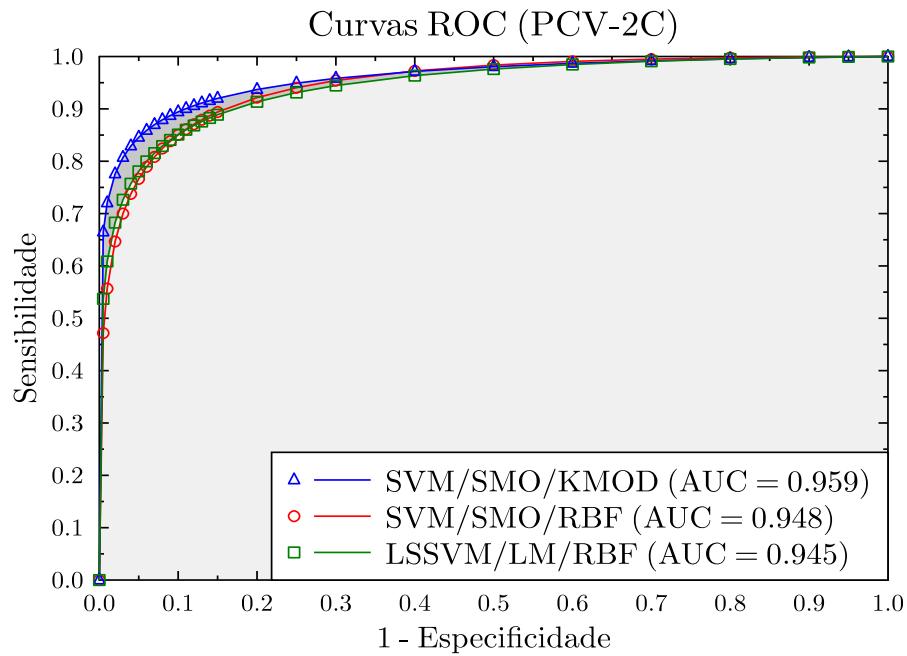


Figura 2.12: Curvas ROC para o problema da coluna vertebral com 2 classes, para os classificadores LSSVM/LM/RBF, SVM/SMO/KMOD e SVM/SMO/RBF, bem como os valores AUC correspondentes.

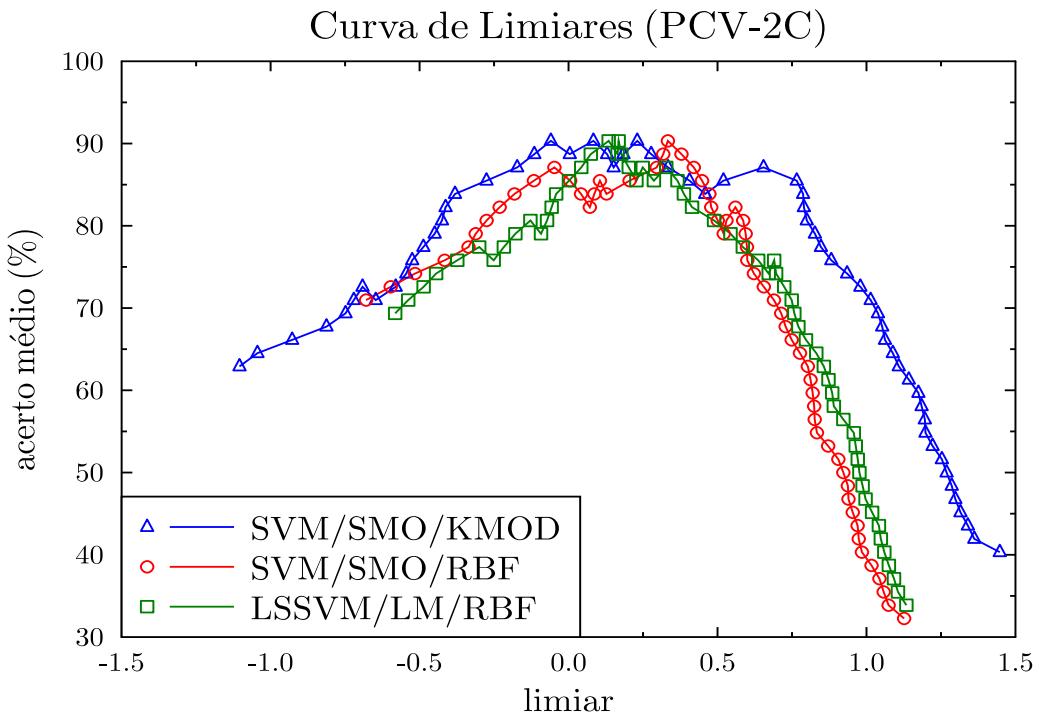


Figura 2.13: Gráfico com o desempenho dos classificadores LSSVM/LM/RBF, SVM/SMO/KMOD e SVM/SMO/RBF em função do limiar de decisão.

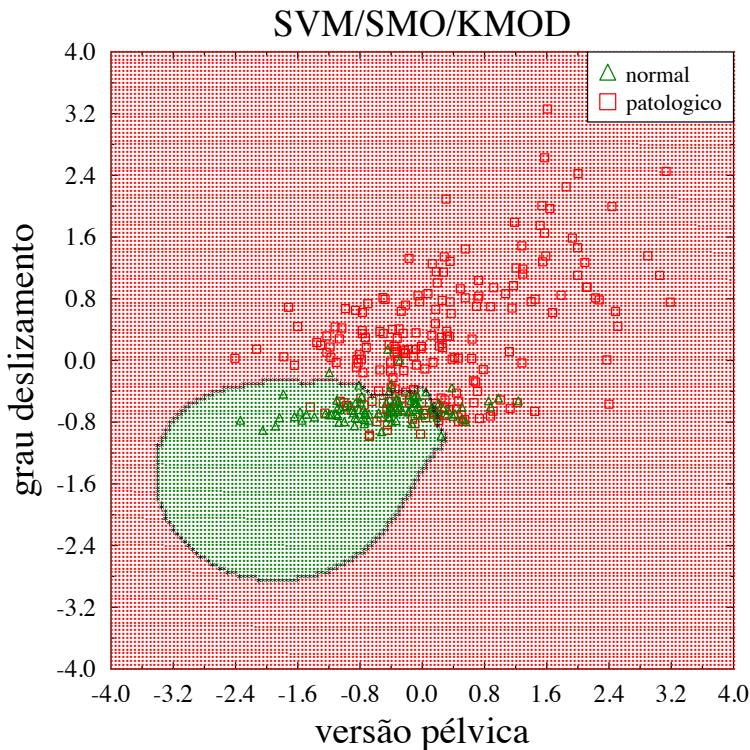


Figura 2.14: Superfície de decisão obtida a partir do classificador SVM/SMO/KMOD para o problema binário da coluna vertebral.

2.8.2 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 (DUAN; KEERTHI, 2005). 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 agrupa 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 2.6 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 2.6, 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 2.15, 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 2.6: Resultados do classificador SVM para o problema da coluna vertebral com 3 classes.

A Tabela 2.7 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 2.6. 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.

Na Figura 2.16 são mostrados os diagramas de caixa referentes aos classificadores e resultados apresentados na Tabela 2.7. 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 obtidos para os classificadores LSSVM/MI/KMOD e LSSVM/LM/KMOD. Percebe-se ainda que o classificador LSSVM/LM/RBF apresenta o maior acerto e a maior dispersão dentre os classificadores avaliados.

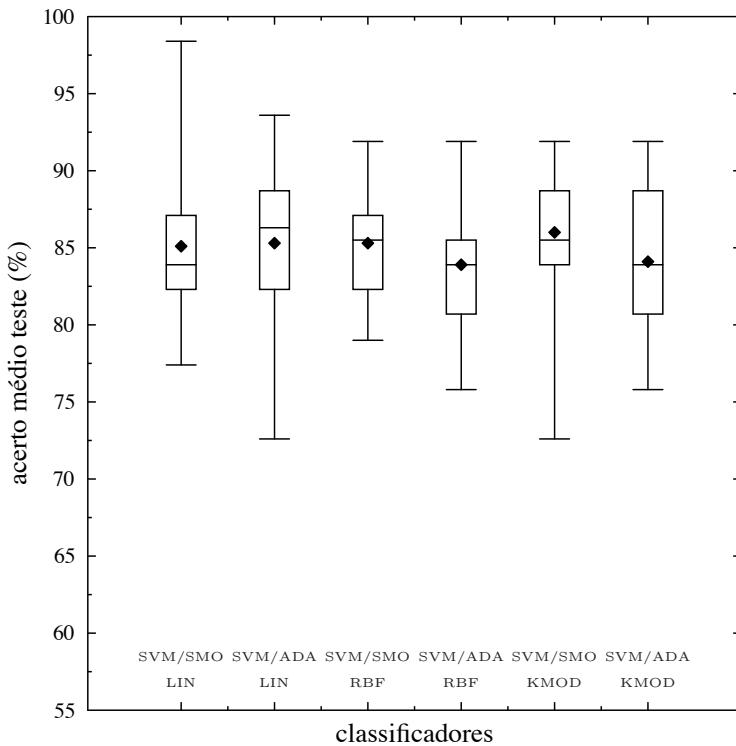


Figura 2.15: Diagramas de caixa com os resultados obtidos para os classificadores SVM quando aplicados ao problema PCV-3C.

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

Para fins de comparação, na Tabela 2.8 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 Rocha-Neto (2006). 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.

¹¹Utiliza-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.

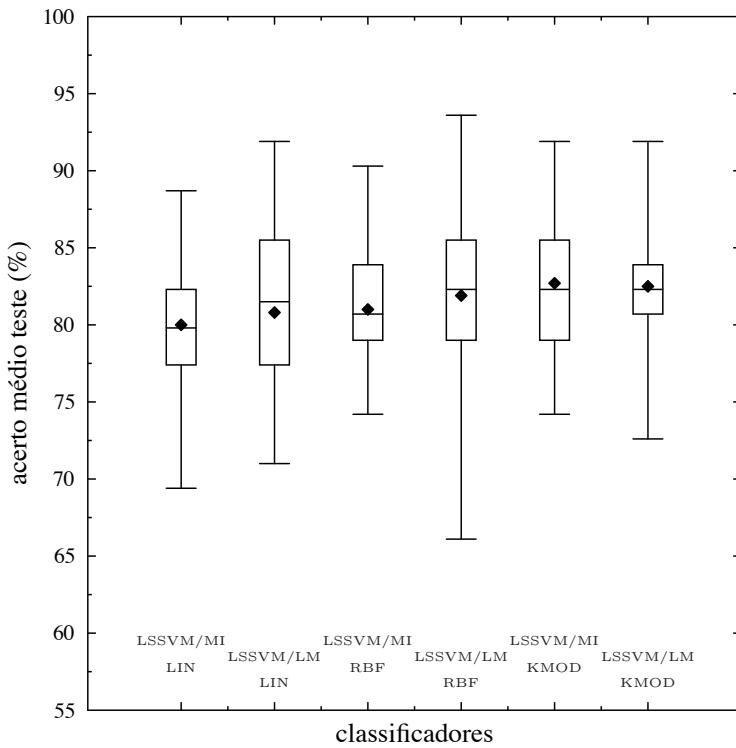


Figura 2.16: Diagramas de caixa com os resultados obtidos para os classificadores LSSVM para o problema PCV-3C.

Na Tabela 2.8 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.

Classificador	Tamanho do Conjunto de Treinamento			
	25%	40%	60%	80%
<i>k</i> -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 2.8: Resultados obtidos para diversos classificadores aplicados ao problema PCV-3C. Pode-se observar os acertos médios obtidos para diferentes tamanhos do conjunto de treinamento.

Os gráficos relacionados aos resultados contidos na Tabela 2.8 são vistos na Figura 2.17. 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 apresenta maior desempenho médio que o classificador SVM/SMO/KMOD na faixa de 25%-40%, para 60% do conjunto de treinamento as acurárias são semelhantes, para 80% do conjunto de dados o classificador SVM/SMO/KMOD apresenta melhor desempenho.

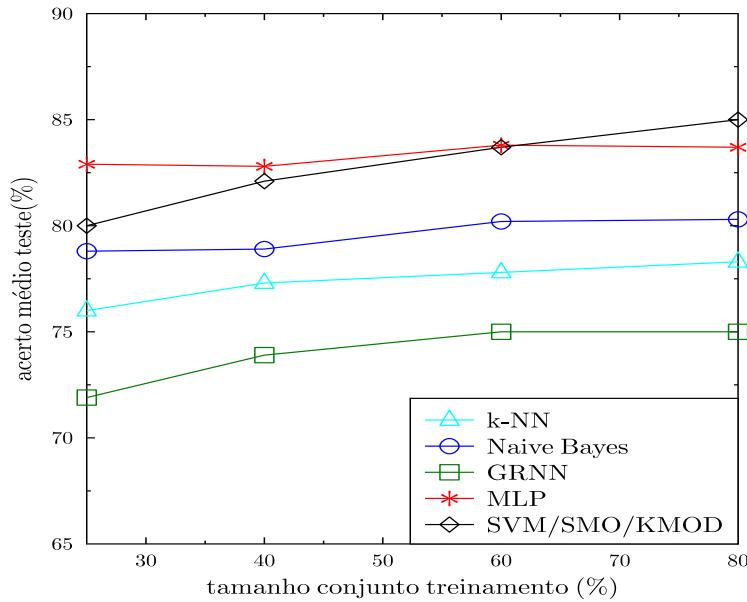


Figura 2.17: Acurácia de diversos classificadores em função do tamanho do conjunto de treinamento.

Na Figura 2.18, 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.

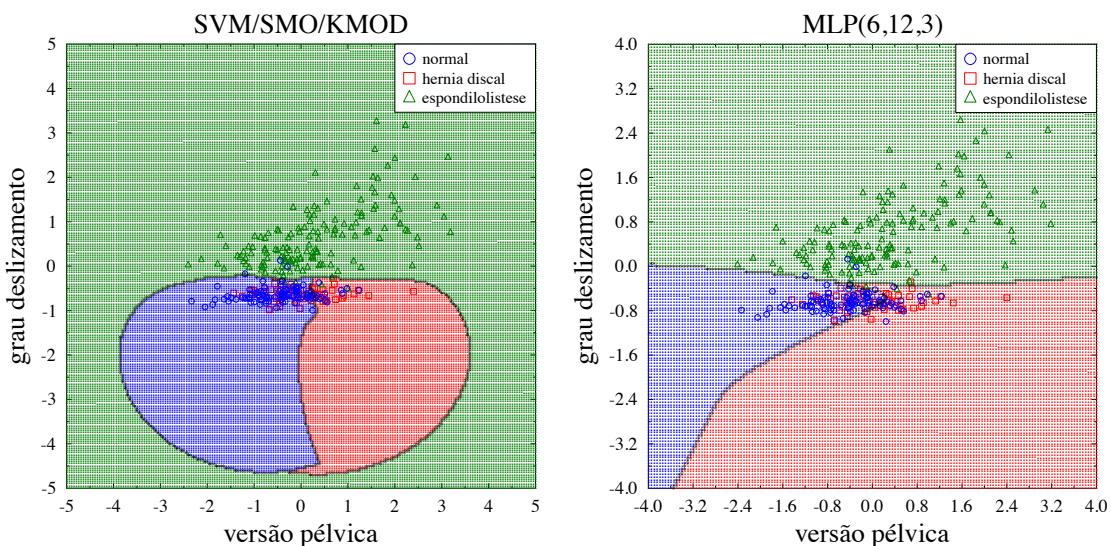


Figura 2.18: (a) Superfície de decisão do classificador SVM/SMO/KMOD [$\gamma = 8, 5; \sigma = 2, 0$ e $C = 2, 5$]. (b) Superfície de decisão da rede MLP(6,12,3) treinada por 1000 épocas com taxa de aprendizado igual a 0,05.

2.9 Conclusão

Neste capítulo 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.

3 *Técnicas para Obtenção de Conjuntos Reduzidos em SVM e LSSVM*

Neste capítulo discute-se o problema da esparsidade em classificadores SVM e LS-SVM, bem como o problema relacionado à obtenção de um conjunto reduzido de vetores para treinamento daqueles classificadores. Para este fim, faz-se inicialmente uma análise dos possíveis valores dos multiplicadores de Lagrange e das variáveis de folga nos classificadores SVM e LSSVM. Em seguida propõe-se um novo método para redução do custo computacional em classificadores SVM e LSSVM.

Diferentemente do encontrado na literatura, o método proposto é aplicável tanto à classificadores LSSVM quanto a classificadores SVM. Ademais, o método se diferencia ainda por ter sua execução em uma etapa anterior ao processo de aprendizagem. Finalmente, um método encontrado na literatura baseado em um algoritmo de quantização vetorial, mais especificamente na rede GNG *Growing Neural Gas*, é estendido para fins de comparação com o algoritmo proposto. No final do capítulo, são apresentados diversos resultados obtidos para os métodos descritos e propostos.

3.1 Introdução

Em mineração de dados diversos conjuntos possuem número elevado de padrões, em torno das centenas ou milhares (HAN, 2005). Este tipo de conjunto tem sido amplamente utilizado para treinar classificadores SVM (WU et al., 2007). No pior caso, o processo de aprendizagem dos classificadores SVM é realizado em $O(N + n.n_{vs} + N.n_{vs}.n)$, em que n representa o número de padrões de treinamento, n_{vs} denota o número total de vetores-suporte e N representa o numero de atributos de \mathbf{x}_i . Isto corresponde ao maior valor entre n^3 e $N.n^2$, visto que nesse caso $n_{vs} \approx n$ (BURGES; CHRISTOPHER, 1998).

Os classificadores SVM resultantes geralmente possuem um grande número de vetores-suporte. Devido a este fato, o processo de classificação pode se tornar muito lento em virtude

do número de operações requeridas, pois o tempo que o classificador leva para classificar um exemplo não-visto é proporcional ao número de vetores-suporte. Assim, classificadores SVM costumam ser consideravelmente mais lentos na fase de testes do que outros métodos de aprendizagem como árvores de decisão ou redes neurais artificiais (BURGES, 1996; BURGES; CHRISTOPHER, 1998; BURGES; SCHÖLKOPF, 1997; LECUN L. BOTOU, 1995; LIU et al., 2003). Não fosse o bastante, o processo de treinamento de um classificador SVM requer a solução de um problema de programação quadrática.

Apesar dos inconvenientes supracitados, classificadores SVM têm tido bastante sucesso devido à sua sólida base formal e abordagem elegante, bem como à sua capacidade de generalização sobre uma gama de problemas reais. A elevada capacidade de generalização desses classificadores é assegurada por propriedades especiais do hiperplano ótimo que maximiza a margem de separação buscando a minimização do erro estrutural no espaço de características (CORTES; VAPNIK, 1995; BOSEN et al., 1992; VAPNIK, 1995).

Geralmente, os algoritmos padrões para treinamento de classificadores SVM como o SMO, produzem soluções com um maior número de vetores-suporte do que o estritamente necessário (HUSSAIN et al., 2008). Devido a isso, vários métodos que buscam a redução da complexidade com base na redução do número de vetores-suporte têm sido propostos. Tais métodos objetivam a eliminação de vetores-suporte menos importantes (poda de vetores-suporte) ou a construção de um novo conjunto reduzido de treinamento, muitas vezes com o mínimo de impacto no desempenho do classificador (DOWNS et al., 2002; HUSSAIN et al., 2008; TANG; MAZZONI, 2006).

Uma alternativa aos classificadores SVM, que requer a resolução de um problema de programação quadrática, é usar classificadores LSSVM. Como visto no capítulo anterior, classificadores LSSVM são uma modificação dos classificadores SVM originais, em que no problema primal são utilizadas restrições de igualdade e as variáveis de folga elevadas ao quadrado. Como resultado, a solução é obtida diretamente a partir de um sistema de equações lineares, em vez da resolução de problema de um programação quadrática (SUYKENS; VANDEWALLE, 1999b). Nesse sentido, é menos complexo resolver um sistema linear do que resolver um problema de programação quadrática. Por outro lado, as alterações introduzidas também resultam na perda da esparsidade do vetor de multiplicadores de Lagrange. Assim, é comum usar todas as instâncias do conjunto de treinamento como vetores-suporte. Para mitigar essa desvantagem, há vários métodos propostos na literatura a fim de aumentar a esparsidade no projeto de classificadores LSSVM (SUYKENS et al., 2000; LI et al., 2006; HOEGAERTS et al., 2004; CARVALHO; BRAGA, 2009).

A seguir são apresentadas as análises dos multiplicadores de Lagrange e das variáveis de folga para fins de esclarecimento sobre o processo de seleção de vetores-suporte em classificadores SVM e LSSVM. Estes conceitos são importantes para o entendimento das técnicas, descritos posteriormente neste capítulo, que visam a redução de complexidade nesses classificadores. A análise realizada sobre os vetores-suporte dos classificadores SVM e LSSVM baseia-se nos trabalhos de Carvalho & Braga (2005) e Carvalho & Braga (2009).

3.2 Análise dos Vetores-Suporte

3.2.1 Vetores-Suporte em Classificadores SVM

O multiplicador de Lagrange α_i apresentado na Equação (2.60) indica a relevância de um padrão de treinamento \mathbf{x}_i para a construção da superfície de decisão. Os padrões que possuem valores maiores que zero são chamados de vetores-suporte. A introdução de variáveis de folga permite que alguns padrões sejam incorretamente classificados com o intuito de obter uma superfície de decisão menos complexa. Variáveis de folga são obtidas automaticamente durante o processo de treinamento dos classificadores SVM (CORTES; VAPNIK, 1995).

Na Figura 3.1 são apresentados os padrões $\mathbf{x}_i \in \mathbb{R}^2$ das classes positiva C_{+1} , representados por quadrados, e padrões da classe negativa C_{-1} , representados por círculos. Percebe-se ainda que o espaço está dividido em quatro regiões (R_1, R_2, R_3 e R_4). A região R_1 abrange a área disposta à esquerda da margem positiva m_{+1} , a região R_2 abrange a área entre a margem da classe positiva m_{+1} e o hiperplano ótimo h_o . Similarmente, a região R_3 abrange a área entre o hiperplano ótimo h_o e a margem da classe negativa m_{-1} , bem como a região R_4 abrange a região à direita da margem da classe negativa m_{-1} .

Caso um padrão pertencente ao conjunto de treinamento esteja posicionado dentro da margem de sua classe, ou mesmo do lado incorreto da superfície de decisão, a variável de folga correspondente a este padrão possui valor diferente de zero. Esta situação pode ser visualizada na Figura 3.1. As variáveis de folga indicam a posição do vetor de treinamento em relação à margem correspondente à sua classe.

Uma característica das soluções obtidas após o treinamento de classificadores SVM é a esparsidade do vetor de multiplicadores de Lagrange, isto é, vários deles apresentam valores nulos após o processo de aprendizagem. Desta forma, apenas uma parte dos padrões de treinamento são considerados na função discriminante. É importante ressaltar, porém, que soluções esparsas podem ainda conter mais vetores-suporte que o estritamente necessário. Uma discussão deta-

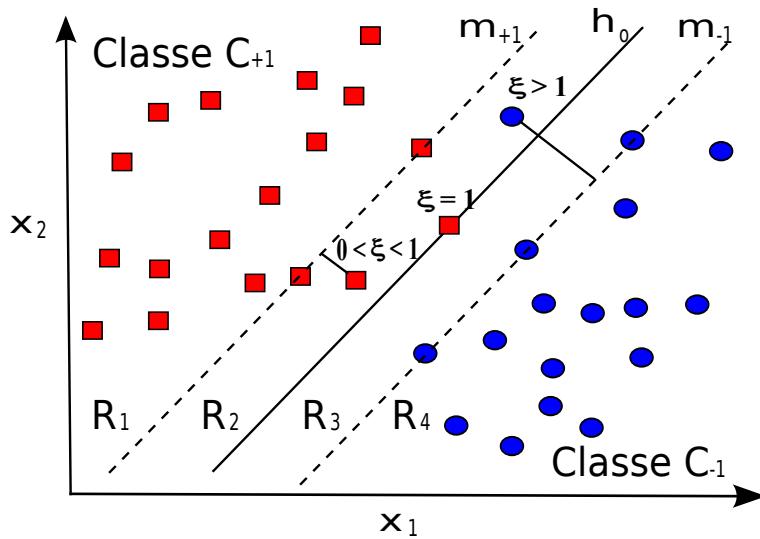


Figura 3.1: Interpretação geométrica das variáveis de folga utilizadas em classificadores SVM. Valores na faixa $0 < \xi < 1$ indicam exemplos na margem de separação de sua classe e valores na faixa $\xi > 1$ indicam exemplos incorretamente classificados.

lhada dos valores que multiplicadores de Lagrange e variáveis de folga de classificadores SVM podem assumir é apresentada a seguir.

- Quando $\alpha_i = 0$, o padrão \mathbf{x}_i não é um vetor-suporte. Este padrão é classificado corretamente e situa-se na região de sua classe, ou seja, na região R_1 ou R_4 , e não faz parte da função discriminante, e, consequentemente, não interfere na construção do hiperplano ótimo. Nesta situação, $\xi_i = 0$ e $y_i \cdot f(\mathbf{x}_i) > 1$.
- Quando $0 < \alpha_i < C$, o padrão \mathbf{x}_i é um vetor-suporte. Este padrão apresenta-se sobre a margem mais próxima à região de sua classe, ou seja, sobre a margem da classe positiva m_{+1} para um padrão da classe positiva ou sobre a margem da classe negativa m_{-1} para um padrão da classe negativa. Neste caso, $\xi_i = 0$ e $y_i \cdot f(\mathbf{x}_i) = 1$.
- Quando $\alpha_i = C$, o padrão \mathbf{x}_i é um vetor-suporte. Este padrão pode estar localizado entre a margem e a superfície de separação, caso $0 < \xi_i < 1$, em que o padrão apresenta-se sobre a região R_2 para padrões da classe positiva ou sobre a região R_3 para padrões da classe negativa; na própria superfície de separação h_0 caso $\xi_i = 1$; ou do outro lado da superfície de separação, na região da classe oposta, caso $\xi_i > 1$, em que o padrão situa-se nas regiões R_3 ou R_4 para padrões da classe positiva ou nas regiões R_1 ou R_2 para padrões da classe negativa. Neste caso, $y_i \cdot f(\mathbf{x}_i) < 1$.

3.2.2 Vetores-Suporte em Classificadores LSSVM

A separação no espaço de características realizada por um classificador LSSVM é diferente da efetuada por um classificador SVM. No espaço de características, são construídos dois hiperplanos, h_{+1} e h_{-1} , paralelos entre si, um para os padrões pertencentes à classe positiva (C_{+1}) e outro para os padrões pertencentes à classe negativa (C_{-1}), respectivamente. Os dois hiperplanos paralelos estão dispostos aproximadamente no centro da distribuição de cada classe e dispostos o mais longe possível um do outro. Esta situação pode ser melhor visualizada na Figura 3.2.

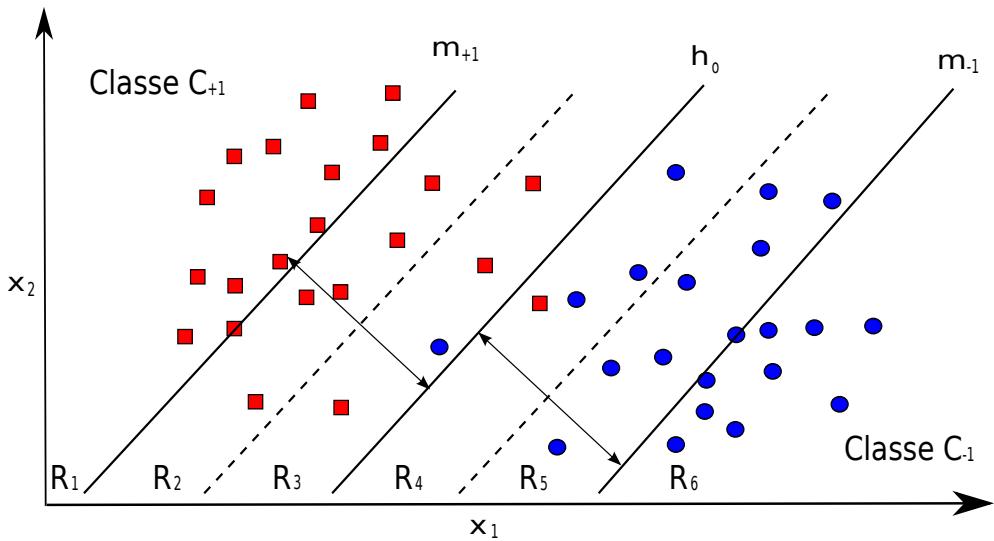


Figura 3.2: Hiperplanos construídos por um classificador LSSVM no espaço de características e regiões correspondentes.

Um padrão da classe C_{+1} é considerado corretamente classificado quanto menor for sua distância em relação ao hiperplano h_{+1} disposto sobre os padrões pertencentes à classe C_{+1} . O mesmo raciocínio se aplica a um padrão da classe C_{-1} . A seguir, são analisados os possíveis valores assumidos pelas variáveis de folga de classificadores LSSVM após o seu processo de aprendizagem.

As variáveis de folga de um classificador LSSVM possuem uma interpretação diferente daquela usada para classificadores SVM. Para classificadores LSSVM, as variáveis de folga indicam a distância e a orientação em relação aos hiperplanos paralelos criados. Conforme apresentado no Capítulo 2, a condição $\alpha_i = \gamma \xi_i$ indica que o valor do multiplicador de Lagrange é diretamente proporcional à variável de folga associada ao padrão de treinamento. Este fato sugere que os multiplicadores de Lagrange podem ser utilizados para determinar a relevância de um vetor para a solução de classificadores LSSVM. Neste sentido, tem-se as seguintes possibilidades.

- quando $\alpha_i \gg 0$, o padrão \mathbf{x}_i está localizado longe do hiperplano da sua classe e próximo ao hiperplano h_o . Um padrão, nesta situação, apresenta-se correta ou incorretamente classificado. Por exemplo, o padrão encontra-se na região R_3 ou R_4 para um padrão da classe C_{+1} . Nesta situação, $\xi_i \gg 0$, pois $\alpha_i = \gamma \xi_i$.
- quando $\alpha_i > 0$, o padrão \mathbf{x}_i está situado entre os dois hiperplanos paralelos, próximo ao hiperplano de sua classe (e.g. na região R_5 para um padrão da classe C_{-1}). Nesta situação um padrão é classificado corretamente, bem como $\xi_i > 0$.
- quando $\alpha_i = 0$, o padrão \mathbf{x}_i está situado exatamente no hiperplano paralelo associado à sua classe (sobre m_{+1} ou m_{-1}). Nesta situação bastante rara, $\xi_i = 0$ e o padrão apresenta-se corretamente classificado.
- quando $\alpha_i < 0$ e, portanto, $\xi_i < 0$, o padrão \mathbf{x}_i está próximo ao hiperplano associado à sua classe, porém do lado oposto ao outro hiperplano. Neste caso, o padrão também está corretamente classificado. Por exemplo, o padrão encontra-se na região R_1 para um padrão da classe C_{+1} , porém mais próximo ao hiperplano m_{+1} .
- quando $\alpha_i \ll 0$, o padrão \mathbf{x}_i está posicionado longe do hiperplano de sua classe e distante dos padrões da outra classe. Neste caso, o padrão encontra-se corretamente classificado e $\xi_i \ll 0$. Por exemplo, o padrão encontra-se na região R_6 , no entanto, mais distante do hiperplano m_{-1} (próximo ao extremo direito).

3.3 Métodos para obtenção de Conjuntos Reduzidos

Nesta seção são descritos alguns dos métodos/técnicas para a redução da complexidade em classificadores SVM e LSSVM. Estes métodos são, em geral, específicos para um tipo de classificador; ou seja, somente podem ser aplicados a classificadores SVM ou a classificadores LSSVM. A especificidade de tais métodos decorre do processo de redução da quantidade de vetores-suporte, que muitas vezes depende da formulação do problema ou do processo de aprendizagem.

Nesta seção, é apresentada ainda uma nova proposta para obtenção de conjuntos reduzidos. A base da técnica proposta é um método que inicialmente utilizou a rede auto-organizável. Este método foi denominado *Opposite Maps* (ROCHA-NETO; BARRETO, 2011), pois utiliza duas redes de Kohonen, uma para cada classe em um problema binário. Posteriormente, a técnica OM foi estendida de modo a poder utilizar qualquer algoritmo de quantização vetorial para obtenção de conjuntos reduzidos.

Linda & Manic (2009) propõem uma técnica que utiliza a rede GNG em conjunto com

o algoritmo SMO em uma etapa preliminar ao processo de aprendizagem para obtenção do conjunto de vetores-suporte. Esta técnica denomina-se GNG-SVM. Originalmente, esta técnica é apenas aplicada a classificadores SVM, porém propõe-se também nesta tese uma extensão para utilização com classificadores LSSVM.

Logo, para compreensão do processo de obtenção de conjuntos reduzidos, são descritos succinctamente dois métodos para a redução de conjuntos em classificadores SVM, a saber: *Reduced Set Method* e *Reduced SVM* e dois métodos para a redução de conjuntos em classificadores LSSVM: *Pruning LSSVM* e *IP-LSSVM*. Em seguida, é descrito o algoritmo GNG-SVM e as técnicas propostas nesta tese que se baseiam no método OM.

3.3.1 Reduced Set Method (RSM)

O método RSM visa reduzir o número de vetores-suporte em uma etapa posterior ao treinamento de um classificador SVM. Ou seja, o classificador SVM é treinado por um algoritmo de treinamento (e.g. SMO) e, em seguida, os vetores-suporte \mathbf{x}_i e seus multiplicadores de Lagrange $\{\alpha_i\}_{i=1}^{n_{vs}}$ obtidos são substituídos por uma combinação de outros vetores \mathbf{z}_i e outros multiplicadores $\{\beta_i\}_{i=1}^{m_{vs} < n_{vs}}$. Neste contexto, considere a função de decisão do classificador SVM, dado por

$$y(\mathbf{x}) = \text{sign} \left(\sum_{i=1}^{n_{vs}} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b \right) \quad (3.1)$$

em que, como descrito no capítulo anterior, α_i é o multiplicador de Lagrange associado ao padrão \mathbf{x}_i , \mathbf{x} é o vetor a ser classificado, $K(\cdot, \cdot) = \phi(\cdot)^T \phi(\cdot)$ é uma função *kernel*, b é o viés, n_{vs} representa a quantidade de vetores-suporte e m_{vs} representa a quantidade reduzida de vetores-suporte.

Neste método o vetor \mathbf{w} do classificador SVM treinado, expresso por

$$\mathbf{w} = \sum_{i=1}^{n_{vs}} \alpha_i y_i \phi(\mathbf{x}_i), \quad (3.2)$$

é aproximado por um pequeno número de vetores \mathbf{z}_i em que o termo $\alpha_i y_i$ é substituído por β_i , tal que

$$\mathbf{w}^* \approx \sum_{i=1}^{m_{vs}} \beta_i K(\mathbf{z}_i, \mathbf{x}) \quad (3.3)$$

Assim, a função de decisão modificada é descrita como segue:

$$y(\mathbf{x}) \approx \text{sign} \left(\sum_{i=1}^M \beta_i K(\mathbf{z}_i, \mathbf{x}) + b \right). \quad (3.4)$$

Das Equações (3.2) e (3.3), a medida de distância ρ definida como a diferença entre o vetor original \mathbf{w} e o vetor aproximado \mathbf{w}^* no espaço de característica, deve ser minimizada, ou seja

$$\min_{\beta, \mathbf{z}} \rho = \min_{\beta, \mathbf{z}} \|\mathbf{w} - \mathbf{w}^*\| = \min_{\beta, \mathbf{z}} \left\| \sum_{i=1}^N \alpha_i y_i \phi(\mathbf{x}_i) - \sum_{i=1}^M \beta_i K(\mathbf{z}_i, \mathbf{x}) \right\| \quad (3.5)$$

Em geral, uma técnica de otimização sem restrições é usada para encontrar o valor mínimo de ρ . Assim, para este propósito, o método do gradiente descendente pode ser utilizado. Detalhes do *Reduced Set Method* podem ser encontrados em (BURGES, 1996; SCHÖLKOPF et al., 1998; SCHOLKOPF; SMOLA, 2002; SCHÖLKOPF et al., 1999; TANG; MAZZONI, 2006).

3.3.2 Reduced SVM (RSVM)

A formulação do método RSVM (LEE; MANGASARIAN, 2001b) é derivada do método *Generalized Support Vector Machine* (GSVM) (MANGASARIAN, 2000) e do método *Smooth Support Vector Machine* (SSVM) (LEE; MANGASARIAN, 2001c). O método RSVM considera a margem flexível similarmente à SVM, e adiciona o termo $\gamma^2/2$ à função objetivo a ser minimizada. O problema de otimização com restrição após algumas operações é convertido em um problema de otimização sem restrições.

A principal idéia desse método consiste na redução da matriz kernel Q de $n \times n$ para $n \times m$, em que m é o tamanho de um subconjunto aleatoriamente selecionado dos dados de treinamento considerados como candidatos a vetores-suporte. Neste método, o viés b também é minimizado, juntamente com o vetor de pesos \mathbf{w} , no problema primal. A solução obtida com RSVM difere das soluções obtidas quando se busca resolver o classificador SVM por problemas menores, ou subconjuntos do conjunto de treinamento, pois as n restrições do problema primal são mantidas durante o processo de otimização.

3.3.3 Pruning LSSVM

Alguns trabalhos propõem o uso do valor absoluto (magnitude) dos multiplicadores de Lagrange como um indicativo para a determinação dos vetores-suporte em um classificador LS-SVM treinado (SUYKENS et al., 2000b; FUNG; MANGASARIAN, 2001).

Ao se considerar o uso do valor absoluto de α_i , $|\alpha_i|$, como critério de seleção dos vetores-suporte, deve-se efetuar as seguintes operações:

- manter os padrões em que $\alpha_i \gg 0$. Estes padrões posicionam-se na região de fronteira entre as duas classes ou na região associada à outra classe.

- descartar os padrões em que $\alpha_i \geq 0$ ou $\alpha_i < 0$. Padrões nesta condição estão classificados de forma correta.
- manter os padrões em que $\alpha_i \ll 0$. Estes padrões estão corretamente classificados.

O método *Prunning LSSVM* permite que o classificador LSSVM seja capaz de selecionar um subconjunto dos padrões de treinamento como pertencentes ao conjunto de vetores-suporte (SUYKENS et al., 2000b). Esta abordagem baseia-se na utilização de um método de poda no qual padrões de treinamento são removidos de acordo com o valor absoluto do multiplicador de Lagrange $|\alpha_i|$ associado a eles. A eliminação dos multiplicadores de Lagrange ocorre de forma recursiva, de modo que em cada iteração uma pequena quantidade de padrões de treinamento é eliminada do conjunto de vetores-suporte. Utiliza-se, como conjunto de validação, um subconjunto dos vetores de treinamento para avaliação do desempenho do novo classificador. O critério de parada deste processo é determinado pela diminuição do desempenho do classificador treinado com o conjunto reduzido de padrões, em relação à um conjunto de validação. Em suma, o método *Prunning LSSVM* tem seu funcionamento descrito pelos seguintes passos:

PASSO 1 - Treinar um classificador LSSVM com todo o conjunto de treinamento.

PASSO 2 - Remover uma pequena quantidade dos vetores pertencentes ao conjunto de treinamento, cujos valores de $|\alpha_i|$ sejam os menores dentre os existentes.

PASSO 3 - Retreinar o classificador LSSVM usando o método da pseudo inversa com o conjunto de treinamento reduzido.

PASSO 4 - Retornar ao PASSO 2, desde que a performance no conjunto de validação não diminua. Caso contrário, finalizar o processo com o conjunto de treinamento da iteração anterior.

O processo descrito acima permite que se obtenha um conjunto reduzido de vetores-suporte. Para isto, basta que se tenha mais de uma iteração desse processo. Os parâmetros de treinamento nesta abordagem são o parâmetro de regularização γ , os parâmetros da função de *kernel* e a quantidade de vetores a ser reduzida a cada iteração.

3.3.4 IP-LSSVM

Carvalho & Braga (2009) propõem avaliar diretamente o valor dos multiplicadores de Lagrange α_i , e não apenas seu valor absoluto. Nesta situação, deve-se realizar as seguintes operações:

- manter os padrões em que $\alpha_i \gg 0$. Estes padrões posicionam-se na região de fronteira entre as duas classes ou na região da outra classe.

- descartar os padrões em que $\alpha_i \geq 0$, $\alpha_i < 0$ ou $\alpha_i \ll 0$.

De uma forma comparativa, padrões que descrevem a condição $\alpha_i \gg 0$ como verdadeira em classificadores LSSVM equivalem aos vetores-suporte obtidos no processo de aprendizagem de classificadores SVM. Já os padrões que descrevem uma das seguintes condições: $\alpha_i \geq 0$, $\alpha_i < 0$ ou $\alpha_i \ll 0$ como verdadeira em classificadores LSSVM equivalem aos vetores de classificadores SVM que possuem valores nulos para seus multiplicadores de Lagrange (CARVALHO; BRAGA, 2005).

O método IP-LSSVM obtém o conjunto reduzido em duas fases. Na primeira fase, utiliza-se a inversa para obtenção da solução do sistema linear. Enquanto na segunda utiliza-se o método da pseudo inversa para resolução daquele sistema linear modificado, em que algumas colunas são excluídas com base nos multiplicadores de Lagrange obtidos na fase anterior (CARVALHO; BRAGA, 2009). O processo de treinamento do classificador IP-LSSVM pode ser descrito como segue.

PASSO 1- Treinar um classificador LSSVM com todo o conjunto de treinamento, usando a matriz inversa pois \mathbf{A} é uma matriz quadrada.

PASSO 2 - Especificar um parâmetro τ , que define o percentual dos vetores de treinamento que serão vetores-suporte.

PASSO 3 - Ordenar os vetores de treinamento $\{\mathbf{x}_i, y_i\}_{i=1}^n$ por seus valores de α_i .

PASSO 4 - Selecionar um percentual dos dados de treinamento $(1 - \tau)$ que corresponde aos menores valores de α_i .

PASSO 5 - Construir uma matriz não-quadrada \mathbf{A}_{rs} pela remoção das colunas de \mathbf{A} associadas aos vetores com menores valores de α_i .

PASSO 6 - Solucionar o sistema linear $\mathbf{x}_{rs} = \mathbf{A}^* \mathbf{b}$, por $\mathbf{A}^* = (\mathbf{A}_{rs}^T \mathbf{A}_{rs})^{-1} \mathbf{A}_{rs}^T$.

PASSO 7 - Os vetores-suporte são os vetores do conjunto de treinamento associados às colunas contidas na matriz \mathbf{A}_{rs} .

PASSO 8 - Os valores de α_i e b para os vetores-suporte podem ser obtidos de \mathbf{x}_{rs} , enquanto o valor de α_i para os outros vetores é igual a zero.

3.3.5 GNG-SVM

Inspirado na rede *Neural Gas* de Martinet & Schulten (1991), Fritzke (1995) propôs a rede *Growing Neural Gas* (GNG). Este algoritmo foi desenvolvido para realização de agrupamento (*clustering*) e quantização vetorial. Mais ainda, o algoritmo combina o mecanismo de inclusão de neurônios com a capacidade de aprendizado topológico do mecanismo de aprendizado com-

petitivo hebbiano (FRITZKE, 1994). Mais detalhes sobre a rede GNG podem ser obtidos no Apêndice B.

O classificador GNG-SVM consiste em duas fases. Na primeira, a topologia dos dados é obtida pelo treinamento de instâncias da rede GNG, uma para cada classe do problema¹. Assim, a base de dados do problema é reduzida para a informação topológica extraída. Na segunda fase, o algoritmo SMO é treinado usando a nova base de dados reduzida.

Formalmente, para um problema binário devem ser realizados os seguintes passos:

PASSO 1 - Dividir o conjunto de treinamento $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ em dois subconjuntos, $\mathcal{D}^{(1)} = \{(\mathbf{x}_i, y_i) | y_i = +1\}_{i=1}^n$ e $\mathcal{D}^{(2)} = \{(\mathbf{x}_i, y_i) | y_i = -1\}_{i=1}^n$;

PASSO 2 - Treinar as redes GNG-1 e GNG-2 usando os conjuntos $\mathcal{D}^{(1)}$ e $\mathcal{D}^{(2)}$, respectivamente. Dentre os possíveis critérios de parada estão o número máximo de iterações, número máximo de neurônios da rede ou convergência dos pesos da rede;

PASSO 3 - Estender os conjuntos de protótipos W_1 e W_2 com a inclusão dos rótulos das classes, ou seja

$$W_1^* = \{(\mathbf{w}_j, +1)\}_{i=1}^{n_1},$$

$$W_2^* = \{(\mathbf{w}_j, -1)\}_{i=1}^{n_2},$$

em que \mathbf{w}_j representa o j -ésimo protótipo e n_1 e n_2 a quantidade de protótipos das redes GNG-1 e GNG-2, respectivamente;

PASSO 4 - Construir um nova base de dados \mathcal{D}^* , tal que $\mathcal{D}^* = W_1^* \cup W_2^*$;

PASSO 5 - Treinar o classificador SVM usando a base de dados \mathcal{D}^* e o algoritmo SMO.

3.4 Opposite Maps (Proposta 2)

O método *Opposite Maps* (OM) também visa obter um conjunto reduzido a partir do total de vetores de treinamento. Os vetores obtidos como resultado do processamento deste algoritmo são fortes candidatos à pertencerem ao conjunto de vetores-suporte. Estes vetores correspondem aos padrões da classe C_{+1} que estão mais próximos da classe C_{-1} , e os padrões da classe C_{-1} que estão mais próximos da classe C_{+1} . Esta situação ocorre quando não há sobreposição entre as distribuições de dados das classes. Na situação em que os dados se sobrepõem, o método OM obtém justamente os vetores que se encontram na região de sobreposição entre os padrões das classes (ROCHA-NETO; BARRETO, 2011). Do exposto, pode-se notar, que o projeto do método OM também utiliza o critério proposto em Carvalho & Braga (2009), pois o método

¹Para um problema binário temos duas redes GNG.

OM busca os padrões que apresentam $\alpha_i \gg 0$ nas LSSVM. Para os classificadores SVM, o método busca os padrões com $\alpha_i > 0$. A formalização do método é mostrado a seguir.

PASSO 1 - Dividir o conjunto de dados de treinamento $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^l$ em dois subconjuntos:

$$\mathcal{D}^{(1)} = \{(\mathbf{x}_i, y_i) | y_i = +1\}, \quad i = 1, \dots, l_1 \quad (\text{para classe 1}) \quad (3.6)$$

$$\mathcal{D}^{(2)} = \{(\mathbf{x}_i, y_i) | y_i = -1\}, \quad i = 1, \dots, l_2 \quad (\text{para classe 2}) \quad (3.7)$$

em que l_1 e l_2 são as quantidades de padrões dos subconjuntos $\mathcal{D}^{(1)}$ e $\mathcal{D}^{(2)}$, respectivamente.

PASSO 2 - Treinar uma rede SOM usando um subconjunto $\mathcal{D}^{(1)}$ e uma outra rede SOM usando um subconjunto $\mathcal{D}^{(2)}$. As redes SOM treinadas são denominadas SOM-1 e SOM-2, respectivamente.

PASSO 3 - Para cada vetor $\mathbf{x}_i \in \mathcal{D}^{(1)}$ encontrar o neurônio vencedor correspondente na rede SOM-1. Em seguida, deve-se podar todos os *neurônios mortos*² na rede SOM-1. Deve-se repetir o mesmo procedimento para cada vetor $\mathbf{x}_i \in \mathcal{D}^{(2)}$: encontrar os neurônios vencedores correspondentes na rede SOM-2 e podar todos os neurônios mortos. As redes podadas são denominadas PSOM-1 e PSOM-2, respectivamente.

PASSO 4 - Neste passo, os neurônios vencedores para um dado subconjunto ($\mathcal{D}^{(1)}$ ou $\mathcal{D}^{(2)}$) são buscados no conjunto de protótipos da rede oposta.

PASSO 4.1 - Para cada $\mathbf{x}_i \in \mathcal{D}^{(1)}$ encontrar o neurônio vencedor correspondente na rede PSOM-2:

$$c_i^{(2)} = \arg \min_{\forall j} \|\mathbf{x}_i - \mathbf{w}_j^{(2)}\|, \quad i = 1, \dots, l_1, \quad (3.8)$$

em que $\mathbf{w}_j^{(2)}$ é o j -ésimo vetor protótipo na rede PSOM-2. Assim, $c_i^{(2)}$ representa o índice do neurônio vencedor na rede PSOM-2 para o i -ésimo exemplo em $\mathcal{D}^{(1)}$.

PASSO 4.2 - Para cada $\mathbf{x}_i \in \mathcal{D}^{(2)}$ encontrar o neurônio vencedor correspondente na rede PSOM-1:

$$c_i^{(1)} = \arg \min_{\forall j} \|\mathbf{x}_i - \mathbf{w}_j^{(1)}\|, \quad i = 1, \dots, l_2, \quad (3.9)$$

em que $\mathbf{w}_j^{(1)}$ é o j -ésimo vetor protótipo na PSOM-1. Logo, $c_i^{(1)}$ denota o índice do neurônio vencedor na rede PSOM-1 para o i -ésimo exemplo em

²Neurônios que não tiveram sido selecionados como BMU para qualquer vetor $\mathbf{x}_i \in \mathcal{D}^{(1)}$.

$\mathcal{D}^{(2)}$.

PASSO 5- Considere $\mathcal{C}^{(2)} = \{c_1^{(2)}, c_2^{(2)}, \dots, c_{l_2}^{(2)}\}$ como sendo o conjunto de índices de todos os neurônios vencedores encontrados no Passo 4.1, e $\mathcal{C}^{(1)} = \{c_1^{(1)}, c_2^{(1)}, \dots, c_{l_1}^{(1)}\}$ como sendo o conjunto de índices de todos os BMUs encontrados no PASSO 4.2.

PASSO 6 - Neste passo, o conjunto reduzido de vetores de dados é formado.

PASSO 6.1 - Para cada protótipo da rede PSOM-1 em $\mathcal{C}^{(1)}$, deve-se encontrar o padrão mais próximo dentre os vetores de dados $\mathbf{x}_i \in \mathcal{D}^{(1)}$. Seja $\mathcal{X}^{(1)}$ o subconjunto de padrões mais próximos aos protótipos da rede PSOM-1 em $\mathcal{C}^{(1)}$.

PASSO 6.2 - Para cada protótipo da rede PSOM-2 em $\mathcal{C}^{(2)}$ deve-se encontrar o padrão mais próximo dentre os vetores de dados $\mathbf{x}_i \in \mathcal{D}^{(2)}$. Seja $\mathcal{X}^{(2)}$ o subconjunto de padrões mais próximos aos protótipos da rede PSOM-2 em $\mathcal{C}^{(2)}$.

Daí, o conjunto reduzido de vetores é formado por $\mathcal{X}_{rs} = \mathcal{X}^{(1)} \cup \mathcal{X}^{(2)}$.

3.4.1 Opposite Maps: Passo-a-Passo

Os passos apresentados no algoritmo acima estão descritos na Figura 3.3-3.13. Nesta figura, os passos estão dispostos seqüencialmente de cima para baixo. No último quadro, é apresentado o conjunto reduzido \mathcal{X}_{rs} obtido pelo OM para este problema hipotético.

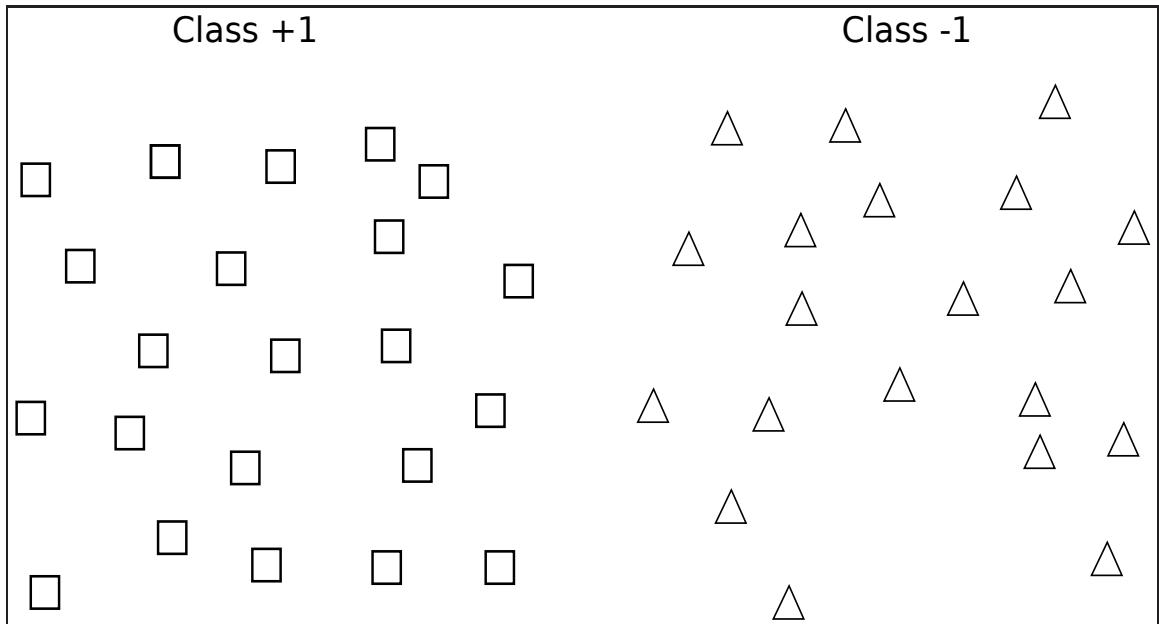


Figura 3.3: Separação do conjunto de dados $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^l$ em dois subconjuntos: $\mathcal{D}^{(1)} = \{(\mathbf{x}_i, y_i) | y_i = +1\}$ (quadrados) e $\mathcal{D}^{(2)} = \{(\mathbf{x}_i, y_i) | y_i = -1\}$ (círculos). Equivale ao **Passo 1** do método OM.

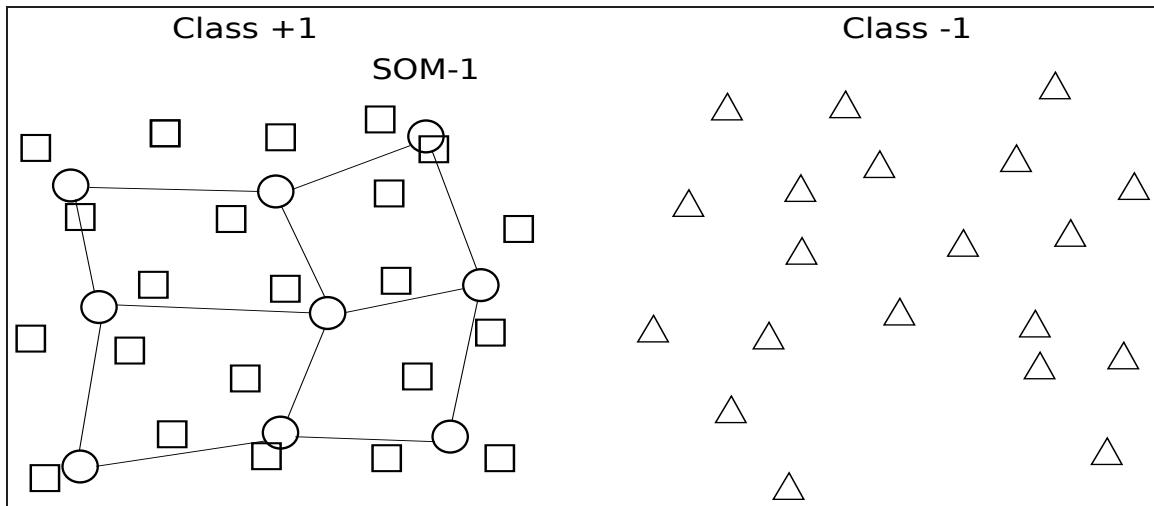


Figura 3.4: Treinamento da rede SOM usando o subconjunto $\mathcal{D}^{(1)}$ (SOM-1). Equivale ao **Passo 2.a**.

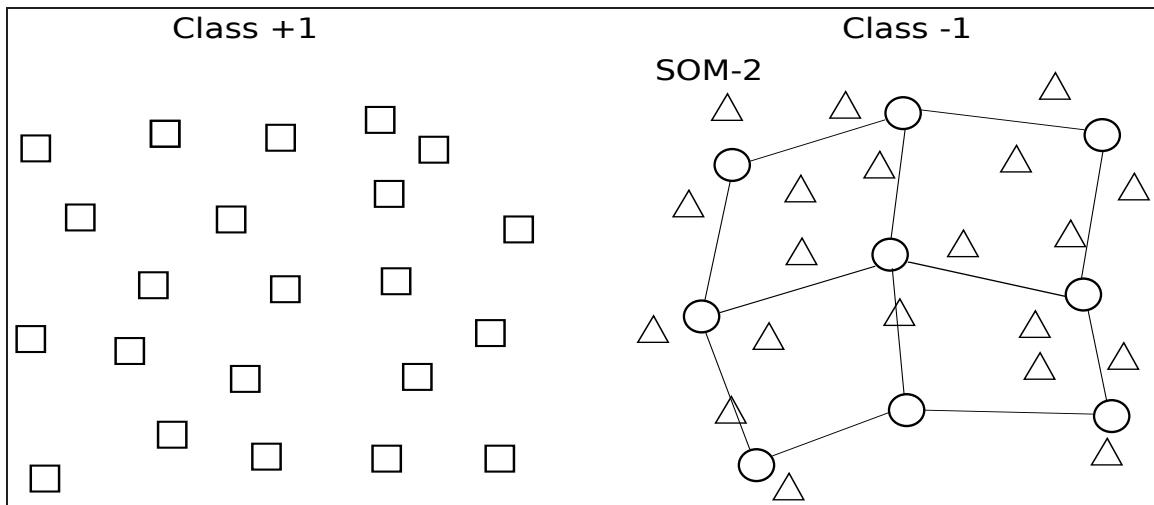


Figura 3.5: Treinamento da rede SOM usando o subconjunto $\mathcal{D}^{(2)}$ (SOM-2). Equivale ao **Passo 2.b**.

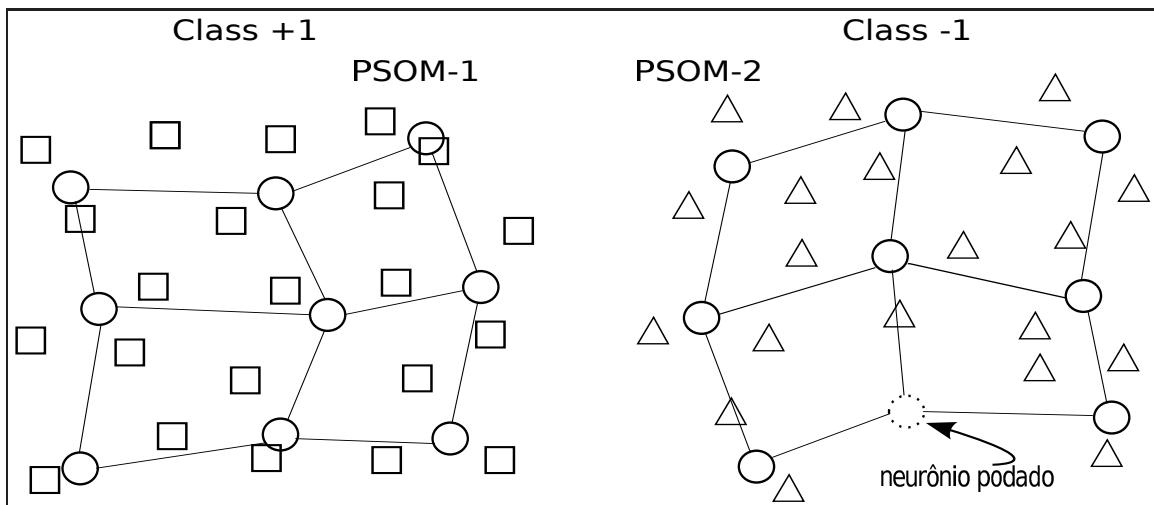


Figura 3.6: Realização de poda de todos os neurônios mortos nas redes SOM-1 e SOM-2. Como resultado tem-se as redes PSOM-1 e PSOM-2. Equivale ao **Passo 3**.

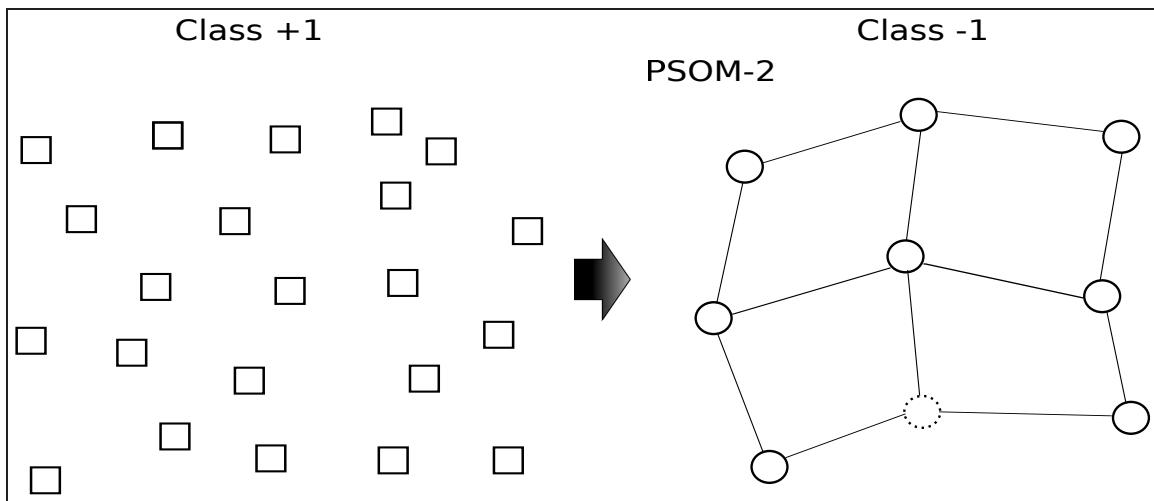


Figura 3.7: Apresentação do conjunto $\mathcal{D}^{(1)}$ à rede PSOM-2. Equivale ao **Passo 4.1a**.

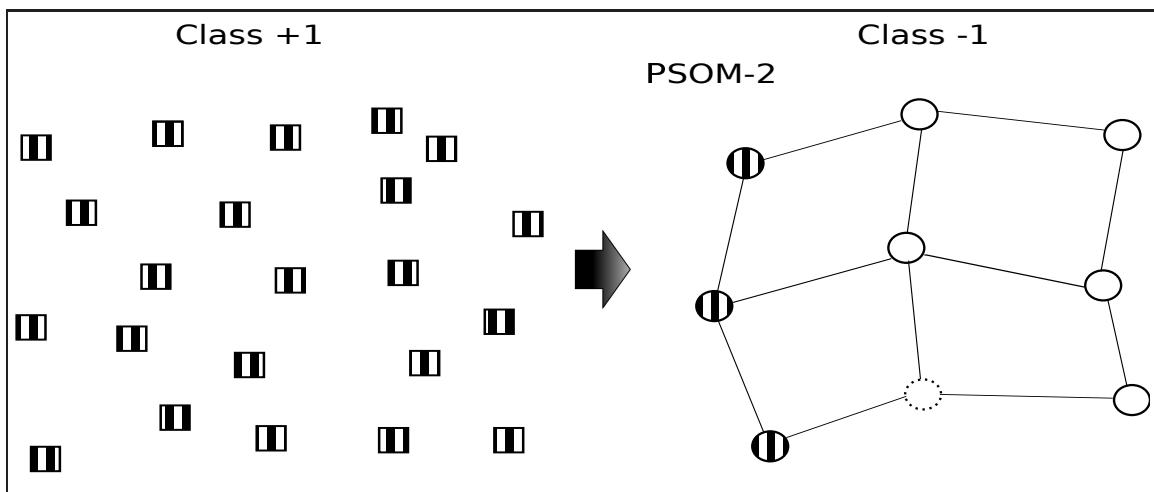


Figura 3.8: Busca dos neurônios vencedores para o conjunto de dados $\mathcal{D}^{(2)}$ na rede PSOM-2. Equivale ao **Passo 4.1b**.

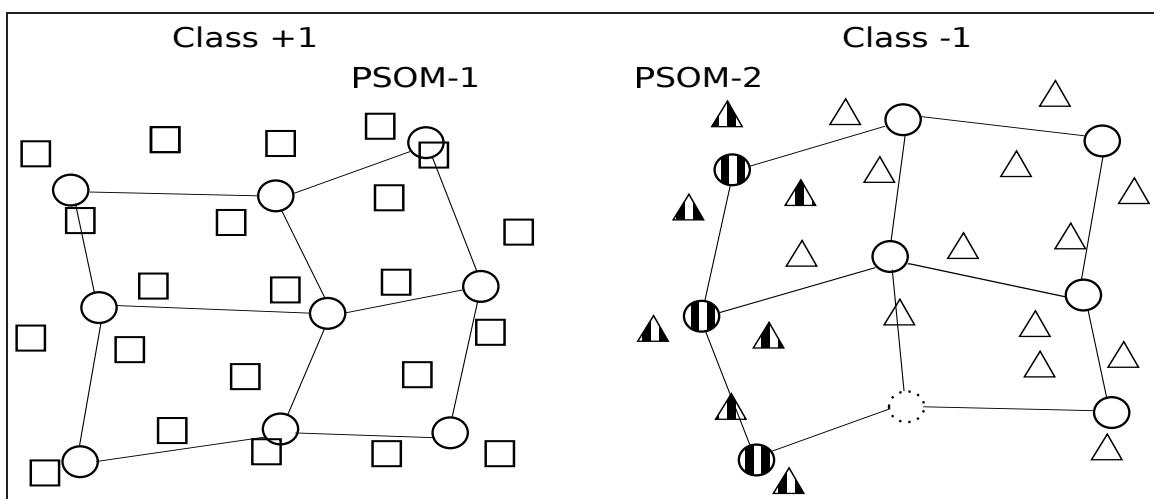


Figura 3.9: Para cada neurônio ativado no **Passo 4.1a**, encontrar seus K pontos mais próximos. Refere-se a este conjunto como $\mathcal{X}^{(1)}$. Equivale ao **Passo 6.1**.

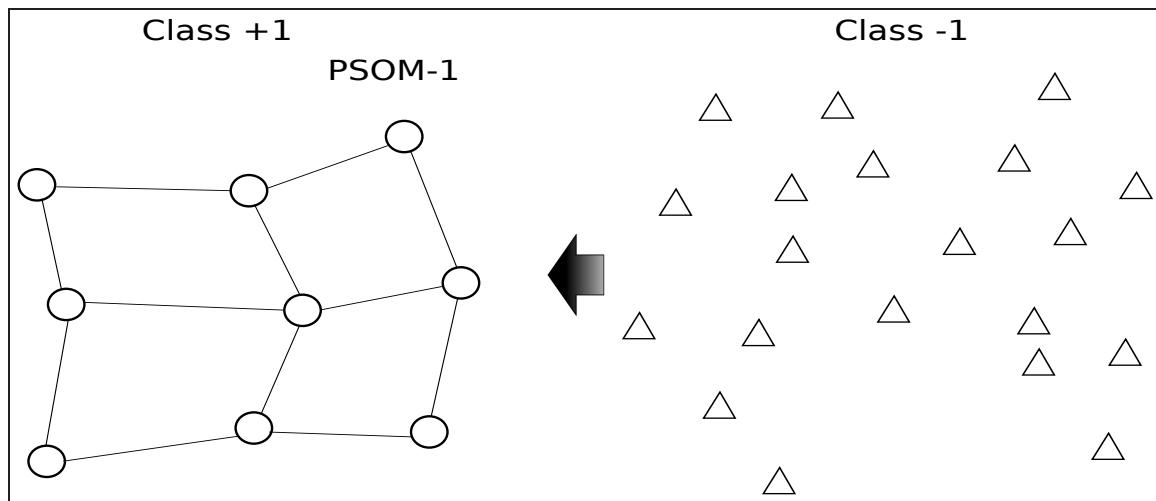


Figura 3.10: Apresentação do conjunto $\mathcal{D}^{(2)}$ à rede PSOM-1. Equivale ao **Passo 4.2a**.

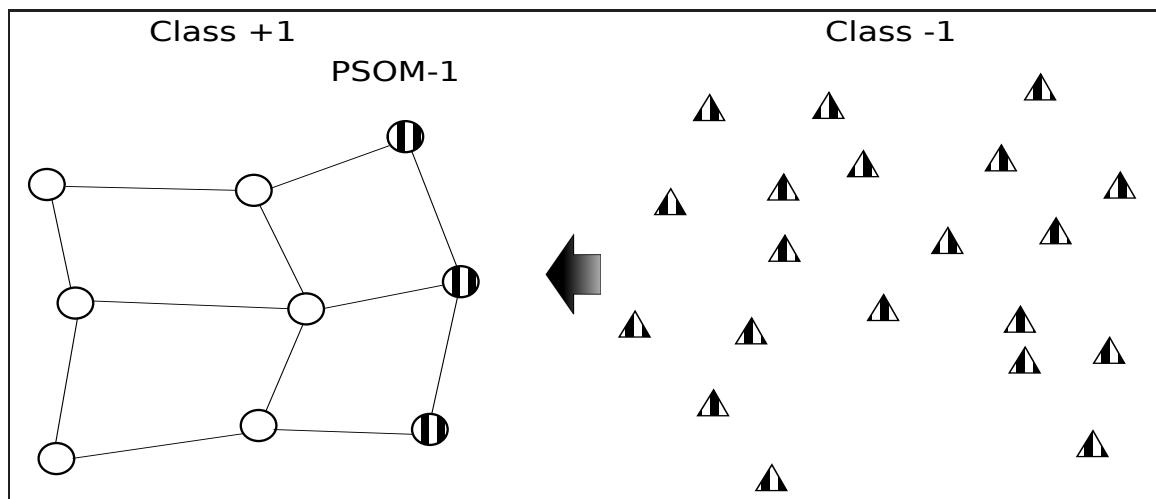


Figura 3.11: Busca dos neurônios vencedores para o conjunto $\mathcal{D}^{(1)}$ na rede PSOM-1. Equivale ao **Passo 4.2b**.

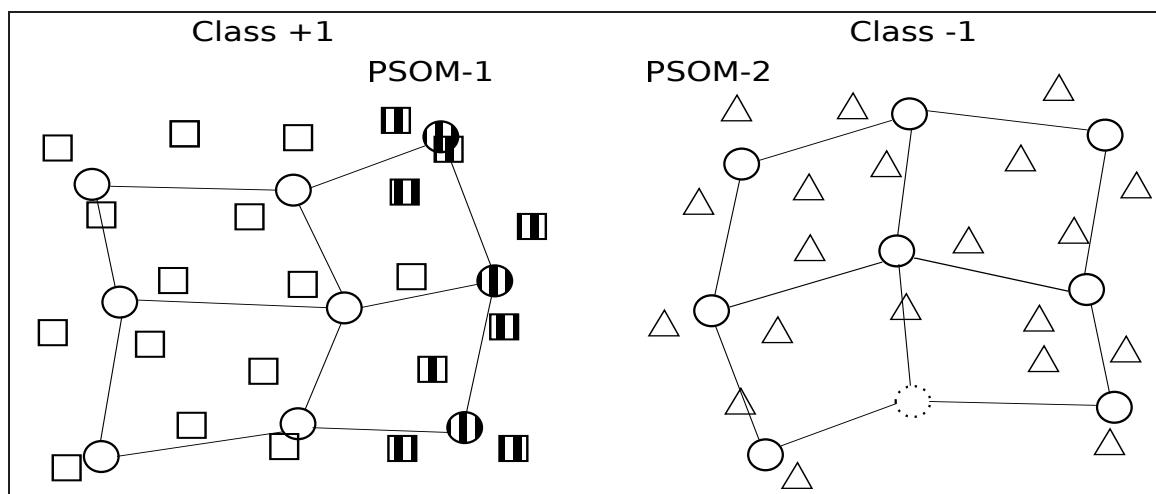


Figura 3.12: Para cada neurônio ativado no **Passo 4.2a**, encontrar seus K pontos mais próximos. Refere-se a este conjunto como $\mathcal{X}^{(2)}$. Equivale ao **Passo 6.2**.

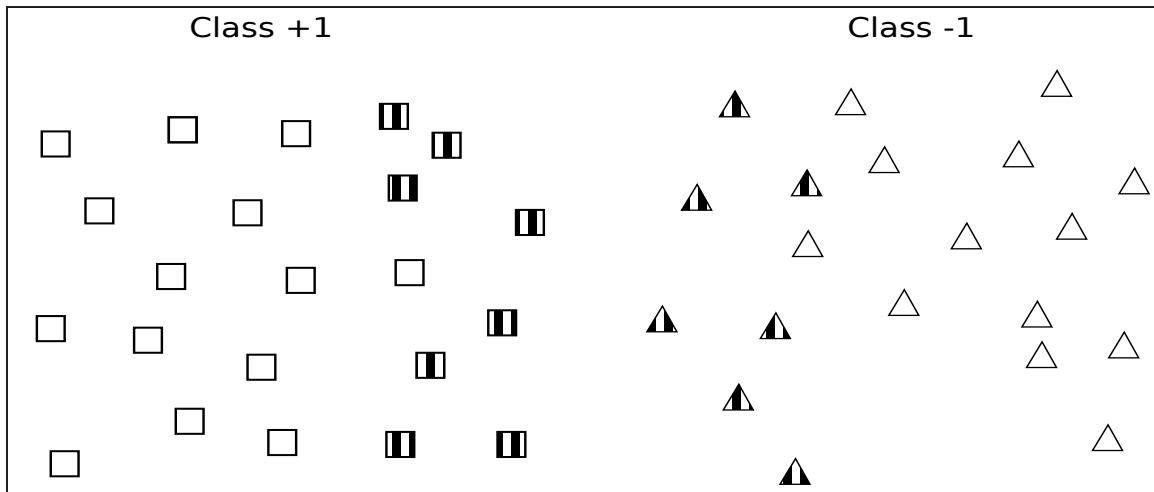


Figura 3.13: O conjunto reduzido de padrões é dado por $\mathcal{X}_{rs} = \mathcal{X}^{(1)} \cup \mathcal{X}^{(2)}$. **Passo Final.**

3.4.2 Classificador OM-SVM

A principal idéia nesta combinação é entregar ao algoritmo SMO um “problema quase-resolvido”, pois considera-se que todos os padrões que não estão contidos no conjunto reduzido (i.e. $\mathbf{x}_i \notin \mathcal{X}_{rs}$) devem possuir multiplicadores de Lagrange com valores iguais a zero $\{\alpha_i = 0\}_{i \notin \mathcal{X}_{rs}}$. Assim, realiza-se o treinamento com base no conjunto obtido via método OM, e não mais com base em todos os padrões de treinamento. Neste contexto, a seleção do par de vetores, necessária a cada iteração do processo de aprendizagem do algoritmo SMO, é realizada com base nos padrões contidos no conjunto reduzido \mathcal{X}_{rs} obtido pelo método OM. Portanto, o algoritmo SMO deve apenas calcular os valores dos multiplicadores de Lagrange e excluir mais alguns vetores que porventura não deveriam pertencer ao conjunto final de vetores-suporte. Ao classificador resultante dá-se o nome de OM-SVM.

3.4.3 Classificador OM-LSSVM

Deve-se considerar, no tocante à combinação do método OM com os classificadores LS-SVM, que os multiplicadores de Lagrange e o viés estão contidos no vetor \mathbf{x} do sistema linear $\mathbf{Ax} = \mathbf{B}$. Neste sistema, as colunas que estão associadas aos multiplicadores devem ser eliminadas, visto que a eliminação de uma coluna equivale a tornar o multiplicador de Lagrange associado com valor igual a zero. No entanto, vale destacar que as linhas da matriz A , as quais descrevem às restrições impostas ao problema quadrático, não devem ser eliminadas, pois isto pode levar a perda de desempenho (LEE; MANGASARIAN, 2001b; VALYON; HORVÁTH, 2004).

Do exposto acima, utiliza-se o método OM para construir uma versão modificada da versão

LSSVM padrão. Ao invés de construir um matriz quadrada original \mathbf{A} e em seguida realizar uma operação de inversão nela, a fim de obter o vetor \mathbf{x} , deve-se construir uma matriz reduzida não quadrada \mathbf{A}_{rs} considerando apenas os vetores pertencentes ao conjunto reduzido \mathcal{X}_{rs} . Uma vez que \mathbf{A}_{rs} é uma matriz não-quadrada, pode-se resolver o sistema e obter \mathbf{x} pelo método da pseudo-inversa, $\mathbf{x} = \mathbf{A}^* \mathbf{B}$, tal que:

$$\mathbf{A}^* = (\mathbf{A}_{rs}^T \mathbf{A}_{rs})^{-1} \mathbf{A}_{rs}^T. \quad (3.10)$$

3.5 Generalized Opposite Maps

Como dito anteriormente, a idéia intrínseca ao método OM pode ser generalizada para qualquer algoritmo de quantização vetorial. Assim, com o intuito de generalizar o método OM utilizam-se o algoritmo *K-Means* e a rede GNG. Como o método OM foi proposto inicialmente para ser utilizado com a rede de Kohonen, o termo “mapa” foi mantido na nomenclatura da versão estendida, que passa a ser identificada por Generalized Opposite Maps (GOM).

Vale destacar que todas as operações a serem realizadas com todos os algoritmos de quantização vetorial supracitados transcorrem no espaço de entrada. Por conta disto, os classificadores SVM e LSSVM baseados no método GOM devem necessariamente utilizar um *kernel* linear, pois nesta situação as operações realizadas internamente aos classificadores também transcorrem no espaço de entrada. Nesse caso, os classificadores resultantes do uso do algoritmo *K-Means* e da rede GNG para treinar o classificador SVM são denotados por GOM-SVM/KM e GOM-SVM/GNG, respectivamente. Enquanto para os classificadores resultantes do uso do algoritmo *K-Means* e da rede GNG para treinar o classificador LSSVM são denotados por GOM-LSSVM/KM e GOM-LSSVM/GNG, respectivamente.

A fim de permitir que outros tipos de *kernel* (além do linear) sejam aplicados, pode-se optar por algoritmos que realizem a quantização vetorial no espaço de características. Nesta tese, utiliza-se o algoritmo *Kernel k-Means* (ZHANG; RUDNICKY, 2002). Neste caso, as operações de obtenção do conjunto aproximado de vetores-suporte ocorrem no espaço de características. Por fim, vale enfatizar que os parâmetros da função *kernel* devem ser os mesmos tanto nos classificadores SVM e LSSVM, quanto no algoritmo *Kernel k-Means*. Mais detalhes sobre os algoritmos *K-Means* e *Kernel k-Means* encontram-se no Apêndice B.

Resumidamente, os métodos e classificadores propostos neste capítulo são listados na Tabela 3.1, onde também são apresentados as principais características de cada classificador.

Os classificadores propostos são comparados com os classificadores SVM e LSSVM con-

Propostas	Descrição	Kernel
Método OM	Obtém VS candidados com base na rede Kohonen	Linear
Método GOM	Obtém VS candidados com base em algoritmos de quantização vetorial	Linear/RBF
OM-SVM	Classificador SVM, treinado pelo algoritmo SMO e método OM	Linear
OM-LSSVM	Classificador LSSVM, treinado pela pseudo inversa (PI) e método OM	Linear
GOM-SVM/KM	Classificador SVM, treinado pelo SMO e método GOM (<i>K-Means</i>)	Linear
GOM-LSSVM/KM	Classificador LSSVM, treinado pela PI e método GOM (<i>K-Means</i>)	Linear
GOM-SVM/GNG	Classificador SVM, treinado pelo SMO e método GOM (rede GNG)	Linear
GOM-LSSVM/GNG	Classificador LSSVM, treinado pela PI e método GOM (rede GNG)	Linear
GOM-SVM/K ² M	Classificador SVM, treinado pelo SMO e método GOM (Kernel K-Means)	RBF
GOM-LSSVM/K ² M	Classificador LSSVM, treinado pela PI e método GOM (Kernel K-Means)	RBF

Tabela 3.1: Métodos (OM e GOM) e classificadores propostos nesta capítulo.

vencionais, bem como são comparados com os classificadores GNG-SVM e GNG-LSSVM. Este último classificador, também proposta desta tese, obtém os vetores-suporte segundo o trabalho de Linda & Manic (2009) e utiliza a matriz inversa para calcular os valores do vetor de multiplicadores de Lagrange e do viés.

3.6 Simulações Computacionais

Nesta seção são apresentados os resultados obtidos para os classificadores SVM e LSSVM, treinados a partir de conjuntos reduzidos construídos pelo método OM e por sua versão generalizada (i.e. método GOM). Para fins de comparação, são apresentados ainda os resultados obtidos para o classificador GNG-SVM. Inicialmente são apresentados os resultados obtidos pela aplicação direta do método OM a problemas artificiais. Os métodos OM e GOM, bem como todos os classificadores avaliados foram implementados utilizando a linguagem Java no ambiente Eclipse.

3.6.1 Resultados para Conjuntos de Dados Artificiais

Os resultados obtidos referentes à aplicação do método OM em dois problemas artificiais diferentes podem ser vistos nas Figuras 3.14 e 3.15. No conjunto de dados mostrado na Figura 3.14, o método OM busca os vetores-suporte para um problema linearmente separável. Na Figura 3.14(a) são mostrados os exemplos (pontos) de cada classe. Na Figura 3.14(b) são mostrados os exemplos de cada classe, candidatos a vetores-suporte (quadrados e triângulos preenchidos). Na Figura 3.14(b) também aparecem em destaque os protótipos (asteriscos) que atraem os padrões da classe oposta e os padrões associados a estes protótipos.

No segundo problema, apresentado na Figura 3.15, verifica-se a aplicação do método OM a

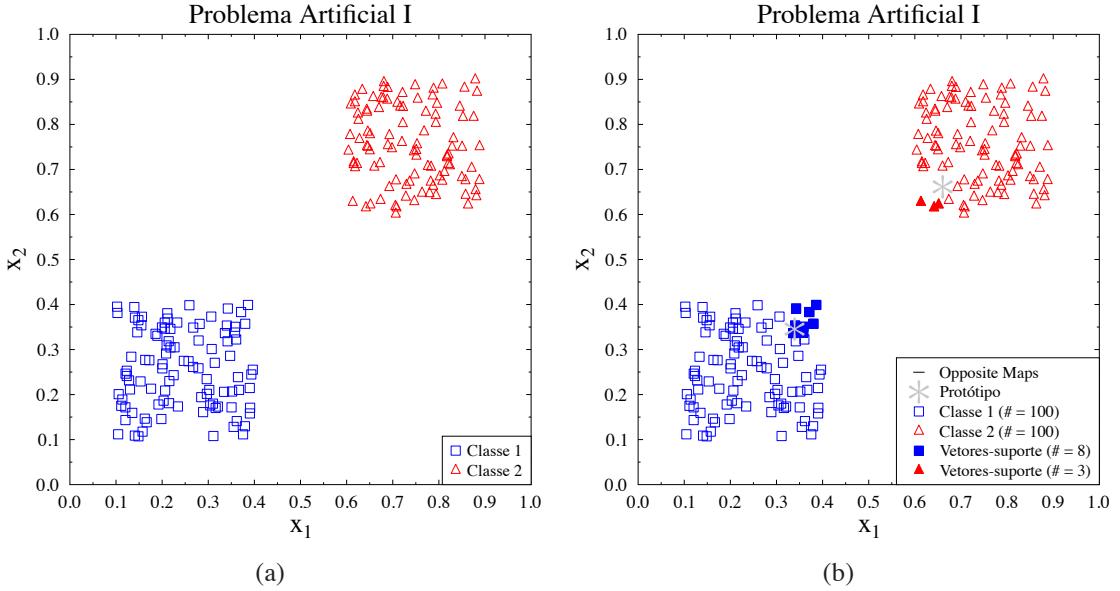


Figura 3.14: (a) Problema artificial linearmente separável. (b) Resultado do método OM para o problema Artificial I.

um problema não linearmente separável. Tanto nesta figura quanto na anterior, pode-se constatar a capacidade de obtenção dos vetores localizados nas regiões de fronteira das classes ou dos vetores localizados na região de sobreposição entre as classes. Como descrito anteriormente, há uma maior probabilidade dos vetores-suporte estarem dispostos nesta região ou nas bordas mais próximas da outra classe.

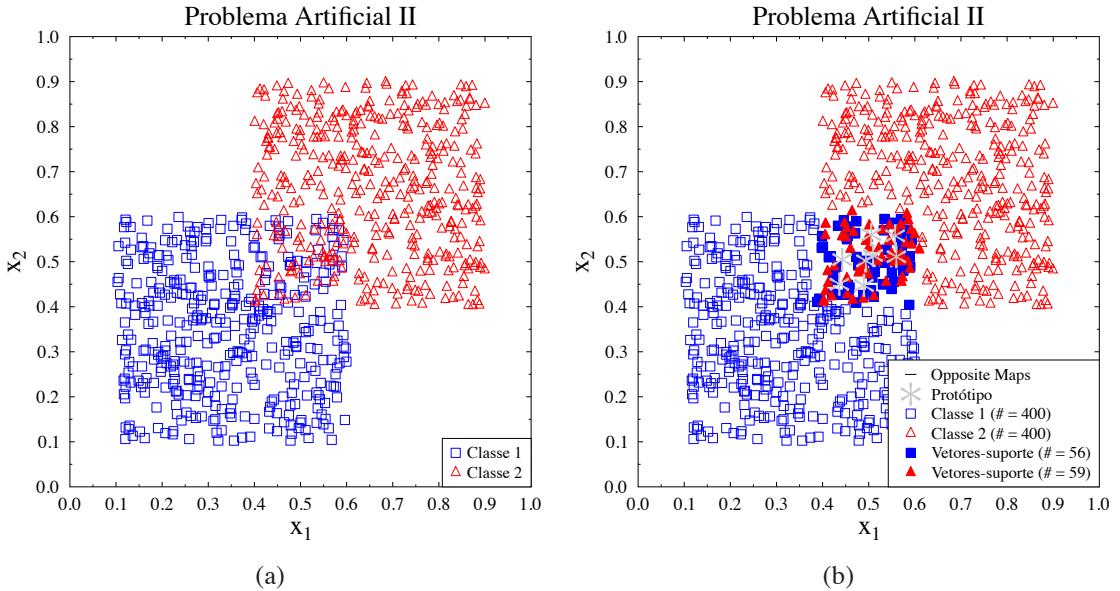


Figura 3.15: (a) Problema artificial não-linearmente separável. (b) Resultado do método OM para o problema Artificial II.

As Figuras 3.14 e Figura 3.15 foram geradas pelo método OM usando duas redes SOM, uma para cada classe. A Tabela 3.2 apresenta os valores dos parâmetros de treinamento de cada uma das redes SOM para os problemas Artificial I e Artificial II.

Parâmetro	Artificial I	Artificial II	Artificial III	Artificial IV
dimensões da grade	5×5	5×5	8×8	15×15
taxa de aprendizado inicial	0,5	0,5	0,5	0,5
taxa de aprendizado final	0,01	0,01	0,1	0,1
raio de vizinhança inicial	5,0	5,0	5,0	5,0
raio de vizinhança final	0,1	0,1	0,1	0,1
número máximo de épocas	20	80	200	20
tamanho conjunto de dados	200	800	800	700

Tabela 3.2: Parâmetros de treinamento da rede SOM para os problemas Artificial I, Artificial II, Artificial III e Artificial IV.

Outros resultados relacionados ao método OM são apresentados nas Figuras 3.16 e 3.17. Na primeira, tem-se um problema também linearmente separável, enquanto na segunda tem-se um problema não-linearmente separável tal que os padrões da classe mais interna estão envoltos por padrões da classe mais externa. A Tabela 3.2 também apresenta os valores dos parâmetros de treinamento da rede SOM para os problemas Artificial III e Artificial IV.

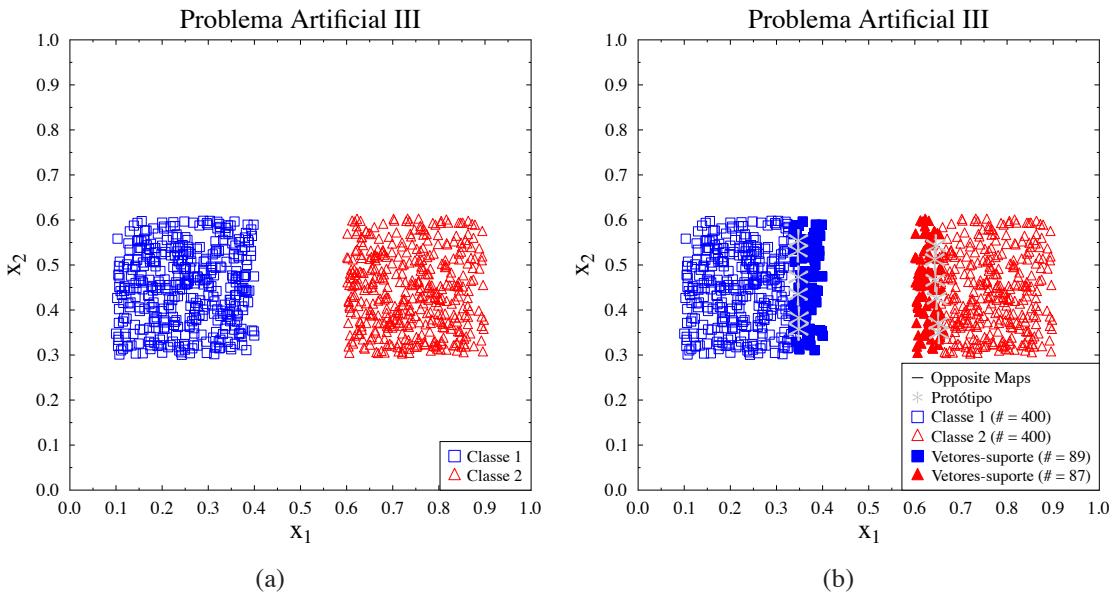


Figura 3.16: (a) Problema artificial linearmente separável. (b) Resultado do método OM para o problema Artificial III.

Inicialmente, como prova de conceito, aplicou-se o classificador OM-SVM ao problema artificial apresentado na Figura 3.14. Os resultados mostrados na Figura 3.18 indicam que o classificador OM-SVM produz uma superfície linear de decisão equivalente ao SVM padrão. Além disto, pode-se perceber que o classificador OM-SVM possui uma quantidade menor de vetores-suporte que o classificador SVM, sendo cinco vetores-suporte para o classificador OM-SVM e onze vetores-suporte para o classificador SVM. Os parâmetros para as redes SOM do classificador OM-SVM são apresentados na Tabela 3.2, na coluna do problema Artificial I,

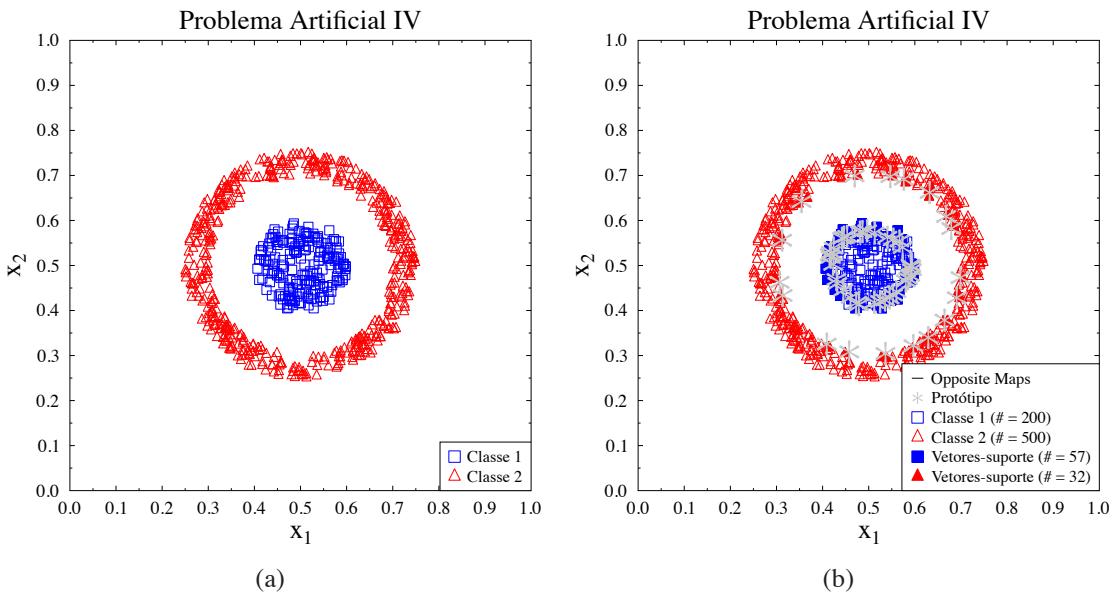


Figura 3.17: (a) Problema artificial não-linearmente separável, em que os dados de uma classe estão envoltos pelos dados da outra classe. (b) Resultado do método OM para o problema Artificial IV.

enquanto os valores dos demais parâmetros do classificador OM-SVM são apresentados na Tabela 3.3.

As superfícies de decisão geradas pelos classificadores SVM/RBF e GOM-SVM/K²M/RBF para outro problema artificial são apresentadas na Figura 3.19(a) e 3.19(b), respectivamente. O algoritmo K²-Médias para o classificador GOM-SVM/K²M/RBF é treinado utilizando 100 protótipos durante 20 épocas. Os demais valores dos parâmetros para o classificador GOM-SVM/K²M/RBF são apresentados na Tabela 3.3. Percebe-se que o classificador GOM-SVM/K²M/RBF apresenta uma quantidade menor de vetores-suporte (48 no total) quando comparado com o classificador SVM-RBF padrão com 58 vetores-suporte. Nota-se ainda que as superfícies de decisão de ambos são bastante similares. Esta situação demonstra a capacidade do método OM de manter a qualidade dos classificadores gerados no processo de aprendizagem e ao mesmo tempo reduzir a quantidade de vetores-suporte.

Parâmetro	OM-SVM	GOM-SVM/K ² M
C	2,5	2,5
Kernel	Linear	RBF[$\sigma = 1,0$]
Tolerância	0,001	0,01

Tabela 3.3: Parâmetros dos classificadores OM-SVM e GOM-SVM/K²M.

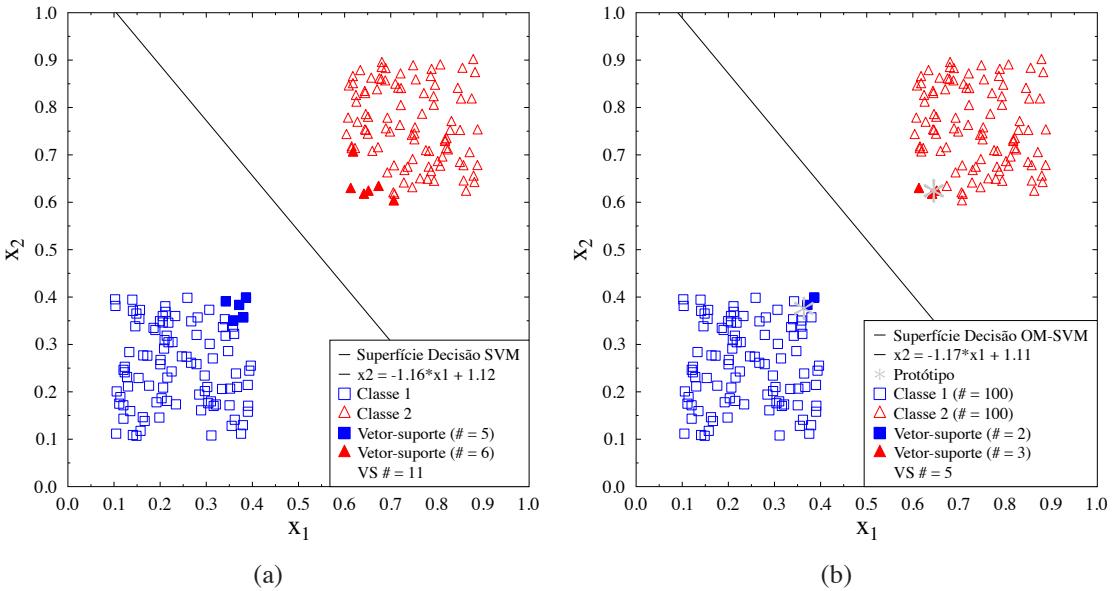


Figura 3.18: (a) Superfície de decisão e número de vetores-suporte para a SVM padrão treinada com o algoritmo SMO. (b) Superfície de decisão e quantidade de vetores-suporte para o classificador OM-SVM.

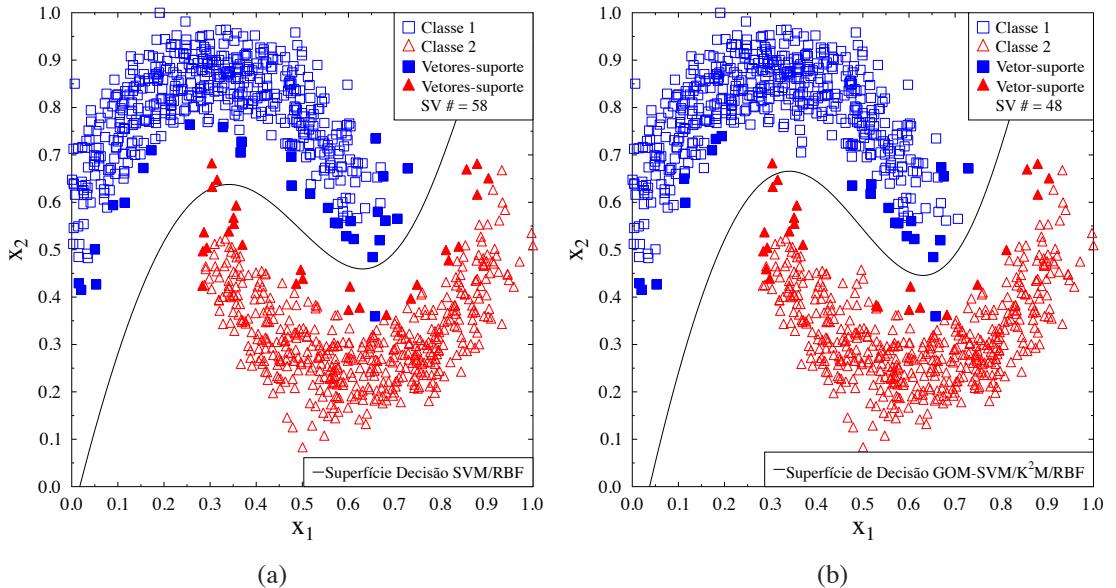


Figura 3.19: (a) Superfície de decisão e os vetores-suporte para o classificador SVM padrão com *kernel* RBF. (b) Superfície de decisão e os vetores-suporte obtidos do classificador GOM-SVM/K²M/RBF.

3.6.2 Resultados para o problema PCV-2C

Para todos os experimentos a serem apresentados nesta subseção, 80% do conjunto de dados PCV-2C é selecionado aleatoriamente para propósito de treinamento. O restante 20% dos dados são utilizados para verificação do desempenho de generalização dos diversos classificadores. Além dos resultados apresentados nesta subseção para o conjunto PCV-2C, pode-se observar outros resultados para o método OM quando aplicado aos conjuntos Diabetes (UCI, 2011b) e

Câncer de Mama (UCI, 2011a) no trabalho de Rocha-Neto & Barreto (2011).

As simulações envolvendo os classificadores OM-SVM e OM-LSSVM foram conduzidas usando uma grade bidimensional, vizinhança hexagonal, função de vizinhança gaussiana, com inicialização randômica dos pesos. A simulação baseia-se em uma rede SOM formada por uma grade 10×10 , com uma quantidade máxima de épocas igual a 250 e raio de vizinhança (taxa de aprendizado) inicial e final de 5 (0.5) e 0.1 (0.01), respectivamente. Como descrito anteriormente, a resolução do problema binário utiliza dois mapas, cada um deles com a configuração descrita acima. No tocante aos classificadores GOM-SVM/KM e GOM-LSSVM/KM, as simulações foram conduzidas considerando-se um $k = 100$ e uma quantidade máxima de épocas igual a 200. Enquanto as simulações envolvendo o GOM-SVM/GNG e GOM-LSSVM/GNG foram conduzidas usando um número máximo de protótipos igual a 100 e um número máximo de épocas igual a 200. Um novo protótipo é adicionado a cada 500 iterações do algoritmo e a taxa de aprendizado do vencedor apresenta um decaimento exponencial, com valor inicial de 0,50 e valor final de 0,05. As taxas de aprendizado do vizinhos também possuem decaimento exponencial, com valor inicial e final de 0,25 e 0,005, respectivamente.

Os resultados para os classificadores SVM, OM-SVM, GOM-SVM/KM, GOM-SVM/GNG e GNG-SVM são mostrados na Tabela 3.4. Enquanto na Tabela 3.5 são mostrados os resultados para os classificadores LSSVM, OM-LSSVM, GOM-LSSVM/KM, GOM-LSSVM/GNG e GNG-LSSVM. Nestas tabelas são apresentadas métricas de desempenho, tais como o valor médio (acurácia) e desvio padrão da taxa de classificação no conjunto de teste, medidas sobre 50 rodadas independentes. São apresentados ainda o número médio de vetores-suporte (# VSs) dos classificadores obtidos durante as 50 rodadas realizadas, o tamanho do conjunto de treinamento (# Trein.), bem como os valores dos parâmetros C (SVM), γ (LSSVM) e a margem de tolerância (tol). Vale ressaltar que os resultados para os classificadores GNG-SVM são obtidos com base no procedimento descrito a seguir.

Classificador	Kernel	C	Tol.	Acurácia	# Trein.	# VSs	Redução
SVM	Linear	2,5	0,001	$84,9 \pm 3,9$	248	87,2	
GOM-SVM/KM	Linear	2,5	0,001	$85,6 \pm 4,1$	248	71,9	17,5%
OM-SVM	Linear	2,5	0,001	$84,9 \pm 4,8$	248	70,1	19,6%
GOM-SVM/GNG	Linear	2,5	0,001	$85,5 \pm 4,8$	248	70,2	19,5%
GNG-SVM	Linear	2,5	0,001	$80,8 \pm 3,8$	248	70,3	19,4%

Tabela 3.4: Resultados para os classificadores SVM, OM-SVM, GOM-SVM/KM, GOM-SVM/GNG e GNG-SVM para o conjunto PCV-2.

Classificador	Kernel	γ	Acurácia	# Trein.	# VSs	Redução
LSSVM	Linear	0,05	$80,4 \pm 4,3$	248	248,0	
OM-LSSVM	Linear	0,05	$81,9 \pm 4,4$	248	111,1	55,2%
GOM-LSSVM/KM	Linear	0,05	$81,8 \pm 4,8$	248	105,7	57,4%
GOM-LSSVM/GNG	Linear	0,05	$80,3 \pm 5,0$	248	104,5	57,9%
GNG-LSSVM	Linear	0,05	$74,4 \pm 5,4$	248	106,0	57,3%

Tabela 3.5: Resultados para os classificadores LSSVM, OM-LSSVM, GOM-LSSVM/KM, GOM-LSSVM/GNG e GNG-LSSVM para o conjunto PCV-2C.

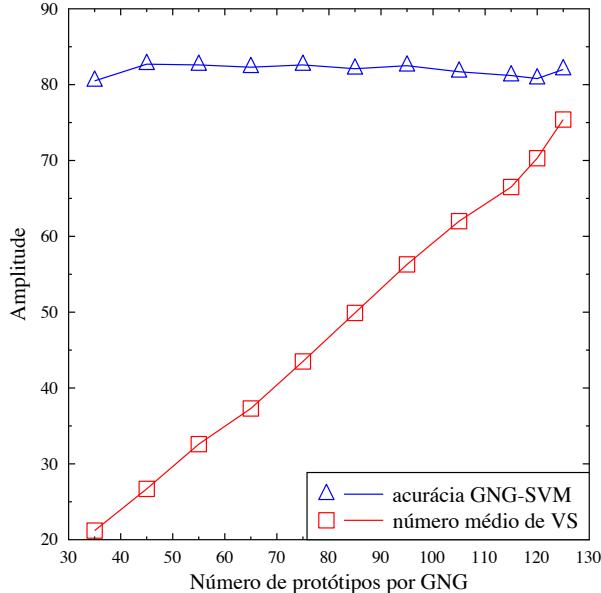


Figura 3.20: Evolução do número médio de vetores-suporte (série com quadrados) e do acerto médio (série com triângulos) em função do número de protótipos por rede GNG, no classificador GNG-SVM, para o problema PCV-2C.

Para o classificador GNG-SVM é implementado um aumento gradual no número de protótipos presentes em cada uma das duas redes GNG até que se alcance a quantidade média de vetores-suporte ou a taxa de classificação no teste do classificador OM-SVM. Por exemplo, inicia-se com 35 protótipos em cada rede, ou seja, considera-se uma base de dados com 70 padrões. Em seguida, aumenta-se o tamanho para 45, e portanto utiliza-se uma base contendo 90 padrões de treinamento; depois com 55 protótipos em cada rede, e por isto utiliza-se 110 padrões de treinamento; e assim por diante. Para cada quantidade de protótipos são realizadas 50 rodadas independentes. O gráfico apresentado na Figura 3.20 mostra os resultados obtidos para o problema PCV-2C. Pode-se observar no gráfico o número de protótipos (padrões) do conjunto de treinamento, o número médio de vetores-suporte e o acerto médio no teste.

Pode-se observar ainda na Figura 3.20 que a taxa de classificação no teste mantém-se com pouca variação, mesmo com um aumento significativo no número de protótipos por rede GNG. A acurácia para 120 protótipos por rede GNG, com um número médio de vetores-suporte de 68,9, é igual a 80,8. A maior taxa de classificação desta série é de 82,7 quando com 45 protóti-

pos em que cada uma das redes o GNG-SVM possui um número médio de vetores-suporte igual a 26,7. Nota-se que as taxas de classificação dos classificadores OM-SVM, GOM-SVM/KM e GOM-SVM/GNG apresentam-se bastante superiores às do classificador GNG-SVM.

A partir da análise das Tabelas 3.4 e 3.5, pode-se concluir que os desempenhos dos classificadores propostos para redução de vetores-suporte são equivalentes aos classificadores SVM e LSSVM. Em alguns casos, como mostrado na Tabela 3.5 para os conjunto PCV-2C, os desempenhos dos classificadores OM-LSSVM e GOM-LSSVM são até mesmo melhores que o classificador LSSVM. Verifica-se ainda na Tabela 3.4 que para o conjunto PCV-2C o classificador GNG-SVM apresenta pior desempenho do que os outros classificadores avaliados (e.g OM-SVM e GOM-SVM). O mesmo pode ser afirmado, ao se analisar a Tabela 3.5, para o classificador GNG-LSSVM quando comparado com os classificadores GOM-SVM e GOM-LSSVM para o problema PCV-2C.

Na Figura 3.21 apresenta-se a evolução da quantidade média de vetores-suporte e da acurácia dos classificadores SVM e GOM-SVM/KM, em função do tamanho do conjuntos de treinamento. Percebe-se que há, na prática, um aumento linear do número de vetores-suporte, que o GOM-SVM/KM sempre possui uma menor quantidade média de vetores-suporte e que, a partir de 60% de conjunto de treinamento, a acurácia do classificador GOM-SVM/KM apresenta-se superior à do SVM. Em valores inferiores a 60%, ainda verifica-se uma significativa redução da quantidade média de vetores-suporte, bem como uma pequena diferença em termos de acurácia. Estes resultados demonstram que o classificador GOM-SVM/KM é bastante robusto para treinamentos com pequenas parcelas de dados.

As curvas ROC obtidas a partir dos classificadores SVM e OM-SVM são apresentadas na Figura 3.22. Por analisar os valores das áreas sob a curva (valores AUC), pode-se notar novamente que o desempenho dos classificadores padrão e com conjunto reduzido são equivalentes.

Na Tabela 3.6 são apresentados os resultados dos classificadores SVM e GOM-SVM/K²M, que utilizam o *kernel* RBF. Nesta tabela também são mostrados os parâmetros de treinamento dos classificadores, bem como a taxa de classificação, o tamanho do conjunto de treinamento, e o percentual de redução de vetores-suporte. Os resultados foram obtidos após a realização de 20 rodadas independentes e o critério de parada do algoritmo *Kernel K-Médias* foi de 20 épocas de treinamento.

Similarmente, na Tabela 3.7, são descritos os resultados obtidos para os classificadores LS-SVM e GOM-LSSVM/K²M, que utilizam o *kernel* RBF. Os parâmetros do *kernel*, a tolerância da margem, o desempenho e o número médio de vetores-suporte estão também contidos nesta

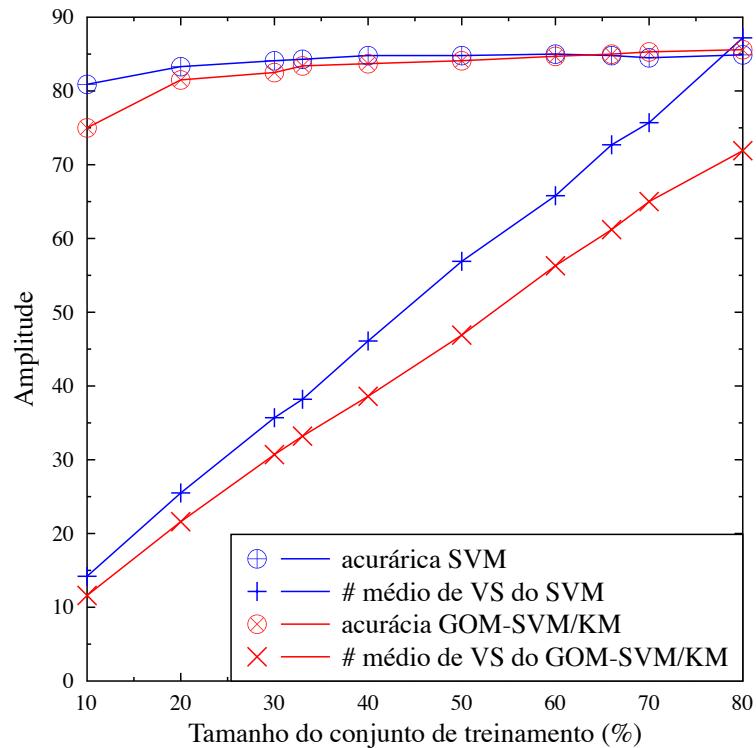


Figura 3.21: Gráfico comparativo entre os classificadores SVM e OM-SVM para o problema PCV-2C quando os classificadores são treinados com diferentes percentuais do conjunto de treinamento.

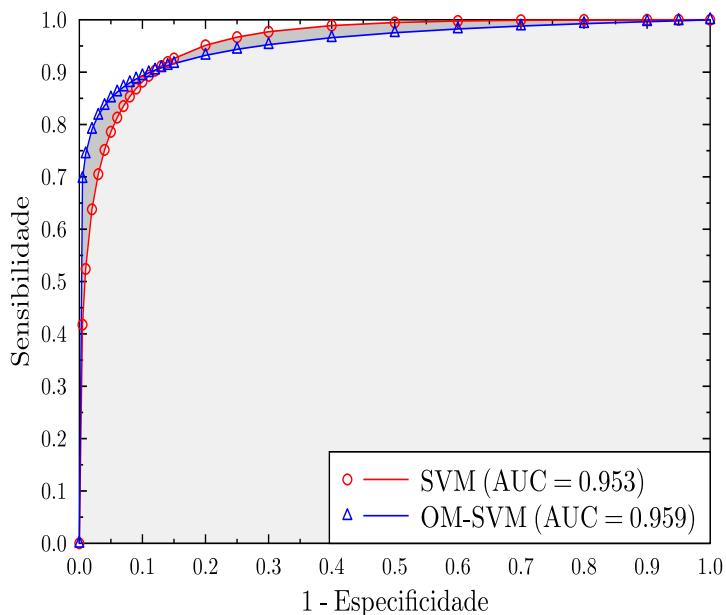


Figura 3.22: Curva ROC obtida a partir dos classificadores SVM e OM-SVM para o problema da coluna vertebral (PCV-2C).

Base	Classificador	Kernel	C	Tol.	Acurácia	# Trein.	# VSs	Redução
PCV-2C	SVM	RBF(1.5)	2.5	0.1	85.3 ± 4.0	248	124.4	
PCV-2C	GOM-SVM/K ² M	RBF(1.5)	2.5	0.1	84.6 ± 3.5	248	99.9	19.7%

Tabela 3.6: Resultados para os classificadores SVM/RBF e GOM-SVM/K²M/RBF.

Base	Classificador	Kernel	γ	Acurácia	# Trein.	# VSs	Redução
PCV-2C	LSSVM	RBF(1.0)	0.3	84.4 ± 3.7	248	248.0	
PCV-2C	GOM-LSSVM/K ² M	RBF(1.0)	0.3	85.1 ± 4.5	248	116.6	53.0%

Tabela 3.7: Resultados para os classificadores LSSVM/RBF e GOM-LSSVM/K²M/RBF.

tabela. O critério de parada do algoritmo *Kernel K-Médias* foi de 20 épocas de treinamento.

Percebe-se, das Tabelas 3.6 e 3.7, que também no espaço de características o método OM consegue reduzir significativamente o número de vetores-suporte dos classificadores treinados, bem como consegue obter desempenho semelhante ou até mesmo superior ao dos classificadores convencionais.

As curvas ROC dos classificadores SVM/RBF e GOM-SVM/K²M/RBF são apresentadas na Figura 3.23. Percebe-se que os valores de AUC para os classificadores SVM/RBF e GOM-SVM/K²M/RBF são bastante próximos, demonstrando que o método GOM permite obter classificadores com desempenho equivalente e, ao mesmo tempo, com uma quantidade reduzida de vetores-suporte.

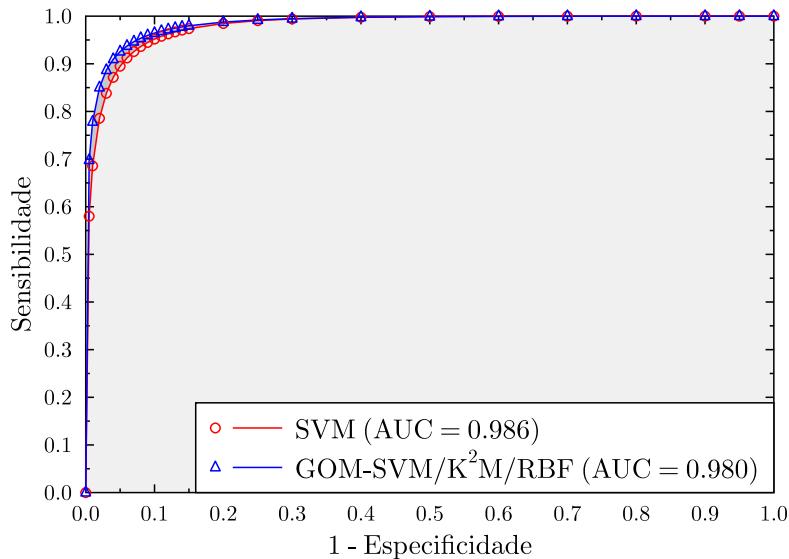


Figura 3.23: Curvas ROC para os classificadores SVM/RBF e GOM-SVM/K²M/RBF para o problema PCV-2C.

3.7 Conclusão

Nesta seção foi proposto e validado o método Opposite Maps que visa a obtenção de um conjunto reduzido de vetores-suporte. Nesta tese, foram propostos, implementados e validados os classificadores OM-SVM, GOM-SVM/KM, GOM-SVM/GNG; bem como os classificadores OM-LSSVM, GOM-LSSVM/KM e GOM-LSSVM/GNG, todos derivados dos métodos OM e GOM. Os diversos classificadores avaliados foram capazes de obter uma redução significativa do número de vetores-suporte, com redução de cerca de 17% (como no caso dos classificadores GOM-SVM/KM) ou de até cerca de 58% (como no caso dos classificadores GOM-LSSVM/GNG).

Estes classificadores estão aptos a utilizar um *kernel* linear, pois as operações do método OM ocorrem no espaço de entrada. No entanto, propõe-se também uma extensão do método OM para permitir que se utilize um outro tipo de *kernel*, RBF, e desta forma foram propostos e implementados os classificadores GOM-SVM/ K^2M e GOM-LSSVM/ K^2M . Isto foi possível em virtude da implementação do método OM com suas operações transcorrendo no espaço de características a partir do algoritmo *Kernel K*-Médias.

Comparativamente, o classificador GNG-SVM apresenta desempenho inferior aos dos classificadores OM-SVM quando aplicados ao problema PCV-2C. Comparando os classificadores OM-LSSVM e GNG-LSSVM, também pode-se perceber que a acurácia desse para o problema PCV-2C é inferior à daquele. De uma forma geral, os classificadores propostos para redução do número de vetores-suporte apresentam melhores resultados que os obtidos dos classificadores GNG-SVM e GNG-LSSVM, respectivamente.

Os classificadores GOM-SVM/ K^2M /RBF e GOM-LSSVM/ K^2M também apresentam acurárias similares aos dos classificadores SVM e LSSVM, respectivamente; bem como apresentam redução média de vetores-suporte bastante significativa e em proporções bastante semelhantes às dos classificadores com *kernel* linear.

De uma forma geral, os classificadores SVM e LSSVM treinados com conjunto reduzido de vetores de treinamento, apresentam desempenhos bastante similares aos dos classificadores que usam todo o conjunto de dados de treinamento disponível mesmo com quantidades bem inferiores de vetores-suporte. Esta afirmação apenas não é válida quando são comparados os classificadores SVM e LSSVM com os classificadores GNG-SVM e GNG-LSSVM, respectivamente; pois estes últimos apresentam desempenhos bastante inferiores.

4 *Classificação com Opção de Rejeição*

Neste capítulo são descritos os conceitos relacionados à classificação com opção de rejeição aplicada ao problema de classificação de patologias da coluna vertebral. Neste contexto, são apresentados os conceitos de taxa de rejeição de um classificador, regra de Chow, risco empírico em termos de rejeição, bem como diversas abordagens encontradas na literatura. As abordagens apresentadas baseiam-se em classificadores MLP e SVM. São também propostas abordagens que se baseiam nas redes SOM.

Os fundamentos teóricos necessários ao completo entendimento dos classificadores propostos neste capítulo encontram-se nos trabalhos de Chow (1970), Fumera & Roli (2002), Cardoso & Costa (2007) e Rocha-Neto et al. (2011). Vale ressaltar também que os resultados obtidos neste capítulo foram consequência de trabalho conjunto realizado com os pesquisadores Ricardo Souza e Jaime S. Cardoso.

4.1 Introdução

Em um problema binário, um classificador geralmente define uma saída $y = +1$ ou $y = -1$ com base em estimativas de probabilidades a posteriori $p(C_i|\mathbf{x})$, para um dado vetor de entrada \mathbf{x} . Isto acontece mesmo quando a diferença entre os valores das probabilidades a posteriori das classes é pequena. Como consequência, a automatização das decisões mais difíceis¹ pode conduzir a previsões erradas e, portanto, à elevação do erro de classificação.

Nesse tipo de sistema, os vetores de entrada \mathbf{x} costumam ser rotulados apenas como “bom” ($+1$) ou “ruim” (-1), ou mesmo como “normal” ou “anormal” para problemas de diagnóstico de patologias. No entanto, em muitos ambientes, pode ser desejável que tais sistemas sejam capazes de rejeitar casos críticos, que causam maior confusão, delegando-os assim para a avaliação de um especialista. Nesse sentido, uma nova classe pode ser definida, chamada de classe de rejeição, tal que seus padrões dispõem-se na região entre as classes “bom” e “ruim” (ou

¹Decisões em que as probabilidades a posteriori encontram-se muito próximas ao limiar.

“normal” e “anormal”).

O trabalho de Chow (1957) foi o pioneiro no que concerne a classificação com opção de rejeição e envolveu a tarefa de reconhecimento de caracteres. No entanto, em um trabalho posterior, Chow (1970) descreve mais completamente o problema de classificação com opção de rejeição, sendo este então normalmente considerado como o trabalho clássico inicial na área. Em Friedel et al. (2006) é demonstrado que a inclusão da opção de rejeição aumenta consideravelmente a acurácia dos classificadores. Em um outro trabalho recente, Hanczar & Dougherty (2008) utilizam a classificação com opção de rejeição em dados de *microarray*. Neste artigo é reforçada a idéia de que o desempenho do classificador depende de sua acurácia e da inclusão de mecanismos de rejeição.

O trabalho de Fumera & Roli (2002) apresenta um método para classificação com opção de rejeição baseado no classificador SVM, porém, diferentemente das abordagens propostas anteriormente, a formulação do problema permite que o classificador incorpore a capacidade de rejeitar como parte do processo de treinamento. Mais recentemente, foi proposta por Sousa et al. (2009a) uma outra estratégia para rejeição baseada no classificador *Ordinal SVM* (CARDOSO; COSTA, 2007) e em um método de replicação de dados. Esta estratégia, denominada *rejoSVM*, também adquire a capacidade de rejeitar como parte do processo de treinamento do classificador e é baseada no classificador. Além destes trabalhos que utilizam o classificador SVM para rejeição, também são encontrados na literatura a utilização de métodos que baseiam-se na rede MLP (CORDELLA et al., 1995; FUMERA et al., 2000; LIU et al., 2002). Uma das contribuições desta tese consiste na aplicação de classificação com opção de rejeição ao problema de diagnóstico de patologias da coluna vertebral (problema PCV-2C) (ROCHA-NETO et al., 2011).

4.2 Fundamentação Teórica

A taxa de erro de um classificador, calculada como o número de decisões errôneas em relação ao total de decisões, é a medida mais comumente utilizada para avaliar o desempenho de sistemas de reconhecimento de padrões. Uma outra medida importante, porém bem menos comum, é a taxa de rejeição, definida como a quantidade de decisões rejeitadas em relação ao número total de decisões realizadas.

Um erro ocorre quando um padrão de determinada classe é identificado como pertencente a outra. Uma rejeição ocorre quando o sistema de reconhecimento evita tomar uma decisão, rejeitando o padrão atual, para que o mesmo seja posteriormente alvo de um exame minucioso

por parte de especialistas na área de interesse. Assim, do ponto de vista de Chow (1970), considera-se que uma descrição mais adequada do desempenho de um sistema de tomada de decisão é dada pelo balanceamento entre a sua taxa de erro e a sua taxa de rejeição (*tradeoff error-reject*).

Por conta das incertezas e dos ruídos inerentes às tarefas de reconhecimento de padrões, erros são inevitáveis. A opção de rejeição é inserida para salvaguardar contra erros excessivos frutos de tomadas de decisão difíceis. Assim, o sistema deve converter potenciais erros em rejeição. Porém, vale destacar também que sempre que a classificação com opção de rejeição é utilizada algumas decisões potencialmente corretas podem vir a ser também convertidas em rejeição. Vale observar que os custos de errar e rejeitar raramente estão na proporção um-para-um. Logo, deve-se analisar a acurácia de predição para diferentes custos de rejeição, ou de forma semelhante, analisar a rejeição para diferentes desempenhos do classificador, tal que se permita escolher o melhor classificador com base na acurácia desejável ou mesmo em uma taxa de rejeição aceitável, sendo esta escolha dependente do tipo de problema a ser abordado.

4.2.1 Regra de Decisão Ótima

Uma regra de decisão para classificação com opção de rejeição é ótima se para uma determinada taxa de erro (ou probabilidade de erro) também se minimiza a taxa de rejeição (ou probabilidade de rejeição). A regra de decisão ótima deve rejeitar um determinado padrão, se o maior valor das probabilidades a posteriori é menor do que algum limiar (CHOW, 1957).

Nas equações que se seguem $\mathbf{x} \in \mathbb{R}^N$ representa o padrão a ser analisado, C_i representa a i -ésima classe, c representa o número de classes, p_i representa a probabilidade a priori da i -ésima classe, $F(\mathbf{x}|C_k)$ descreve a função de verossimilhança dos dados da classe k , e o limiar t é uma constante entre 0 e 1 ($0 \leq t \leq 1$). Mais explicitamente, a regra de decisão ótima $\delta(d_k|\mathbf{x})$ é dada por

$$(a) \quad \delta(d_k|\mathbf{x}) = 1 \quad (k \neq 0), \quad (4.1)$$

$$(b) \quad \delta(d_0|\mathbf{x}) = 1 \quad (k = 0). \quad (4.2)$$

em que d_0 representa a decisão de rejeitar o padrão \mathbf{x} , e d_k representa a decisão de aceitar o padrão \mathbf{x} e atribuí-lo à classe $k = 1, \dots, c$, tal que c representa o número de classes do problema. A Equação (4.1) considera que se aceite o padrão \mathbf{x} para fins de reconhecimento e identifique-o como pertencente à k -ésima classe, se

$$p_k F(\mathbf{x}|C_k) \geq p_j F(\mathbf{x}|C_j), \quad \forall j = 1, \dots, c \quad (4.3)$$

e

$$p_k F(\mathbf{x}|C_k) \geq (1-t) \sum_{i=1}^c p_i F(\mathbf{x}|C_i). \quad (4.4)$$

A Equação 4.2, considera que se rejeite o padrão, se

$$\max_i \{p_i F(\mathbf{x}|C_i)\} < (1-t) \sum_{i=1}^c p_i F(\mathbf{x}|C_i). \quad (4.5)$$

Definindo $m(\mathbf{x})$, tal que

$$m(\mathbf{x}) = \frac{p_i F(\mathbf{x}|C_i)}{\sum_{i=1}^c p_i F(\mathbf{x}|C_i)}, \quad (4.6)$$

então, a regra ótima de reconhecimento pode ser simplificada e entendida como

1. aceitar o padrão \mathbf{x} , se

$$m(\mathbf{x}) \geq (1-t) \quad \text{ou} \quad (4.7)$$

2. rejeitar o padrão \mathbf{x} , se

$$m(\mathbf{x}) < (1-t). \quad (4.8)$$

Logo, a classificação com opção de rejeição considera a rejeição do padrão \mathbf{x} sempre que o maior valor entre as probabilidades a posteriori for pequeno, ou seja, menor que $(1-t)$ (CHOW, 1957).

O parâmetro t é denominado limiar de rejeição. Para qualquer valor deste limiar, a regra de decisão δ particiona o espaço de entrada em dois conjuntos (regiões) disjuntos, \mathcal{V}_A e \mathcal{V}_R , tal que

$$\mathcal{V}_A(t) = \{\mathbf{x} \mid m(\mathbf{x}) \geq (1-t)\} \quad (4.9)$$

e

$$\mathcal{V}_R(t) = \{\mathbf{x} \mid m(\mathbf{x}) < (1-t)\}. \quad (4.10)$$

A título de ilustração na Figura 4.1 são apresentadas as regiões descritas pelos conjuntos disjuntos \mathcal{V}_A e \mathcal{V}_R , que são denominadas região de aceitação e região de rejeição, respectivamente. Uma região de rejeição é definida no espaço de entrada e todos os exemplos dentro desta região são rejeitados pelo classificador. Nessa figura são ilustrados também os padrões que devem ser rejeitados para um dado limiar, como sendo aqueles que encontram-se na região sombreada.

Chow (1970) descreve duas propriedades importantes do limiar de rejeição. A primeira, refere-se à monotonicidade em t das taxas de erro e rejeição. A outra mostra que o parâmetro t é limitante superior da taxa de erro.

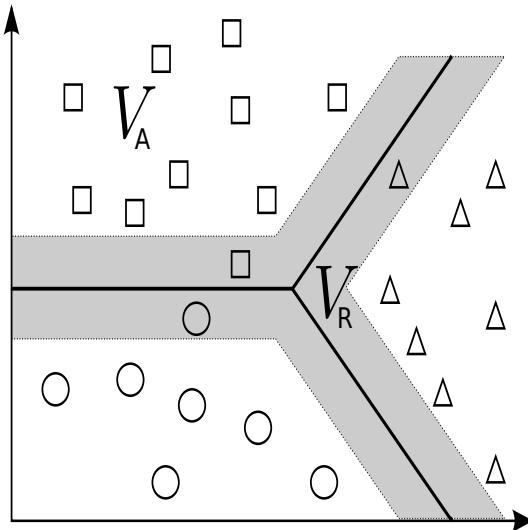


Figura 4.1: Ilustração das regiões de rejeição e aceitação da regra ótima de reconhecimento.

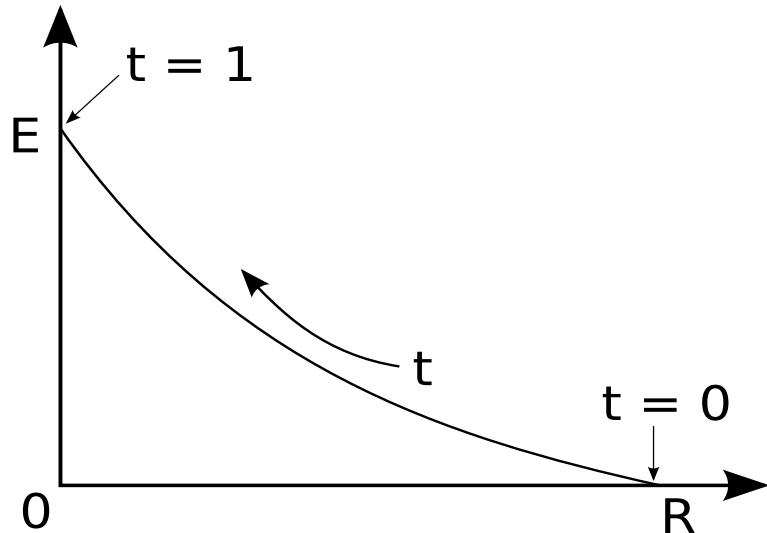


Figura 4.2: Ilustração da curva de compromisso entre o erro e a rejeição.

Como mencionado anteriormente, uma completa descrição do desempenho de um sistema de reconhecimento é dada por uma relação de compromisso entre erro e rejeição, nas mais diversas intensidades. Uma curva típica para tal relação é apresentada na Figura 4.2. Pode-se observar que à medida que a rejeição aumenta a taxa de erro diminui, e vice-versa. Desde que o erro (E) e rejeição (R) de um sistema de reconhecimento ótimo são funções monotônicas do limiar de rejeição t , pode-se computar o compromisso entre a taxa de erro e a taxa de rejeição para um determinado limiar.

No trabalho de Chow (1970) é mostrado ainda que a taxa de rejeição em função do limiar de rejeição é suficiente para descrever o desempenho do sistema de reconhecimento. Um caso típico deste tipo de curva é apresentado na Figura 4.3. Nela pode-se observar as regiões que

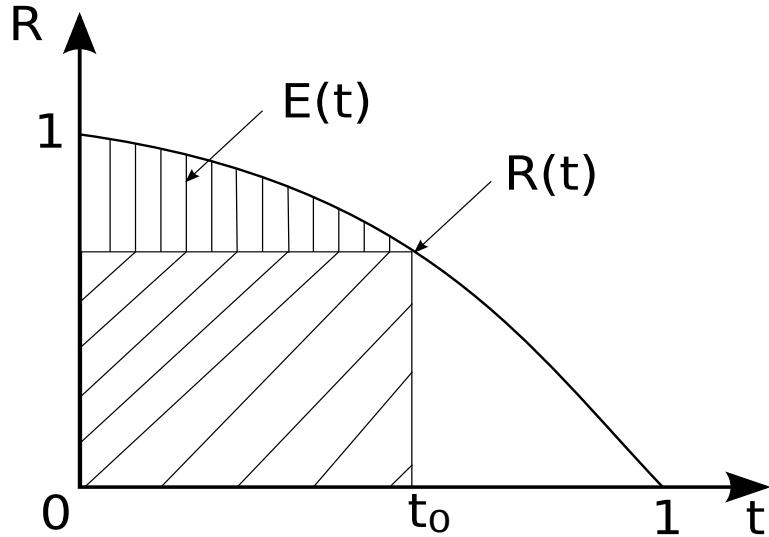


Figura 4.3: Exemplo ilustrativo de uma curva hipotética de compromisso entre o erro e a rejeição.

representam a taxa de erro quando nenhuma rejeição é permitida (a soma das duas áreas hachuradas), ou então a região que representa a taxa de erro quando são permitidas rejeições (região hachurada superior).

O limiar de rejeição pode então ser relacionado aos custos como segue (CHOW, 1957):

$$t = \frac{W_r - W_c}{W_e - W_c}, \quad (4.11)$$

em que W_r , W_e e W_c são os custos de rejeitar, de errar e de acertar, respectivamente. A regra de decisão ótima apresentada nas Equações (4.1) a (4.5) é também uma regra de risco mínimo se os custos associados aos erros, aos acertos e às rejeições são iguais. Usualmente, tem-se que $W_c < W_r < W_e$. Considerando $W_c = 0$ e $W_e = 1$, pode-se inferir que o custo de rejeição W_r é igual ao limiar de rejeição t ($W_r = t$). Assim, o risco mínimo para um determinado limiar é dado por

$$r(t) = E(t) + tR(t) = \int_0^{t_0} R(t)dt,$$

o que corresponde à área hachurada na Figura 4.3. Enquanto o risco empírico, para uma função de decisão e um determinado conjunto de treinamento, é definida como

$$r_{emp}(\alpha) = E(\alpha) + W_r R(\alpha), \quad (4.12)$$

em que α representa o vetor de parâmetros do modelo, dentre eles o limiar de rejeição.

Em um problema de classificação binária, a condição para rejeitar, descrita na Equação (4.5), pode não ser satisfeita quando $t > 0,5$, uma vez que a taxa de rejeição é sempre zero se o limiar de rejeição t exceder $1/2$. Esta é a razão do valor de W_r estar sempre no intervalo $[0;0,5]$.

Nas próximas seções são descritas as abordagens para projeto de classificadores com opção de rejeição engajados em tarefas de classificação binária. Grosso modo, tais técnicas baseiam-se em 2 possibilidades: (i) otimizar o limiar de rejeição após o treinamento usual do classificador e (ii) incorporar estratégias de rejeição ao aprendizado do classificador.

4.3 Abordagens para Classificação Binária com Opção de Rejeição

Em termos implementacionais existem três diferentes abordagens para classificação binária com opção de rejeição, a saber: (i) usando apenas um classificador; (ii) usando dois classificadores independentes; e (iii) usando um classificador com opção de rejeição incorporada na regra de aprendizado. As abordagens (ii) e (iii) enquadram-se na categoria das técnicas que incorporam o custo de rejeitar no aprendizado do(s) classificador(es). Estas abordagens são descritas em mais detalhes a seguir.

4.3.1 Um Classificador Padrão

Nesta abordagem, um padrão é rejeitado se o máximo das probabilidades a posteriori, ou seja, $\max\{p(\mathbf{x}|C_{-1}), p(\mathbf{x}|C_{+1})\}$, for menor que um determinado limiar. Se o classificador não fornecer saídas probabilísticas pode-se ainda definir o limiar de rejeição desde que uma função de saída seja monotônica. A região de rejeição é determinada após o treinamento padrão do classificador em um processo de otimização anterior à fase de teste, definindo valores limite adequados sobre a saída do classificador. Neste processo de otimização busca-se minimizar

$$\min E(\alpha) + W_r R(\alpha),$$

para se obter o limiar ótimo para um valor específico de W_r . No trabalho de Hanczar & Dougherty (2008) é apresentado um processo de otimização do limiar de rejeição. Esta abordagem leva a uma situação em que não há interseção entre as superfícies de decisão, conforme ilustrado na Figura 4.4 (SOUZA et al., 2009b).

Nesta tese, as estratégias um classificador MLP padrão e um classificador SVM padrão são denotadas MLP-1C e SVM-1C, respectivamente.

4.3.2 Dois Classificadores Independentes

Nessa abordagem requer-se dois classificadores, doravante denotados \mathbf{m}_1 e \mathbf{m}_2 . O primeiro classificador \mathbf{m}_1 é treinado com o intuito de penalizar os erros do tipo falso negativo. Por isso,

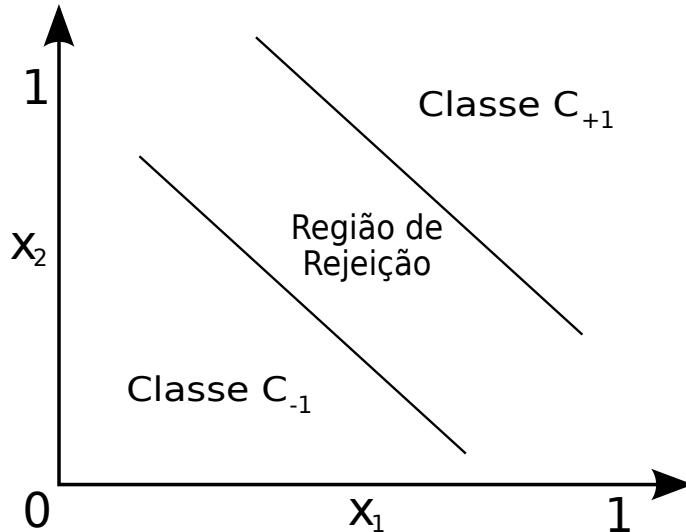


Figura 4.4: Regiões de decisão obtidas pela utilização de um único classificador para um problema binário.

espera-se que quando o classificador \mathbf{m}_1 classificar um padrão como pertencente à classe C_{-1} , este padrão deve pertencer verdadeiramente à classe C_{-1} . Por outro lado, o classificador \mathbf{m}_2 é treinado com o intuito de penalizar os erros do tipo falso positivo. Assim, quando o classificador \mathbf{m}_2 classificar um padrão como pertencente à classe C_{+1} , este padrão deve pertencer verdadeiramente à classe C_{+1} . Desta forma, o classificador \mathbf{m}_1 gera elevado valor de probabilidade para exemplos da classe C_{+1} , enquanto o classificador \mathbf{m}_2 produz elevado valor para exemplos da classe C_{-1} .

A regra de aceitação ou rejeição de um padrão para a estratégia baseada em dois classificadores é segundo Bounsiar et al. (2007) dada por

$$\text{Se } P_1(C_{+1}|\mathbf{x}) \geq 0,5 \text{ e } P_2(C_{-1}|\mathbf{x}) < 0,5 \text{ então Classe } C_{+1}. \quad (4.13a)$$

$$\text{Se } P_1(C_{+1}|\mathbf{x}) < 0,5 \text{ e } P_2(C_{-1}|\mathbf{x}) \geq 0,5 \text{ então Classe } C_{-1}. \quad (4.13b)$$

$$\text{Se } P_1(C_{+1}|\mathbf{x}) \geq 0,5 \text{ e } P_2(C_{-1}|\mathbf{x}) \geq 0,5 \text{ então rejeitar.} \quad (4.13c)$$

$$\text{Se } P_1(C_{+1}|\mathbf{x}) < 0,5 \text{ e } P_2(C_{-1}|\mathbf{x}) < 0,5 \text{ então rejeitar.} \quad (4.13d)$$

em que $P_1(C_{+1}|\mathbf{x})$ e $P_2(C_{+1}|\mathbf{x})$ são as probabilidades a posteriori dos classificadores \mathbf{m}_1 e \mathbf{m}_2 , respectivamente.

Esta abordagem pode fazer com que ocorra uma interseção entre as superfícies de decisão das classes, levando à ocorrência de regiões em que a confiança na tomada de decisão é bastante fraca (SOUZA et al., 2009b). Esta situação está ilustrada na Figura 4.5.

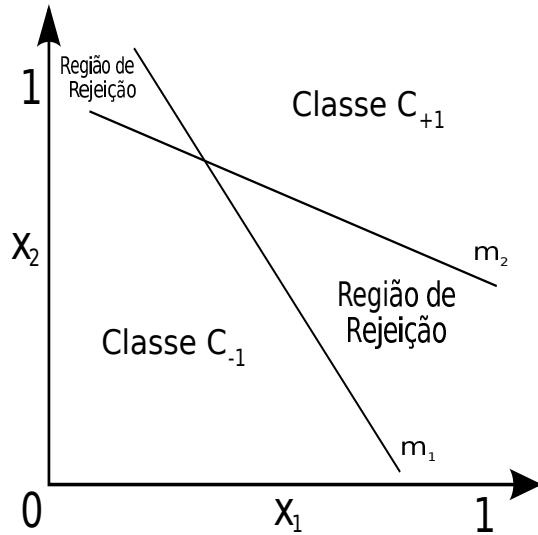


Figura 4.5: Regiões de decisão obtidas pela utilização de dois classificadores independentes para um problema binário.

Implementação Baseada em Duas Redes MLP

A abordagem baseada em duas redes MLP, doravante denominada MLP-2C, deve penalizar a tomada de decisão da classe C_{-1} no classificador \mathbf{m}_1 e deve penalizar a tomada de decisão da classe C_{+1} no classificador \mathbf{m}_2 . A penalização do classificador \mathbf{m}_1 pode ser realizada com base na alteração do cálculo do erro, ou seja

$$e_i = (y_i - d_i)W_r, \quad \text{se } d_i = +1 \quad (4.14a)$$

$$e_i = (y_i - d_i)(1 - W_r), \quad \text{se } d_i = -1 \quad (4.14b)$$

A penalização do classificador \mathbf{m}_2 é realizada de forma similar, também com base na alteração do cálculo da função de erro, a saber:

$$e_i = (y_i - d_i)(1 - W_r), \quad \text{se } d_i = +1 \quad (4.15a)$$

$$e_i = (y_i - d_i)W_r, \quad \text{se } d_i = -1 \quad (4.15b)$$

Desta maneira, cada uma dos classificadores, \mathbf{m}_1 e \mathbf{m}_2 , penaliza a aprendizagem para uma determinada classe, C_{-1} e C_{+1} , respectivamente. No mais, todo o processo de aprendizagem desses classificadores continua igual ao processo de aprendizagem usual da rede MLP. Vale destacar que nessa abordagem não há nenhum processo de otimização para busca de limiares, pois a alteração no cálculo do erro já considera os custos de rejeição para cada um dos classificadores e, por consequência, no classificador MLP-2C resultante.

Implementação Baseada em Dois Classificadores SVM

De forma semelhante, a abordagem baseada em dois classificadores SVM, doravante denominado SVM-2C, deve penalizar a tomada de decisão da classe C_{-1} no classificador \mathbf{m}_1 e deve penalizar a tomada de decisão da classe C_{+1} no classificador \mathbf{m}_2 . A penalização do classificador \mathbf{m}_1 pode ser realizada com base na alteração das restrições do problema dual, a saber:

$$\begin{aligned} 0 \leq \alpha_i &\leq CW_r, & \text{se } d_i = -1 \\ 0 \leq \alpha_i &\leq C(1 - W_r), & \text{se } d_i = +1 \end{aligned} \quad (4.16)$$

A penalização do classificador \mathbf{m}_2 é realizada de forma semelhante, também com base na alteração das restrições do problema dual, ou seja

$$\begin{aligned} 0 \leq \alpha_i &\leq C(1 - W_i), & \text{se } d_i = -1 \\ 0 \leq \alpha_i &\leq CW_i, & \text{se } d_i = +1 \end{aligned} \quad (4.17)$$

Com estas penalizações cada classificador adquire a capacidade de predizer com maior confiança uma determinada classe. As regras para aceitação e rejeição dos padrões são aquelas listadas na Seção 4.3.2.

4.3.3 Classificador com Opção de Rejeição Embutida

Esta abordagem consiste em projetar classificadores que trazem embutido no processo de otimização de suas funções-custo a capacidade de classificação com opção de rejeição. Assim, não é necessário uma etapa para obtenção do limiar de rejeição ótimo, também não é necessário especializar-se uma determinada classe como ocorre na abordagem com dois classificadores independentes. Na abordagem com opção de rejeição embutida, o modelo aprende intrinsecamente a decidir entre as três saídas possíveis, C_{-1} , rejeição e C_{+1} . Alguns trabalhos relacionados a esta abordagem são apresentados em Fumera & Roli (2002), Bounsiar et al. (2008) e Sousa et al. (2009b).

O primeiro trabalho a propor um classificador único com opção de rejeição embutido foi o desenvolvido por Fumera & Roli (2002). O classificador considerado clássico apresenta-se como uma variação do classificador SVM e é apresentado de forma sucinta a seguir.

O hiperplano de separação ótimo com opção de rejeição, descrito no trabalho de Fumera & Roli (2002), é um hiperplano que minimiza o seguinte funcional:

$$\min \tau(\mathbf{w}, \xi_i, \varepsilon) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^n h(\xi_i, \varepsilon) \quad (4.18)$$

sujeito a

$$\begin{aligned} d_i[(\mathbf{w}^T \mathbf{x}_i) + b] &\geq 1 - \xi_i, \quad i = 1, \dots, n \\ \xi_i &\geq 0, \quad i = 1, \dots, n \\ 0 &\leq \varepsilon \leq 1. \end{aligned}$$

O termo $h(\xi_i, \varepsilon)$ é ilustrado na Figura (4.6) e definido como

$$h(\xi_i, \varepsilon) = W_c S(\xi_i) + (W_r - W_c)S(\xi_i - 1 + \varepsilon) + (1 - W_r)S(\xi_i - 1 - \varepsilon) + a\xi_i^2, \quad (4.19)$$

em que W_c é um termo constante tal que $0 < W_c < W_r$ e $S(u)$ é a função sigmóide logística, ou seja

$$S(u) = \frac{1}{1 + e^{-au}} \quad (4.20)$$

para valores da constante $a(a > 0)$ suficientemente grandes.

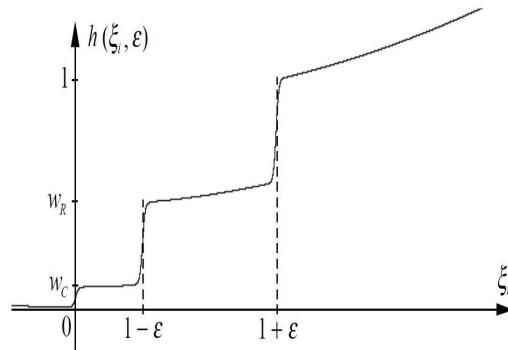


Figura 4.6: Exemplo ilustrativo da função $h(\xi_i, \varepsilon)$.

A solução do problema apresentado na Equação (4.18) pode ser obtida pela minimização da função lagrangeana em relação a \mathbf{w} , b , ξ_i e ε , sob determinadas restrições ($0 \leq \varepsilon \leq 1$); e pela maximização da função com relação aos multiplicadores de Lagrange não-negativos (BAZARAA et al., 1992). Note que a função lagrangeana é a soma de uma função convexa de \mathbf{w} e b , e de outra função não-convexa de ξ_i e ε . Neste sentido, o mínimo da função em relação a \mathbf{w} e b pode ser obtido com base nas seguintes condições

$$\frac{\partial \tau(\mathbf{w}, b, \xi, \varepsilon, \alpha)}{\partial \mathbf{w}} = \mathbf{0} \quad \Rightarrow \quad \mathbf{w} = \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i \quad (4.21a)$$

$$\frac{\partial \tau(\mathbf{w}, b, \xi, \varepsilon, \alpha)}{\partial b} = 0 \quad \Rightarrow \quad \sum_{i=1}^n \alpha_i y_i = 0 \quad (4.21b)$$

Ressalta-se que a Equação (4.21a) implica que o vetor de pesos \mathbf{w} apresenta a mesma expan-

são em relação aos vetores de treinamento como o apresentado no classificador SVM padrão. O mínimo da função lagrangeana em relação a ξ_i e ε , sobre determinadas restrições ($0 \leq \varepsilon \leq 1$), não pode ser obtido analiticamente. A forma dual, após a substituição das Equações (4.21a) e (4.21b) na Equação (4.18), é dada por

$$\max L(\alpha, \xi, \varepsilon) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j d_i d_j K(\mathbf{x}_i, \mathbf{x}_j) + C \min_{\xi_i: 0 \leq \varepsilon \leq 1} \sum_{i=1}^n \left(h(\xi_i, \varepsilon) - \frac{\alpha_i}{C} \xi_i \right) \quad (4.22)$$

sujeito a

$$\sum_{i=1}^n \alpha_i d_i = 0 \quad \text{e} \quad \alpha_i \geq 0 \quad \text{para } i = 1, \dots, n. \quad (4.23)$$

O problema dual é similar ao apresentado no capítulo anterior relacionado ao classificador SVM padrão. Formalmente, a diferença apresenta-se pelo termo adicional:

$$C \min_{\xi_i: 0 \leq \varepsilon \leq 1} \sum_{i=1}^l \left(h(\xi_i, \varepsilon) - \frac{\alpha_i}{C} \xi_i \right), \quad (4.24)$$

dentro da função objetivo, e pela ausência da restrição $\alpha_i \leq C$. O inconveniente do problema dual acima descrito surge devido à função-custo ser não convexa. Além disso, outra desvantagem é perda da garantia de unicidade e esparsidade da solução.

Embora a função-custo do problema seja côncava, o problema pode ser maximizado utilizando a mesma estratégia de solução iterativa do algoritmo SMO (PLATT, 1998). Pode-se ver que as restrições apresentadas na Equação (4.23), quando aplicadas a algum par de multiplicadores, fazem com que estes apresentem-se sobre uma segmento de reta. O valor máximo da função objetivo côncava em relação a um dado par de multiplicadores de Lagrange pode ser encontrado usando o método da seção áurea. Para avaliar a função-custo, o trabalho de Fumera & Roli (2002) descreve um algoritmo específico para resolver o problema de otimização proposto. Para selecionar um par de multiplicadores em cada iteração, e implementar um critério de parada, foram utilizadas heurísticas específicas, que exploram as características do problema. Mais detalhes sobre estes algoritmos encontram-se em (FUMERA, 2002).

Outra proposta no que se refere a um classificador único com opção de rejeição embutida é desenvolvida por Sousa et al. (2009b), e denominada Rejo SVM. A formulação do classificador RejoSVM baseia-se na replicação dos dados para um espaço expandido, na detecção de regiões de rejeição com base no espaço expandido e no uso de classificadores SVM.

O método de replicação de dados assume que os padrões, no problema de classificação,

originam-se de k classes ordenadas, tal que devem ser rotuladas de C_1, \dots, C_K , correspondendo à sua ordem natural. Então, considere o conjunto de treinamento $\{\mathbf{x}_i^{(k)}\}_{i=1}^l$, em que $k = 1, \dots, K$ representa o número de classes. Considere também que $\mathbf{x}_i^{(k)} \in \mathbb{R}^d$, com d descrevendo a dimensão do padrão no espaço de entrada. Exemplos de problemas artificiais que obedecem à condição de ordem são apresentados nas Figuras 4.7(a) e 4.7(b). O processo de replicação

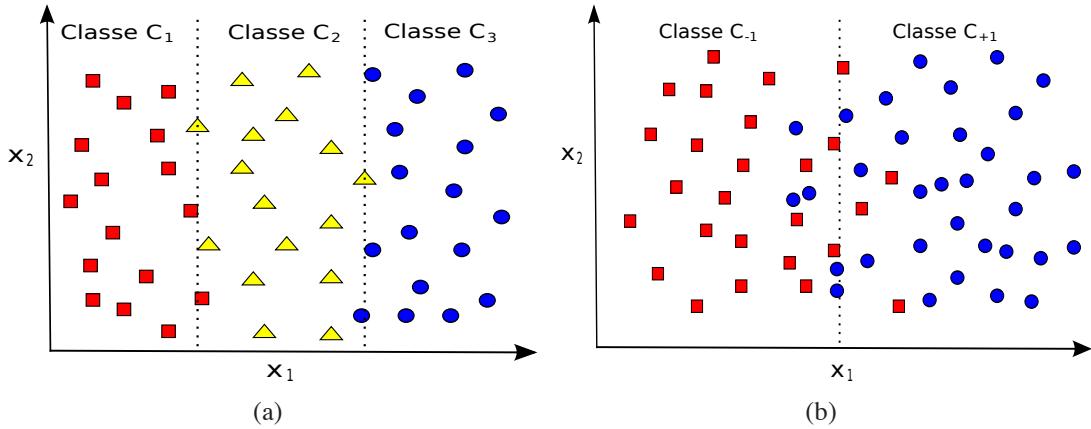


Figura 4.7: (a) Problema artificial com 3 classes em que são apresentados os padrões no espaço \mathbb{R}^2 obedecendo o conceito de ordem. (b) Problema binário artificial.

transforma o problema binário original para um espaço expandido mostrado na Figura 4.8, em que há padrões replicados, de acordo com a seguinte regra:

$$\mathbf{x} \in \mathbb{R}^d \begin{cases} \begin{bmatrix} \mathbf{x} \\ h \end{bmatrix} \in \mathbb{R}^3 \\ \begin{bmatrix} \mathbf{x} \\ 0 \end{bmatrix} \in \mathbb{R}^3 \end{cases} \quad (4.25)$$

em que $h \in \mathbb{R}^+$ é uma constante positiva. Note que qualquer padrão replicado difere apenas do seu padrão base em termos da dimensão adicionada. Então, um classificador binário pode ser aplicado sobre o conjunto de dados expandido, a fim de obter uma superfície (plano) de separação entre as duas classes tal como ilustrado na Figura 4.9. As interseções da superfície de decisão com o hiperplano que contém os pontos $\mathbf{x}^T = [\mathbf{x}^T 0]$ e com o hiperplano que contém os pontos $\mathbf{x}^T = [\mathbf{x}^T h]$ pode ser usada para obtenção de superfícies de decisão no espaço de entrada original (veja a Figura 4.10).

Logo, com uma escolha apropriada para os custos de classificação, o método de replicação pode ser usado para aprender uma região de rejeição, definida por duas superfícies de decisão que não se interceptam. Nesse sentido, pode-se notar que a região de rejeição é otimizada durante o processo de treinamento e não posteriormente de forma heurística. Fronteiras de

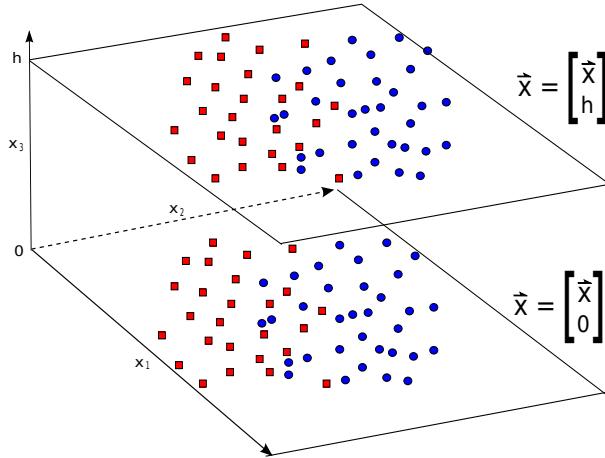


Figura 4.8: Processo de replicação de dados de um problema binário artificial em que os padrões são expandidos para o espaço \mathbb{R}^3 .

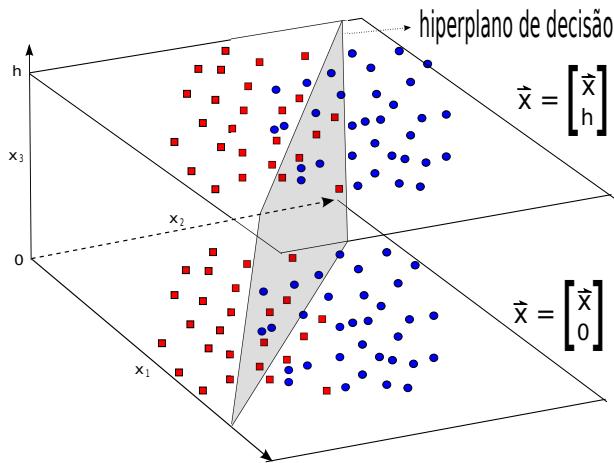


Figura 4.9: Hiperplano de separação entre as classes.

decisão não-lineares (que não se interceptam) são tratadas exatamente como no cenário de dados ordinais. Logo, a classificação segue o mesmo raciocínio.

Comumente atribui-se os mesmos custos para erro e rejeição em um problema binário. Desta maneira, a matriz de custos é expressa com na Tabela 4.3.3.

		predição		
		C_{-1}	$C_{rejeicao}$	C_{+1}
rótulo	C_{-1}	0	C_{baixo}	C_{alto}
	C_{+1}	C_{alto}	C_{baixo}	0

Tabela 4.1: Exemplo da matriz de custos.

Portanto, $C_{rejeicao} = \frac{C_{baixo}}{C_{alto}} = W_r$ define o custo de rejeitar normalizado pelo custo de errar. A fim de classificar um padrão não utilizado no processo de aprendizagem, deve-se classificar ambas as réplicas do padrão no espaço expandido. Assim, da seqüência de rótulos binários

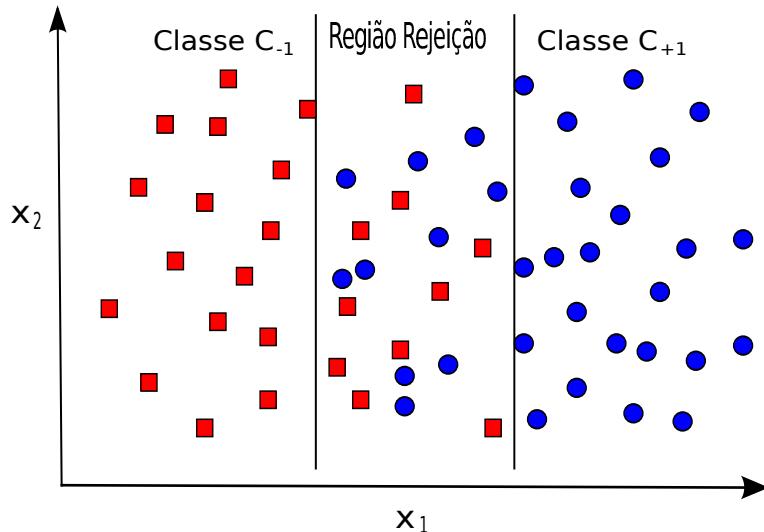


Figura 4.10: Hiperplanos de separação entre as classes no espaço original.

obtidos pode-se inferir o rótulo predito, tal que:

$$C_{-1}C_{-1} \Rightarrow C_{-1} \quad (4.26a)$$

$$C_{-1}C_{+1} \Rightarrow C_{rejeição} \quad (4.26b)$$

$$C_{+1}C_{+1} \Rightarrow C_{+1}. \quad (4.26c)$$

A Equação (4.26a) descreve a situação na qual se classifica o padrão \mathbf{x} como da Classe C_{-1} quando o padrão $\mathbf{x}_{(0)}^T = [\mathbf{x}^T 0]$ é classificado como pertencente à Classe C_{-1} e o padrão $\mathbf{x}_{(h)}^T = [\mathbf{x}^T h]$ é classificado como pertencente à Classe C_{-1} . De forma similar, a Equação (4.26c) descreve a classificação de um padrão \mathbf{x} para a Classe C_{+1} quando o padrão $\mathbf{x}_{(0)}^T$ é classificado como pertencente à Classe C_{+1} e o padrão $\mathbf{x}_{(h)}^T$ é classificado como pertencente à Classe C_{+1} . Por fim, a Equação (4.26b) representa a situação em que se rejeita o padrão \mathbf{x} , sempre que $\mathbf{x}_{(0)}^T$ é classificado como pertencente à Classe C_{-1} e o padrão $\mathbf{x}_{(h)}^T$ é classificado como pertencente à Classe C_{+1} .

4.4 Novas Propostas para Classificação com Opção de Rejeição

Nesta seção são apresentadas as estratégias propostas para classificação com opção de rejeição baseadas na rede SOM, a saber: Classificador Som Padrão e Dois Classificadores SOM Independentes. A primeira proposta é baseada na estratégia de rejeição com base em um classificador padrão, enquanto a segunda é baseada na estratégia de rejeição com base em dois classificadores independentes.

4.4.1 Classificador SOM Padrão (Proposta 3)

A Proposta 3 desta tese baseia-se em um classificador implementado por uma rede SOM, denominada SOM-1C. Detalhes sobre o uso da rede SOM para classificação supervisionada podem ser obtidos em (ROCHA-NETO, 2006). Após o processo usual de treinamento da rede SOM, 3 etapas fazem-se necessárias. Na primeira etapa deve ser determinado o neurônio vencedor para cada um dos padrões do conjunto de treinamento. Ao final desta, os neurônios que não possuírem padrões associados devem ser podados da rede.

Na segunda etapa, deve ser calculada uma medida de impuridade (ou heterogeneidade) da distribuição das classes dos padrões mapeados em um dado neurônio da rede SOM. A medida escolhida neste trabalho é o índice de Gini que representa a desigualdade de uma distribuição (XU, 2004). O índice de Gini adaptado à presente proposta é expresso como

$$\text{índice de Gini} = IG_i = 1 - \sum_{j=1}^c [P(C_j|w_i)]^2, \quad (4.27)$$

em que c representa o número de classes e $P(C_j|w_i)$ representa a probabilidade da j -ésima classe para o i -ésimo neurônio. A probabilidade $P(C_j|w_i)$ é estimada como

$$P(C_j|w_i) \approx \frac{n_{ij}}{n_i}, \quad (4.28)$$

em que n_{ij} é o número de exemplos da j -ésima classe mapeados no i -ésimo neurônio e n_i é o número total de exemplos mapeados no i -ésimo neurônio. Suponha, por exemplo, que 50 padrões foram mapeados no neurônio i^* , sendo 10 da classe C_{+1} e 40 da classe C_{-1} . Neste caso, $P(C_{+1}|w_{i^*}) = \frac{10}{50} = 0,2$, $P(C_{-1}|w_{i^*}) = \frac{40}{50} = 0,8$ e $IG_{i^*} = 1 - [0,2^2 + 0,8^2] = 0,32$.

Para o índice de Gini, um valor 0 indica que todos os rótulos dos padrões atraídos são da mesma classe, enquanto um valor 1 indica que todos os rótulos são diferentes. O limiar de impuridade (de rejeição) também é definido em um processo de otimização que busca minimizar a Equação (4.12).

Na terceira e última etapa, cada um dos neurônios deve ser rotulado de acordo com a classe mais representativa, se houver um valor de impuridade menor que um determinado limiar. Em caso contrário, o neurônio deve ser rotulado como da classe de “rejeição”. Para fins de classificação, um padrão não-visto deve ser apresentado à rede para obtenção do seu neurônio vencedor, em seguida deve ser atribuído ao padrão o rótulo do protótipo vencedor.

4.4.2 Dois Classificadores SOM Independentes (Proposta 4)

Uma outra proposta para projeto de classificadores com opção de rejeição consiste em treinar duas redes SOM. A primeira rede (\mathbf{m}_1) treinada preferencialmente com padrões da classe C_{-1} , enquanto a segunda (\mathbf{m}_2) é treinada preferencialmente com padrões da classe C_{+1} . Para controlar a penalização de uma determinada classe no processo de treinamento é necessária uma nova regra de atualização dos protótipos da rede SOM. Esta estratégia, doravante denotada SOM-2C, consiste na especialização durante o treinamento das duas redes SOM envolvidas, de tal maneira que uma delas se especializa nos padrões de treinamento da classe C_{-1} , enquanto a outra se especializa nos padrões de treinamento da classe C_{+1} . Este processo de aprendizado especializado aplicado aos classificadores \mathbf{m}_1 e \mathbf{m}_2 considera o uso de todo o conjunto de treinamento. A especialização em uma classe significa que uma rede aprende mais (e não apenas) os padrões dessa classe do que os padrões da outra classe.

A proposta SOM-2C difere do que tradicionalmente se encontra na literatura, uma vez que tais abordagens baseiam-se ou na minimização do erro de classificação (como em classificadores MLP) ou na minimização do risco estrutural (como em classificadores SVM). Além disso, ambas as abordagens fundamentam-se em processos de aprendizagem supervisionado. Na proposta SOM-2C utiliza-se um algoritmo de quantização vetorial, tal que o treinamento ocorre de forma não supervisionada, sem a minimização explícita de uma função-custo.

No processo de aprendizagem do classificador \mathbf{m}_1 , os padrões da classe C_{-1} são “preferidos” em relação aos padrões pertencentes à outra classe. Esta preferência é expressa em termos de um peso W_r que equivale a definir um custo C_{alto} e C_{baixo} para os exemplos das classes C_{-1} e C_{+1} , respectivamente. Tais custos relacionam-se ao custo de rejeição por meio da seguinte expressão (Sousa et al., 2009a).

$$W_r = C_{baixo}/C_{alto} \quad (4.29)$$

Logo, para este fim, é proposta variante da regra de atualização dos protótipos de uma rede SOM, mais precisamente do classificador \mathbf{m}_1 , dada por

$$\begin{aligned} \mathbf{w}_j(t+1) &= \mathbf{w}_j(t) + \eta(k)h(i^*, j; t)[\mathbf{x}(t) - \mathbf{w}_j(t)]W_r, & \text{se } d_i = +1 \\ \mathbf{w}_j(t+1) &= \mathbf{w}_j(t) + \eta(k)h(i^*, j; t)[\mathbf{x}(t) - \mathbf{w}_j(t)](1 - W_r), & \text{se } d_i = -1 \end{aligned} \quad (4.30)$$

em que \mathbf{w}_j é o protótipo do neurônio j , i^* é o índice do neurônio vencedor, t é a iteração atual do processo de treinamento, η é a taxa de aprendizagem e h é uma função de vizinhança. A regra de atualização dos protótipos do modelo \mathbf{m}_2 que dá “preferência” ao padrões da classe

C_{+1} é expressa da seguinte maneira:

$$\begin{aligned}\mathbf{w}_j(t+1) &= \mathbf{w}_j(t) + \eta(k)h(i^*, j; t)[\mathbf{x}(t) - \mathbf{w}_j(t)](1 - W_r), & \text{se } d_i = +1. \\ \mathbf{w}_j(t+1) &= \mathbf{w}_j(t) + \eta(k)h(i^*, j; t)[\mathbf{x}(t) - \mathbf{w}_j(t)]W_r, & \text{se } d_i = -1\end{aligned}\quad (4.31)$$

Já W_r representa o peso associado à força de atração (forte ou fraca), que deve ter efeito sobre o neurônio i a fim de o mover mais rapidamente, ou mais lentamente na direção dos exemplos pertencentes a uma determinada classe.

Após a etapa de treinamento, deve-se associar cada um dos padrões do conjunto de treinamento ao seu neurônio vencedor. Similarmente ao que ocorre no classificador SOM-1C, os neurônios que não possuem padrões associados devem ser podados da rede. Cada neurônio deve ser rotulado com o rótulo da classe mais freqüente dentre aquelas mapeadas neste neurônio, se o valor de impuridade associado for menor que um determinado limiar. Caso contrário, o neurônio deve ser rotulado como classe de “rejeição”. Para fins de classificação, um padrão não-visto deve ser apresentado à rede para obtenção do seu neurônio vencedor, em seguida deve ser atribuído ao padrão o rótulo associado ao neurônio vencedor.

4.5 Simulações Computacionais

O objetivo deste estudo experimental é avaliar as estratégias de classificação com opção de rejeição propostas nesta tese quando aplicadas ao problema PCV-2C. A apresentação dos resultados é feita em duas partes. A primeira parte descreve os resultados das abordagens que utilizam classificadores que buscam a minimização do risco estrutural (classificadores SVM). Enquanto a segunda parte, descreve os resultados obtidos para as abordagens que usam os classificadores baseados nas redes MLP e SOM.

Os códigos das estratégias para rejeição descritas neste capítulo foram desenvolvidos em MatlabTM, com base em uma API (*toolbox*) disponível online², no tocante à rede SOM, e a *toolbox* de redes neurais do próprio MatlabTM.

4.5.1 Resultados para classificadores SVM

O conjunto de treinamento é composto, em diferentes experimentos, por 5%, 25% 40%, 60% e 80% do total de padrões. A separação do conjunto de treinamento e teste é repetida por 100 vezes a fim de obter estatísticas mais confiáveis. A melhor parametrização de cada modelo é encontrada por uma busca em grade, com base em uma estratégia de validação cruzada de

²<http://www.cis.hut.fi/somtoolbox>

5-partes (*5-fold cross validation*) conduzida sobre o conjunto de treinamento. Nestes experimentos os modelos utilizam um *kernel* linear. A busca em grade sobre o parâmetro C da SVM abrange valores entre 2^{-5} e 2^5 . Ao final, a acurácia e a rejeição do modelo são estimadas sobre o conjunto de teste.

O desempenho de um classificador com capacidade de rejeição pode ser descrito através de uma curva que leva em consideração a taxa de classificação (acurácia) em relação à sua taxa de rejeição. Esta representação é denominada curva A-R (*Accuracy-Reject Curve*), em que cada valor correspondente a uma taxa de erro e a uma taxa de rejeição que depende do custo de rejeição W_r . Isto implica que diferentes pontos da curva A-R correspondem a diferentes valores de W_r ($W_r = 0,04; 0,08; 0,12; \dots; 0,48$). Somente são considerados valores de W_r menores ou iguais a 0,5, devido às razões mostradas apresentadas na Subseção 4.2.1.

Na Figuras 4.11 a 4.13 são apresentados os resultados obtidos para todas as estratégias baseadas em classificadores SVM.

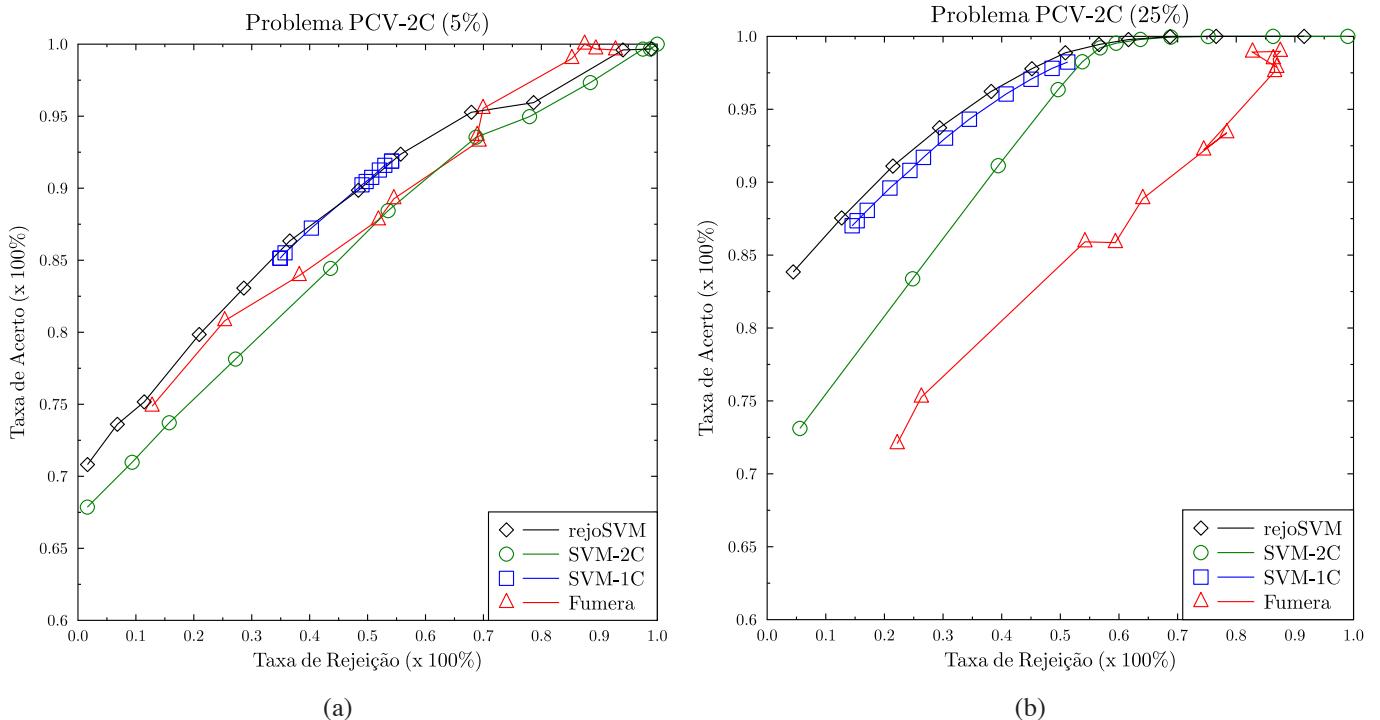


Figura 4.11: (a) Curva A-R quando se utiliza 5% dos dados para treinamento (b) Curva A-R quando se utiliza 25% dos dados de treinamento.

Pode ser observado nas Figuras 4.11-4.13 que para valores elevados da taxa de rejeição, também se verifica valores elevados da acurácia. Pode ser verificado também que as taxas de classificação chegam a aproximadamente 100% quando a taxa de rejeição é de 50%, tal como mostrado nas Figuras 4.12(a), 4.12(b) e 4.13(a). Outras taxas de classificação bastante elevadas são obtidas, por exemplo, quando a taxa de rejeição é de 30% com acurácia de cerca de

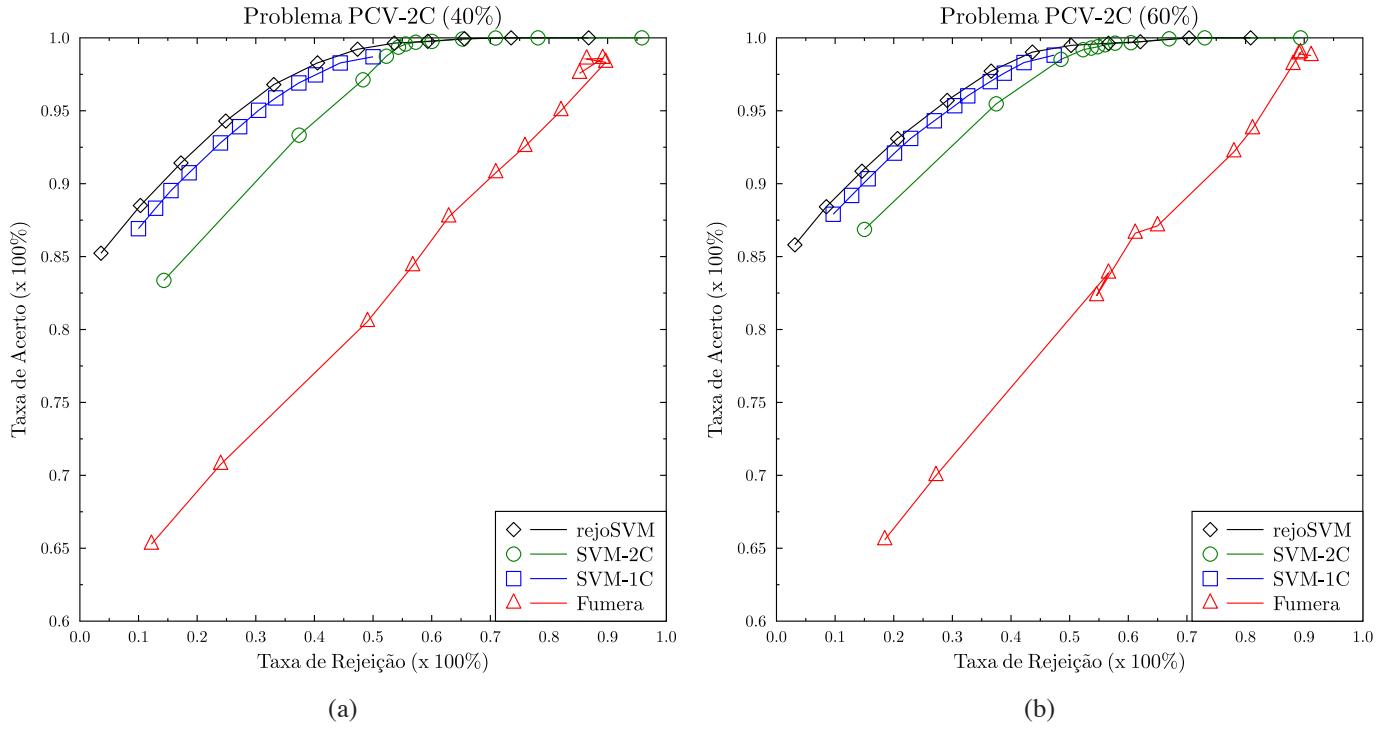


Figura 4.12: (a) Curva A-R quando se utiliza 40% dos dados para treinamento. (b) Curva A-R quando se utiliza 60% dos dados para treinamento.

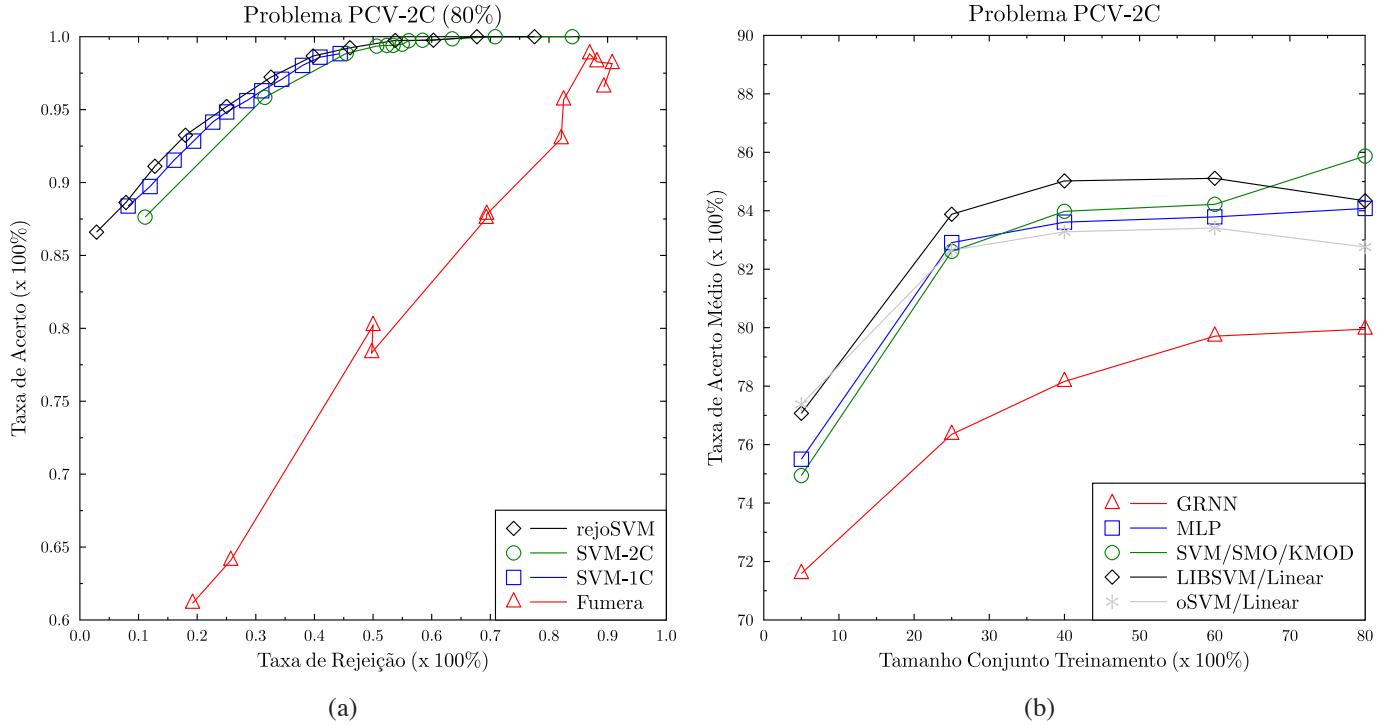


Figura 4.13: (a)Curva A-R quando se utiliza 80% dos dados para treinamento. (b) Resultados para diversas técnicas de aprendizado de máquina.

95%, conforme mostrado na Figura 4.13(a). Os métodos avaliados em geral obtêm resultados similares, com exceção da técnica proposta por (FUMERA; ROLI, 2002). Este desempenho pode ser justificado pelo fato de que a função de otimização não é convexa e, assim, pode não

alcançar o mínimo global. Apesar da similaridade entre os resultados das estratégias SVM-1C, SVM-2C e rejoSVM, esta última apresenta resultados um pouco melhores comparativamente às demais, pois em geral a sua curva está posicionada sobre as curvas das estratégias SVM-1C, SVM-2C e Fumera.

Com o intuito de enfatizar o uso da opção de rejeição no aumento da acurácia de classificadores de um modo geral, na Tabela 4.3 são apresentados experimentos com 5 tipos de classificadores diferentes que não usam opção de rejeição, a saber: SVM/SMO/KMOD, GRNN, MLP, oSVM e libSVM/Linear (CHANG; LIN, 2001). Mais detalhes sobre os classificadores MLP e GRNN podem ser obtidos em (ROCHA-NETO, 2006).

Treinamento	Método	W _r		
		0,04	0,24	0,48
40%	rejoSVM	96,5	87,9	83,5
	SVM-1C	96,7	87,7	82,1
	SVM-2C	96,2	86,0	76,5
80%	rejoSVM	96,9	89,1	85,2
	SVM-1C	97,1	88,8	84,4
	SVM-2C	96,6	86,3	82,3

Tabela 4.2: Resultados para os classificadores rejoSVM, SVM-1C, SVM-2C quando treinados com 40% e 80% do total de dados do problema PCV-2C.

Treinamento	Método	Acurácia
40%	libSVM/Linear	85,0
	SVM/SMO/KMOD	83,9
	MLP	83,6
	oSVM	83,3
	GRNN	78,2
80%	libSVM/Linear	84,3
	SVM/SMO/KMOD	85,9
	oSVM	82,8
	MLP	84,1
	GRNN	80,0

Tabela 4.3: Resultados para os classificadores SVM/SMO/KMOD, libSVM/Linear, oSVM, MLP e GRNN quando treinados com 40% e 80% do total de dados do problema PCV-2C.

Com relação às Tabelas 4.2 e 4.3, os resultados apresentados mostram que para $W_r = 0,48$ e com classificadores treinados com 80% dos dados as estratégias de rejeição e os classificadores tradicionais apresentam resultados bastante similares. Porém, para valores de W_r inferiores as taxas de classificação das estratégias de rejeição possuem valores bem mais elevados.

4.5.2 Resultados das Estratégias que se baseiam em redes MLP e SOM

Os classificadores MLP-1C, MLP-2C, SOM-1C e SOM-2C são aqui avaliados em dois conjunto de dados. O primeiro conjunto foi gerado artificialmente para fins de validação das estratégias de classificação com opção de rejeição, enquanto o segundo conjunto de dados é o problema PCV-2C. O conjunto artificial, denominado Sintético I, contém 400 pontos $\mathbf{x} \in \mathbb{R}^2$ que pertencem ao quadrado unitário, $[0, 1] \times [0, 1] \subset \mathbb{R}^2$, e seguem uma distribuição uniforme. Este conjunto apresenta dois platôs distribuídos uniformemente e uma zona de transição de probabilidade linearmente decrescente, delimitada por duas fronteiras hiperbólicas (CARDOSO; COSTA, 2007).

Na Figura 4.14 são apresentados os resultados experimentais para o conjunto Sintético I. Em cada gráfico desta figura, além das curvas A-R para os classificadores SOM-1C e SOM-2C, também são mostradas as curvas A-R para os classificadores MLP-1C e MLP-2C. Cada ponto em destaque na curva corresponde a um determinado valor de W_r . Os valores considerados para W_r foram 0,4; 0,24 e 0,44. Foram realizadas 50 rodadas do processo de treinamento/teste. Para os classificadores SOM-1C e SOM-2C, foram realizados processos de validação cruzada com 5-partes a fim de se obter número de neurônios da rede SOM dentre os mapas 5×5 , 5×7 , 7×7 e 10×10 . No caso do classificador SOM-2C, foi agregado o rótulo com codificação *1-out-of-K* ao vetor de características para fins de treinamento com $K = 2$. Esta agregação foi necessária para que melhores resultados fossem obtidos. O critério de parada do treinamento da rede SOM é a quantidade de épocas (200) ou então a convergência do erro de quantização. A taxa de aprendizagem e a função de vizinhança possuem um decaimento exponencial em função das iterações do algoritmo de treinamento. O raio de vizinhança (taxa de aprendizado) inicial e final de 5 (0.5) e 0.1 (0.05), respectivamente.

Uma pesquisa em grade, para os classificadores MLP-1C e MLP-2C, foi realizada sobre o número de neurônios ocultos que devem compor a rede na resolução do problema artificial. Os valores analisados foram de 5, 10, 15 e 20 neurônios na única camada oculta, enquanto a camada de saída é sempre composta de um único neurônio. O critério de parada do processo de treinamento é o número de épocas (500). O algoritmo de retropropagação padrão foi utilizado com taxa de aprendizagem igual a 0,05.

Os classificadores SOM-1C e SOM-2C apresentam melhores resultados quando comparadas com os classificadores MLP-1C e MLP-2C (veja Figura 4.14). Isto pode ser verificado na Figura 4.14, pois para uma mesma taxa de rejeição na de 5% a 35%, as curvas A-R dos classificadores SOM-1C e SOM-2C estão acima das curvas para os classificadores MLP-1C e MLP-2C.

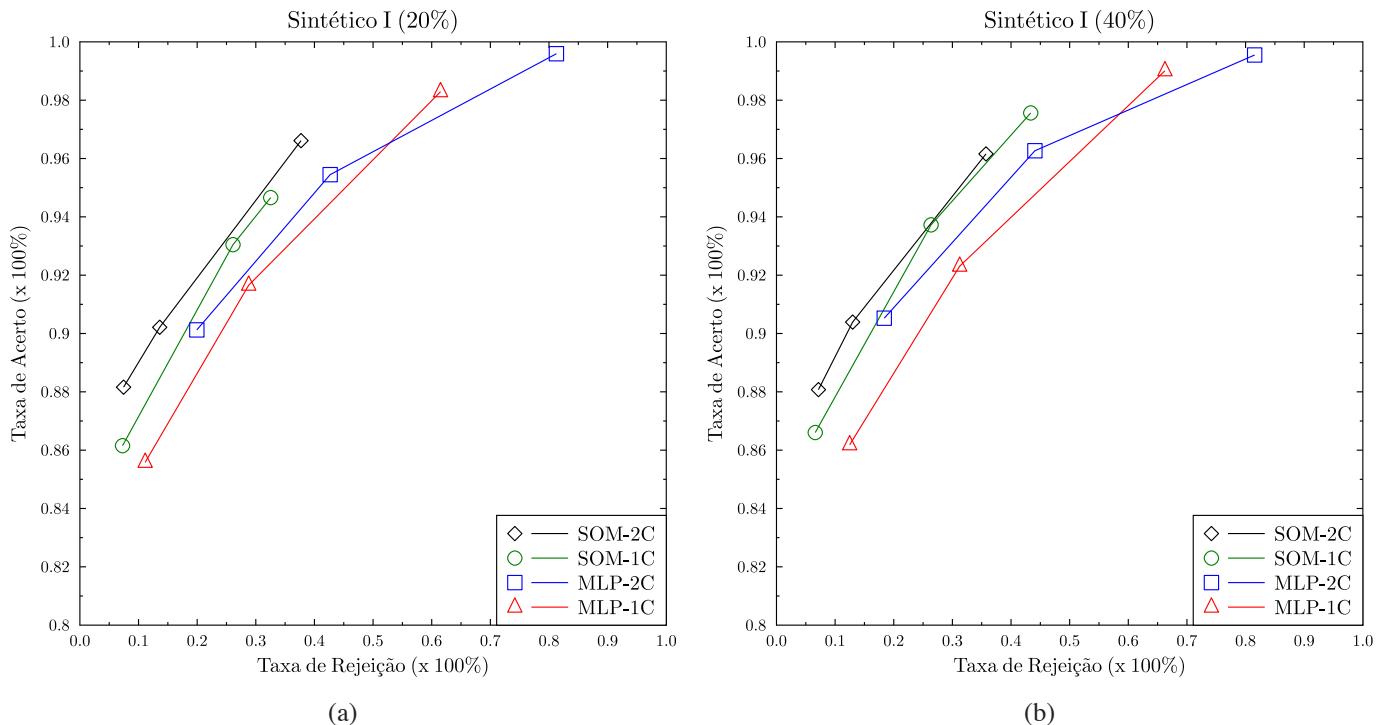
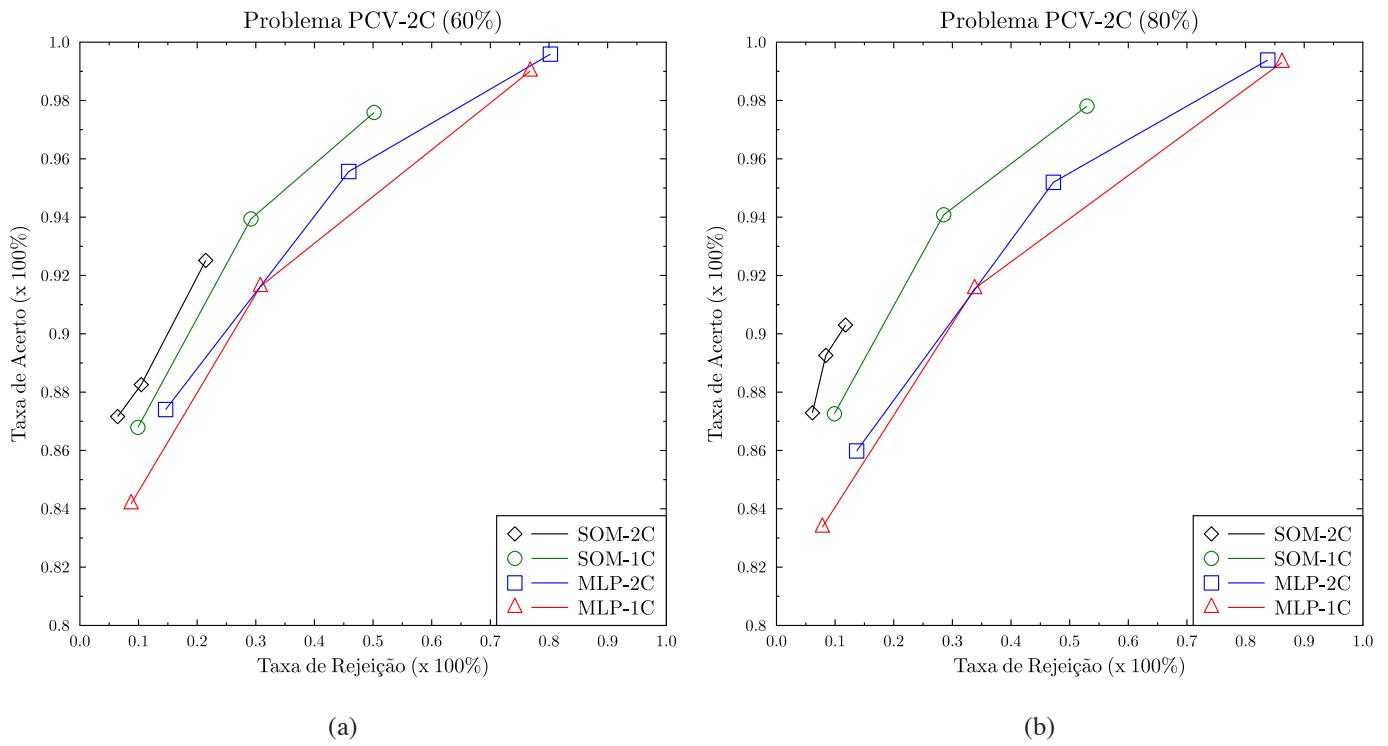
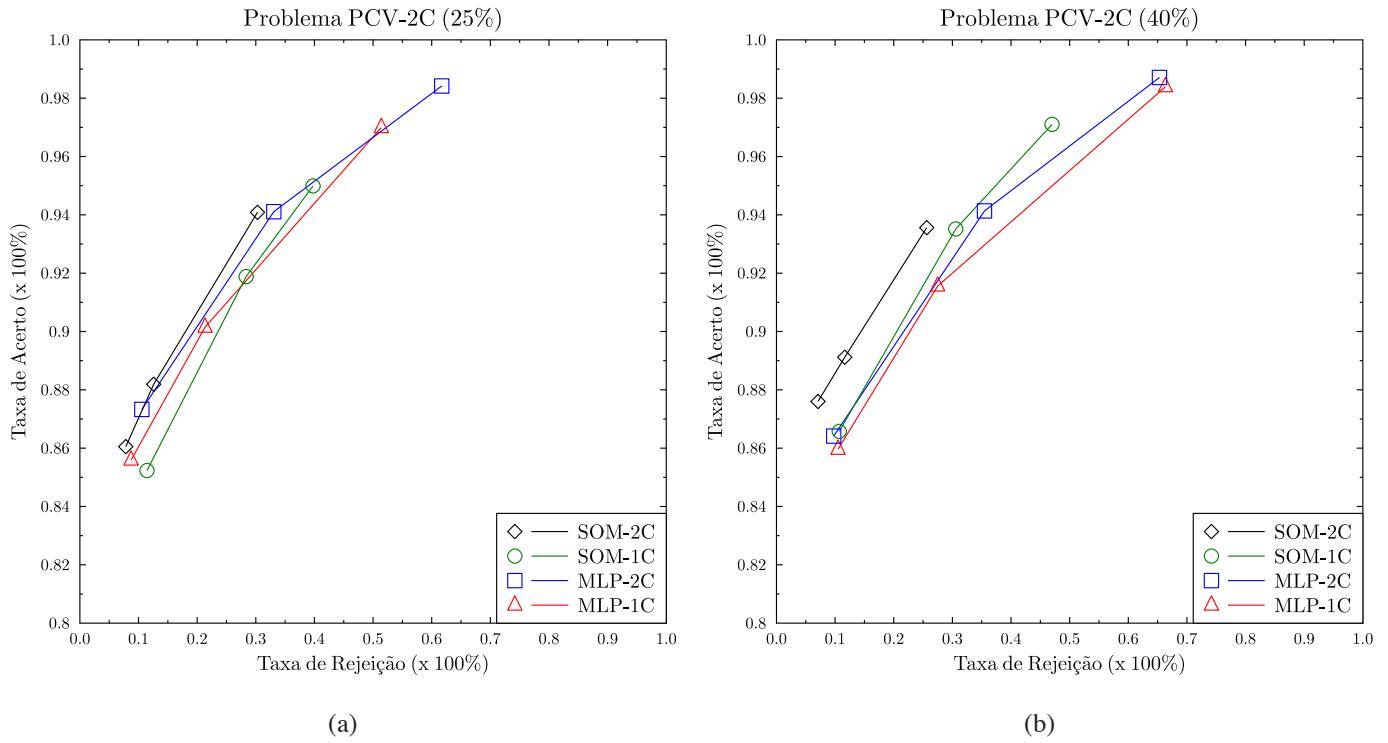


Figura 4.14: (a) Curva A-R obtida para 20% do total de dados no conjunto de treinamento. (b) Curva A-R obtida para 40% do total de dados no conjunto de treinamento. Estes resultados foram obtidos para o conjunto Sintético I.

Entre os classificadores SOM-1C e SOM-2C, pode-se notar facilmente que o classificador SOM-2C possui desempenho superior. Para 15% de rejeição em ambos os gráficos da Figuras 4.14(a) e 4.14(b), são obtidas taxas de classificação pelo classificador SOM-2C superiores a 90%.

Os resultados experimentais para o problema PCV-2C são apresentados nas Figuras 4.15 e 4.16. De forma similar, em cada gráfico desta figura estão as curvas A-R dos classificadores SOM-1C, SOM-2C, MLP-1C e MLP-2C. Cada ponto da curva correspondente a um determinado valor de W_r (0,4; 0,24 e 0,44). Nestas simulações foram executadas 100 realizações do processo de treinamento/teste. Para os classificadores SOM-1C e SOM-2C, foram realizados processos de validação cruzada de 5-partes a fim de se escolher o melhor número de neurônios da rede SOM dentre os mapas 10×10 , 12×15 , 10×20 e 15×15 . No caso do classificador SOM-2C, também houve a agregação do rótulo com codificação *1-out-of-K* ($K = 2$) ao vetor de entrada para fins de treinamento. O critério de parada do treinamento da rede SOM nestes experimentos é a quantidade de épocas (200) ou então a convergência do erro de quantização, o que acontece primeiro. A taxa de aprendizagem e a função de vizinhança são determinadas como descrito anteriormente.



Para os classificadores MLP-1C e MLP-2C, no tocante ao problema PCV-2C, foi realizada

uma pesquisa em grade (*grid search*) com o intuito de se determinar o número de neurônios ocultos correspondentes. As quantidades avaliadas foram de 5, 10, 15 e 20 neurônios na única camada oculta. A camada de saída possui apenas um neurônio. O critério de parada do processo de treinamento é o número de épocas (500). Utilizou-se o algoritmo de retropropagação padrão com taxa de aprendizagem igual a 0,05.

Os resultados obtidos para o problema PCV-2C e apresentados nas Figuras 4.15 e 4.16 também confirmam o melhor desempenho dos classificadores SOM-1C e SOM-2C, pois nota-se que as curvas A-R destas estratégias encontram-se quase sempre sobre as dos classificadores MLP-1C e MLP-2C. Uma exceção ocorre quando o treinamento se dá com apenas 25% dos dados. Este fato pode ser justificado pela necessidade de uma quantidade relativamente maior de padrões para o treinamento das redes SOM, principalmente em problemas mais complexos como o de diagnóstico de PCV-2C. Entre as estratégias que se baseiam na rede SOM, o classificador SOM-2C apresenta melhores taxas de classificação que o classificador SOM-1C. Na Figura 4.16(b) verifica-se que para um pouco mais de 10% de taxa de rejeição a acurácia supera os 90%.

4.5.3 Análise Comparativa das Estratégias de Rejeição

As Figuras 4.17(a) e 4.17(b) apresentam os resumos dos resultados obtidos para as diversas estratégias de classificação com opção de rejeição avaliadas nesta tese. A Figura 4.17(a) apresentada os resultados para os classificadores treinados com 60% do total de dados, enquanto os resultados obtidos para os classificadores treinados com 80% do total de dados são mostrados na Figura 4.17(b). Ambas as figuras apresentam os resultados para W_r com valores 0,04, 0,24 e 0,44.

A Figura 4.17(a) mostra que o classificador SOM-2C apresenta desempenho igual ou superior aos demais classificadores na faixa de 5% a 20%. O classificador SOM-1C também apresenta melhor desempenho que os classificadores MLP-1C, MLP-2C e SVM-2C na faixa de 10% a 50%. Similarmente, a partir da Figura 4.17(a), nota-se que o classificador SOM-2C possui melhor desempenho quando comparado com as demais estratégias, com exceção do classificador rejoSVM, para rejeição próxima de 10%. O Classificador SOM-1C também possui melhor desempenho que os classificadores MLP-1C, MLP-2C e SVM-2C na faixa de 10% a 40%. Nota-se ainda que o classificador rejoSVM apresenta desempenho melhor que os classificadores MLP-1C, MLP-2C, SVM-1C, SVM-2C e SOM-1C.

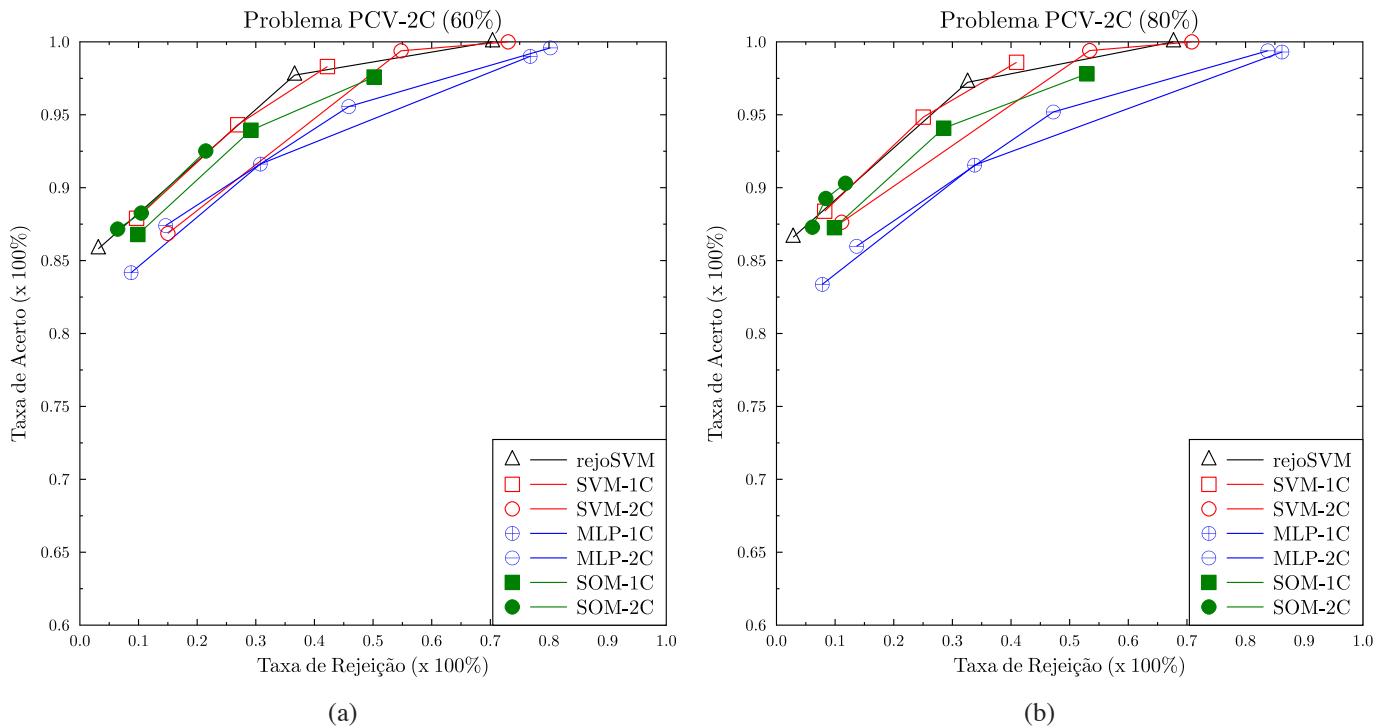


Figura 4.17: (a) Curvas A-R para estratégias de rejeição treinadas com 60% do total de dados. (b) Curvas A-R para estratégias de rejeição treinadas com 80% do total de dados. O problema avaliado é o PCV-2C.

4.6 Conclusão

A formulação do problema incluindo a classificação com opção de rejeição é bastante atrativa, pois permite proteger um sistema contra um elevado número de decisões erradas em situações em que se exige alta confiabilidade. No contexto do SINPATCO II, a incorporação da classificação com opção de rejeição pode ser bastante benéfica, pois ferramentas como esta são projetadas para auxiliar a tomada de decisão local, ou mesmo, em áreas remotas com acesso limitado a recursos modernos e financiamentos. Inclusive, nestas áreas, muitas vezes há carência de médicos ortopedistas. Neste sentido, o sistema deve impor taxas elevadas de classificação e uma maior precisão e confiança sobre o diagnóstico realizado.

Neste capítulo foram apresentadas estratégias contidas na literatura, bem como foram propostas novas estratégias. Tanto as estratégias da literatura quanto as propostas conseguem obter elevadas taxas de classificação para o problema PCV-2C. A taxa de classificação eleva-se na medida em que a taxa de rejeição aumenta, esta característica apresenta-se bastante interessante pois permite que se obtenha modelos com diferentes graus de confiança (quanto maior a rejeição mais confiável é a inferência do sistema), os quais podem ser selecionados de acordo com o nível de conhecimento do especialista médico.

Em geral, os classificadores propostos SOM-1C e SOM-2C, apresentam-se com melhores desempenhos que os classificadores MLP-1C e MLP-2C. Entre as abordagens propostas, o classificador SOM-2C é o que apresentou maior acurácia. A classificação com opção de rejeição apresenta-se bastante adequada no contexto da plataforma SINPATCO II e, portanto, deve ser adicionada ao seu módulo de diagnóstico.

A comparação dos resultados obtidos para os classificadores com opção de rejeição treinados com 60% e 80% do total de dados mostra que os classificadores SOM-2C e rejoSVM são similares e possuem desempenhos melhores que os classificadores SVM-1C, SVM-2C, MLP-1C, MLP-2C e SOM-1C. O classificador SOM-1C apresenta melhor desempenho que os classificadores MLP-1C, MLP-2C e SVM-2C para rejeição na faixa de 10% a 40% para treinamento com 80% do total de dados. Os resultados mostram, portanto, que os classificadores SOM-2C e rejoSVM são adequados para compor a plataforma SINPATCO II.

5 *Comitês de Classificadores*

Este capítulo apresenta propostas e resultados relacionados a estratégias de aprendizado que baseiam-se em Comitês de Classificadores. De início, discutem-se alguns estudos que buscam explicar porque um comitê de classificador apresenta melhor desempenho que um classificador isolado. Além disso, são apresentadas técnicas de treinamento (projeto) dos classificadores base do comitê, métodos de seleção destes classificadores, bem como as técnicas de combinação das saídas dos classificadores base. Ao final do capítulo são apresentados os resultados obtidos para comitês homogêneos e heterogêneos quando aplicados ao problema PCV-3C, construídos a partir dos classificadores MLP, GRNN e SVM.

5.1 Introdução

Muitos sistemas de auxílio ao diagnóstico médico usam modelos paramétricos, como discriminantes lineares gaussianos (LEE et al., 2005b; DOBROWOLSKI et al., 2008). Os custos elevados que resultam de um diagnóstico errado têm motivado uma busca constante por métodos de classificação mais acurados, de natureza não-linear e não-paramétrica, tais como redes neurais MLP e RBF. Em anos mais recentes tem aumentado também o número de aplicações de classificadores SVM no diagnóstico médico (DOBROWOLSKI et al., 2010; RAHMAN et al., 2011; ROCHA-NETO; BARRETO, 2009).

Independente do tipo de classificador, se paramétrico ou não, se linear ou não, usualmente em AM, de posse de um conjunto de dados referentes a uma aplicação específica, os vários classificadores escolhidos são treinados com tais dados, sendo selecionado para uso aquele com melhor desempenho global (i.e. menos erros). Pesquisas recentes sugerem que uma estratégia alternativa à seleção do melhor classificador individual consiste em empregar uma estratégia baseado na combinação de classificadores, formando comitês de classificadores ou *ensembles* (XU et al., 1992; BREIMAN, 1996; HANSEN; SALAMON, 1990). De fato, diversos autores fornecem evidências de que o uso de comitês conduz em geral a melhores resultados que o uso de apenas um classificador (BREIMAN, 1995, 1996; WOLPERT, 1992; ZHANG, 1999a, 1999b).

A idéia principal acerca do uso de comitês consiste em combinar um conjunto de modelos, em que cada um deles resolve o mesmo problema original, a fim de se obter um modelo global melhor, com previsões mais precisas e confiáveis do que se poderia obter com o uso de um único modelo simples. Em suma, um comitê é uma estratégia de aprendizado de máquina, na qual uma coleção finita de preditores (modelos) é utilizada em conjunto com o intuito de propor uma solução única para determinado problema.

Na literatura, o termo comitê (ou *ensemble*) é geralmente reservado para coleções de modelos que combinam o mesmo modelo-base (ou componente-base). No entanto, neste trabalho, também trata-se da combinação heterogênea de modelos que não pertencem à mesma estratégia de aprendizado de máquina. Este tipo de estratégia é normalmente referenciado na literatura especializada por sistemas de múltiplos classificadores (HO et al., 1994; CHINDARO et al., 2007). Porém, por questões de simplicidade, o termo comitê neste trabalho refere-se tanto à combinação de modelos-base diferentes (comitês heterogêneos) quanto à combinação de componentes-base da mesma família de algoritmos de aprendizado de máquina (comitês homogêneos).

A estratégia de formar um comitê inspira-se no comportamento humano, pois baseia-se na busca de diversas opiniões tomadas de forma individual, preferencialmente de especialistas, para então combiná-las com o intuito de se chegar a uma decisão final (POLIKAR, 2006). O primeiro trabalho relevante sobre comitês de classificadores mostra que a capacidade de generalização pode ser melhorada a partir da combinação das saídas de diversas redes neurais treinadas de forma independente (HANSEN; SALAMON, 1990). Apesar deste trabalho ser considerado como o primeiro avanço concreto na teoria dos comitês, a idéia de construir um modelo preditivo através da integração de vários outros modelos tem sido objeto de investigação há bastante tempo. A história dos comitês remonta ao final da década de 70 com o trabalho de Tukey (1977), que combinou dois modelos de regressão linear, tal que o primeiro modelo de regressão linear é ajustado aos dados, enquanto o segundo modelo ajusta-se ao resíduo. Dois anos mais tarde, em Dasarathy & Sheela (1979) é sugerida a partição do espaço de entrada usando dois ou mais classificadores. Este foi o primeiro trabalho sobre comitês em problemas de classificação de padrões.

Concomitantemente ao trabalho de Hansen & Salamon (1990), Schapire (1990) lançou as bases do conhecido algoritmo AdaBoost (FREUND; SCHAPIRE, 1996), mostrando que um classificador forte pode ser gerado pela combinação de classificadores base fracos¹. Vale ressaltar que, nesta tese, a atenção é voltada para o uso de comitês em um problema de classificação de

¹Um classificador é dito ser fraco (*weak*) se seu desempenho é apenas ligeiramente melhor do que a de um classificador aleatório

padrões, porém esta estratégia também pode ser utilizada em outros tipos de problemas, como regressão (GEY; POGGI, 2006; Y.KIM; KOO, 2005; TUTZ; BINDER, 2007) ou em estimação de densidades (RIDGEWAY, 2002).

Estratégia de classificação baseada em comitês tem sido bastante utilizada em diversas áreas, tais como: Finanças (LAM, 2000), Bioinformática (TAN; GILBERT, 2003), Quimioinformática (MERKWIRTH et al., 2004), Produção (ROKACH; MAIMON, 2006; ROKACH, 2008), Geografia (BRUZZONE et al., 2004), Segurança da Informação (MENAHEM et al., 2009; MOSKOVITCH et al., 2008), Recuperação da Informações (TAO X. TANG; WU, 2006) e Medicina (MANGIAMELI et al., 2004; XIANG et al., 2009; LEE et al., 2005a; LI et al., 2008; ZHOU; JIANG, 2003; TSYMBAL et al., 2003; ROCHA-NETO; BARRETO, 2009). Mais especificamente na área de Ortopedia, os trabalhos são bastante escassos para não dizer inexistentes. Apenas em 2009, já no contexto da plataforma SINPATCO II foi proposto um sistema que utiliza comitês de classificadores para resolver o problema de diagnóstico de patologias da coluna vertebral (ROCHA-NETO; BARRETO, 2009). Os resultados obtidos foram consequência dos estudos e pesquisas realizadas nesta tese de doutorado e serão apresentados na Seção 5.4.

A seguir são apresentados alguns conceitos teóricos que tentam responder a questão relacionada ao porque do ensembles serem melhores do que um modelo-base, bem como são apresentadas as vantagens alcançadas com o uso desta estratégia. Estes conceitos são descritos com base no trabalho de Nguyen (2006).

5.2 Fundamentação Teórica

Estudos experimentais conduzidos pela comunidade de aprendizado de máquina mostram que a combinação das saídas dos diversos modelos-base reduzem o erro de generalização (DOMINGOS, 1996; QUINLAN, 1996; BAUER; KOHAVI, 1999; BENNETT et al., 2002); mostram também que estratégias baseadas em comitês são muito eficientes, principalmente porque diversos tipos de classificadores possuem diferentes vieses (MITCHELL, 1997). De fato, comitês podem efetivamente fazer uso de tal diversidade com o intuito de reduzir a variância, sem entretanto aumentar o viés (ALI; PAZZANI, 1996; TUMER; GHOSH, 1996). Porém, em certas situações, um comitê também pode reduzir o viés, como mostrado na teoria dos classificadores de margem larga (BARTLETT; SHawe-Taylor, 1999).

Do exposto, o entendimento da razão de um comitê ser melhor do que um componente-base passa pelo entendimento do dilema viés/variância. Grosso modo, pode-se fazer a seguinte afir-

mação: o melhor modelo sobre o conjunto de treinamento deveria minimizar a diferença entre as saídas do modelo e a saída esperada. Porém, este processo de minimização não considera a presença de ruído nos dados. Assim, um modelo que aprenda com base neste processo também ajusta-se ao ruído que pode existir no conjunto de treinamento e, por isso, quando se apresenta um padrão não-visto pode resultar em um mau desempenho. Um modelo nesta situação é dito possuir baixo viés e alta variância. Na situação oposta em que um modelo possui baixa variância e alto viés, o modelo é menos dependente do conjunto de treinamento e possui boa capacidade de generalização.

Nesse contexto, a motivação subjacente à estratégia baseada em comitês é a busca pelo melhor compromisso entre o viés e a variância. Esta busca se dá pela combinação de componentes-base que possuam diferentes viéses/variâncias. Intuitivamente, a média de diferentes vieses/variâncias é o compromisso ótimo. Alguns pesquisadores têm tentado derivar um arcabouço teórico para provar que esta intuição é válida, tanto em problemas de regressão, quanto em problemas de classificação (KROGH; VEDELSBY, 1995; TUMER; GHOSH, 1996; UEDA; NAKANO, 1996).

5.2.1 Ambigüidade Decomposicional

Krogh & Vedelsby (1995) mostram que em problemas de regressão o erro quadrático de um comitê é menor do que o erro quadrático médio dos componentes-base. O erro quadrático médio do comitê pode ser dado por

$$(y_* - d)^2 = \sum_i w_i(y_i - d)^2 - \sum_i w_i(y_i - y_*) \quad (5.1)$$

em que $y_* = \sum_i w_i y_i$ é a saída ponderada da combinação das saídas dos componentes-base, d é a saída desejada e w_i é o peso dado para a i -ésima saída do modelo-base. O primeiro termo da Equação (5.1) é a média ponderada dos erros quadráticos dos modelos-base (i.e., a acurácia), enquanto o segundo termo é uma medida de variabilidade (i.e., diversidade) dos modelos-base que compõem o comitê. Este último é denominado termo de ambigüidade. Desde que o termo de ambigüidade seja sempre positivo, é garantido que o erro do comitê é sempre menor do que a média ponderada dos componentes-base individuais. Entretanto, apesar de que ao se aumentar o termo de ambigüidade ocorre a redução do erro global do comitê, percebe-se também que há uma tendência de que ocorra um aumento dos erros individuais dos modelos-base e, consequentemente, há um aumento do primeiro termo. Em outras palavras, apenas a diversidade não é suficiente, um correto compromisso entre a acurácia e a diversidade é essencial para garantir um melhor erro para o comitê (BROWN, 2004).

5.2.2 Decomposição Viés/Variância/Covariância

No trabalho de Ueda & Nakano (1996) é derivada uma outra decomposição útil, com base na decomposição viés/variação original para um estimador simples. Desde que a saída de um comitê seja a média simples das saídas dos modelos-base, ou seja

$$y_* = \frac{1}{M} \sum_{i=1}^M y_i \quad (5.2)$$

em que y_* é a saída do ensemble e y_i é a saída do i -ésimo componente-base, o erro quadrático médio de um comitê pode ser decomposto em

$$E [(y_* - d)^2] = \overline{\text{bias}}^2 + \frac{1}{M} \overline{\text{var}} + \left(1 - \frac{1}{M}\right) \overline{\text{covar}}, \quad (5.3)$$

em que $\overline{\text{bias}} = \frac{1}{M} \sum_{i=1}^M \text{bias}(y_i)$ é o viés condicional médio, $\overline{\text{var}} = \frac{1}{M} \sum_{i=1}^M \text{var}(y_i)$ é a variância condicional e $\overline{\text{covar}} = \frac{1}{M(M-1)} \sum_{i} \sum_{j, j \neq i}^M \text{cov}(y_i, y_j)$ é a covariância condicional. Uma derivação mais detalhada pode ser obtida em Ueda & Nakano (1996).

A decomposição mostrada na Equação (5.3) indica que o erro de generalização de um comitê também depende da correlação entre os modelos-base. Assim, um comitê no qual os componentes-base são descorrelacionados (i.e. diversos) generaliza melhor. Porém, esta decomposição é limitada aos métodos de combinação pela média simples e restrita a problemas de regressão.

5.2.3 Correlação do Erro de Classificação

Tumer & Ghosh (1996) propõem um arcabouço teórico para a decomposição da função de erro em problemas de classificação. Neste trabalho é assumido que os classificadores individuais estimam a probabilidade a posteriori da classe e que estas estimativas são então utilizadas para calcular a estimativa global do comitê. Uma vez que a fronteira de decisão estimada pode não ser a fronteira de decisão ótima, o erro total pode ser decomposto em dois termos, o erro de Bayes (E_{bayes}) e um erro adicionado (E_{add}^{ens}), tal que:

$$E_{total} = E_{bayes} + E_{add}^{ens}. \quad (5.4)$$

em que E_{total} é o erro total do comitê e E_{add}^{ens} é o erro adicionado do comitê.

Visto que o erro de Bayes não pode ser alterado, o comitê somente pode ser aprimorado pela alteração do segundo termo (E_{add}^{ens}). Neste trabalho, o erro adicionado de um comitê (E_{add}^{ens})

é dado por

$$E_{add}^{ens} = \frac{1 + \delta(M - 1)}{M} E_{add}^{ind}, \quad (5.5)$$

em que E_{add}^{ind} é o erro adicionado de um modelo-base (também é assumido que o erro adicionado de todos os componentes são iguais para todos os componentes-base) e δ representa a correlação entre os erros de cada classificador. Logo, se os erros dos modelos-base forem totalmente correlacionados (i.e. $\delta = 1$), nenhuma melhoria advém do uso de comitês, i.e. $E_{add}^{ens} = E_{add}^{ind}$. Entretanto, caso os erros sejam descorrelacionados (i.e. $\delta = 0$), então o erro adicionado dos comitês é reduzido por M , ou seja, $E_{add}^{ens} = \frac{E_{add}^{ind}}{M}$.

5.3 Projeto de Comitês

O projeto de um comitê envolve três tarefas distintas. A primeira é a geração de componentes-base que são os classificadores ou modelos que compõem o comitê. A segunda tarefa envolve a escolha dos componentes-base que comporão o comitê. Eventualmente, todos os modelos-base podem ser utilizados para a composição do comitê. Por fim, a terceira tarefa envolve a combinação das saídas dos componentes-base. Esta tarefa consiste em definir a saída do comitê com base nas saídas produzidas pelos componentes-base. A seguir são dados detalhes sobre cada uma dessas tarefas.

5.3.1 Geração de Componentes-Base

O processo de geração dos modelos-base é bastante importante para que se consiga alcançar a melhoria do desempenho em um comitê. Como descrito anteriormente, um comitê necessita de componentes-base com bom desempenho, bem como necessita que a tomada de decisão dos modelos-base seja a mais descorrelacionada possível umas das outras. A obtenção de modelos descorrelacionados pode ser obtida por (i) variação de parâmetros e aspectos estruturais dos modelos-base ou por (ii) variação nos dados de treinamento.

No tocante à variação dos parâmetros e aspectos estruturais dos modelos, como exemplo, pode-se citar o uso de valores diferentes para os pesos iniciais em uma rede neural. Esta situação pode propiciar a convergência para um conjunto de pesos distintos, visto que a convergência pode se dar para um ótimo local. Assim, espera-se que mantidos a mesma arquitetura da rede, mesmo algoritmo de treinamento e mesmo conjunto de treinamento, a rede consiga generalizar de forma diferente. Similarmente, as alterações na arquitetura de uma rede neural também podem conduzir a componentes-base que generalizam de forma diferente, mesmo mantendo-

se o algoritmo de treinamento e o conjunto de treinamento. O argumento também é válido para utilização de diferentes algoritmos de treinamento, por exemplo, gradiente descendente ou Levenberg-Marquardt. Nesta situação mesmo que sejam mantidos inalterados os dados de treinamento a, arquitetura da rede neural e as condições iniciais dos pesos, a convergência da rede pode ocorrer de forma distinta e conduzir a diferentes ótimos locais.

No entanto, as técnicas mais freqüentemente empregadas para a criação de comitês atuam sobre a formação dos conjuntos de treinamento dos modelos-base. Tal estratégia permite que modelos com poucos parâmetros de treinamento ou que possuam certa estabilidade possam ser utilizados como modelos-base. Há na literatura várias descrições de técnicas de geração de conjuntos de treinamento, as quais podem ser aplicadas conjuntamente. O objetivo continua sendo a obtenção de modelos-base que generalizam de forma diversa entre si, e esta diversidade é perseguida pela produção de conjuntos de treinamento distintos. Com este propósito, podem ser listadas as seguintes técnicas:

- **Conjuntos de treinamento disjuntos:** esta técnica baseia-se na obtenção de conjuntos de treinamento mutuamente exclusivos (disjuntos). Comumente é realizado um processo de amostragem sem repetição (SHARKEY, 1996). Portanto, não há sobreposição de dados usados para treinar os diversos componentes-base. Como observado por Tumer & Ghosh (1996), o problema é que o tamanho dos conjuntos de treinamento pode ficar reduzido, e podendo resultar em desempenho de generalização ruim.
- **Reamostragem dos dados:** técnica bastante utilizada para obtenção de conjunto de treinamento distintos, que se caracteriza por criar subconjuntos dos dados de treinamento tal que cada subconjunto tenha pelo menos um padrão que não exista em outro subconjunto. Uma das técnicas mais difundidas e utilizadas é a *Bootstrap Aggregating* (Bagging) (BREIMAN, 1996).
- **Reamostragem adaptativa:** nesta técnica os conjuntos de treinamento são reamostrados de forma adaptativa, de tal maneira que os padrões que mais contribuem para o erro de treinamento dos modelos-base já treinados, em uma fase anterior, têm maior probabilidade de pertencerem a um novo conjunto de treinamento a ser utilizado na síntese de um próximo modelo. Nesta técnica, obviamente, os componentes-base do comitê devem ser obtidos de maneira seqüencial. Uma das técnicas de reamostragem adaptativa mais utilizadas, o Adaboost (*Adaptive boosting*), foi proposto por Sharkey (1996).

A seguir são apresentados em maior nível de detalhes as técnicas Bagging, Adaboost e de geração de conjuntos disjuntos.

Bagging

O aspecto relevante da técnica Bagging consiste na promoção de diversidade dos conjuntos de treinamento obtidos por amostragem bootstrap (EFRON, 1982). Um exemplo de processo de treinamento/teste baseado em Bagging é descrito a seguir:

1. Separar o conjunto total de dados em dois subconjuntos de treinamento (T_r) e teste (T_e), tal que N e M denotam os tamanhos do conjunto T_r e T_e , respectivamente.
2. Realizar reamostragem (com reposição) do conjunto de treinamento, com mesma probabilidade de escolha para cada um dos padrões de treinamento. Este passo é realizado k vezes para produção de k conjuntos reamostrados, com N padrões cada. O i -ésimo subconjunto reamostrado é representado por $T_r^{(i)}$, tal que $i = 1, \dots, k$.
3. Realizar o treinamento do i -ésimo componente-base $C^{(i)}$ utilizando o i -ésimo conjunto reamostrado $T_r^{(i)}$. Ao final, tem-se k modelos-base treinados, cada um utilizando um conjunto reamostrado diferente.
4. Realizar a seleção de componentes-base para compor o comitê, com base em algum critério. Pode-se utilizar todos os componentes-base gerados, se for conveniente.
5. Utilizar alguma função que realize uma combinação das saídas dos modelos-base que constituem o comitê.
6. Avaliar a capacidade de generalização do comitê sobre o conjunto de teste T_e .

Algumas considerações importantes fazem-se necessárias. Todos os padrões dos k conjuntos reamostrados estão presentes no conjunto de treinamento T_r , de tal maneira que a diferença entre os k conjuntos gerados está na presença de padrões repetidos. Como consequência, pode ser verificada a ausência de alguns padrões que compõem o conjunto original T_r . Como é assumida uma mesma probabilidade de escolha para qualquer um dos N padrões, e como N seleções são realizadas para preencher cada um dos k conjuntos de treinamento $T_r^{(i)}$, então a reposição de padrões permite que um padrão já escolhido possa ser selecionado novamente durante a composição de um dos conjuntos de treinamento $T_r^{(i)}$. Logo, para cada padrão repetido, na formação do conjunto reamostrado $T_r^{(i)}$, implica na ausência de algum padrão do conjunto de treinamento T_r .

Vale destacar que no Bagging existe compromisso entre a capacidade de generalização dos componentes-base, que não pode ser muito ruim, e a capacidade de generalização dos componentes não precisa ser muito boa. Assim, o grau ótimo de regularização dos componentes do Bagging deve ser adequadamente sintonizado para cada problema de aplicação (TANIGUCHI; TRESP, 1997).

Boosting e AdaBoosting

A estratégia Boosting utilizada para geração de componentes-base foi proposta por Schapire (1990). Esta técnica é normalmente denominada *boosting* por filtragem. Nesta técnica são gerados apenas três componentes-base. O primeiro componente-base é treinado utilizando N_1 padrões de treinamento, obtidos do conjunto de treinamento de tamanho N . O segundo componente-base é treinado utilizando N_2 padrões do conjunto de treinamento. Estes padrões são selecionados a partir dos dados de treinamento, tal que 50% deles devem ter sido classificados corretamente pelo primeiro componente-base. Este processo possibilita que o segundo componente-base seja treinado utilizando um conjunto com uma distribuição bastante diferente da que gerou o primeiro componente-base. Enquanto o terceiro componente-base é treinado com N_3 padrões que consistem nos exemplos em que os dois componentes treinados discordam quanto à classe a ser atribuída. A classificação final é então realizada pelo voto majoritário simples das saídas dos três componentes. Esta técnica assume que se dispõe de uma quantidade grande de padrões.

A motivação original para Boosting está fundamentada na teoria de aprendizado PAC (*Probably Approximately Correct learning*) (VALIANT, 1984). Esta técnica também requer que os componentes-base apresentem pelo menos um desempenho superior àquele obtido por um classificador aleatório.

A estratégia AdaBoost é uma combinação de conceitos utilizados tanto no Boosting quanto pelo Bagging, tal que não se exige que o conjunto de treinamento possua uma grande quantidade de padrões (FREUND; SCHAPIRE, 1996).

Similarmente ao Bagging, no AdaBoosting, os componentes-base do ensemble são treinados usando reamostragem com reposição. Porém, a probabilidade de seleção de um padrão depende do desempenho dos componentes-base treinados anteriormente. Neste contexto, se um determinado padrão p_j for incorretamente classificado pelo componente-base $C^{(i)}$ gerado do conjunto de treinamento reamostrado $T_r^{(i)}$, então na reamostragem do próximo conjunto de treinamento $T_r^{(i+1)}$ adota-se uma maior probabilidade de escolha do padrão p_j . Considera-se um incremento ainda maior de probabilidade caso o desempenho geral do componente-base $C^{(i)}$ seja bom. Esta técnica utiliza uma função de combinação de componentes específica, tal que há um peso para o i -ésimo componente-base na votação do comitê, de tal maneira que se dá maior ênfase aos componentes-base que possuem bom desempenho. Mais detalhes podem ser obtidos em (FREUND; SCHAPIRE, 1996).

Conjuntos Disjuntos: uma forma bastante simples de gerar componentes consiste em utili-

zar uma amostragem sem repetição. Pode-se, por exemplo, separar o conjunto de dados em três conjuntos disjuntos: treinamento, validação e teste. Os componentes-base podem ser gerados utilizando o conjunto de treinamento, enquanto o conjunto de validação é utilizado para seleção dos modelos-base. Como não poderia deixar de ser, o desempenho é calculado sobre o conjunto de teste.

5.3.2 Seleção de Componentes-Base

A seleção de componentes-base visa maximizar o desempenho de generalização do comitê pela definição de um subconjunto dentre o total de candidatos a compor o mesmo. O processo de seleção pode ocorrer de duas maneiras diferentes, a saber:

- (i) Pela aplicação de procedimentos de seleção a um conjunto de candidatos que foram gerados por meio do uso de métodos concebidos para promover diversidade.
- (ii) Pela realização de um processo continuado de geração e seleção até que um critério de parada seja alcançado.

A segunda destas possibilidades é explorada por Opitz et al. (1996), que apresentam um método que usa algoritmos genéticos para encontrar componentes para comitê que generalizem bem e que apresentem baixa correlação. Trata-se de um método evolucionário que trabalha com uma população de candidatos. Os candidatos selecionados para compor o comitê são combinados via média ponderada. O critério de parada do processo de seleção de componentes depende dos recursos computacionais disponíveis e do nível de desempenho esperado para o comitê.

5.3.3 Combinação de Componentes

Diversos métodos de combinação de componentes estão disponíveis na literatura. Nesta tese, apenas os seguintes métodos são discutidos a seguir: Voto Majoritário Simples, Média Simples e Média Ponderada. Na descrição a seguir, assume-se que os classificadores-base usam a codificação de saída *1-out-of-m*, i.e. o vetor de saídas desejadas tem m componentes e apenas uma delas é não-nula.

Voto Majoritário Simples: a classe com o maior número de votos, dados pelos componentes-base do comitê, é selecionada. Seja $y_{k,l}$ a k -ésima saída do l -ésimo componente-base, e $y_{k,l}^*$ a saída quantizada para 0 ou 1. Logo, o vetor de saídas quantizadas do l -ésimo componente-base

do comitê, para um dado padrão de entrada \mathbf{x} , é representada como

$$\mathbf{y}_l^*(\mathbf{x}) = [y_{1,l}^*(\mathbf{x}) \quad y_{2,l}^*(\mathbf{x}) \quad \dots \quad y_{m,l}^*(\mathbf{x})]^T, \quad (5.6)$$

em que m é o número de classes. Seja $\mathbf{v}(\mathbf{x})$ o vetor resultante da soma dos vetores de saída $\mathbf{y}_l^*(\mathbf{x})$ de todos os componentes-base de um comitê:

$$\mathbf{v}(\mathbf{x}) = \sum_{i=1}^l \mathbf{y}_l^*(\mathbf{x}). \quad (5.7)$$

Pode-se inferir qual o índice a classe $C(\mathbf{x})$ mais votada de um comitê implementando-se a seguinte regra de decisão:

$$C(\mathbf{x}) = \arg \max_{k=1,\dots,m} \{v_k(\mathbf{x})\} \quad (5.8)$$

em que $v_k(\mathbf{x})$ é o k -ésimo elemento do vetor $\mathbf{v}(\mathbf{x})$.

Média Simples: obtém-se uma saída única com base nas diversas saídas dos componentes-base, em seguida atribui-se o padrão à classe com maior valor médio (PERRONE; COOPER, 1993). Seja $y_{k,l}$ a k -ésima saída do l -ésimo componente-base (i.e. saída não-quantizada). Logo, o vetor de saídas do l -ésimo componente-base do comitê, para um dado padrão de entrada \mathbf{x} , é representada como

$$\mathbf{y}_l(\mathbf{x}) = [y_{1,l}(\mathbf{x}) \quad y_{2,l}(\mathbf{x}) \quad \dots \quad y_{m,l}(\mathbf{x})]^T, \quad (5.9)$$

em que m é o número de classes. Seja $\mathbf{v}(\mathbf{x})$ a média simples dos vetores de saída $\mathbf{y}_l^*(\mathbf{x})$ de todos os componentes-base de um comitê:

$$\mathbf{v}(\mathbf{x}) = \frac{1}{l} \sum_{i=1}^l \mathbf{y}_l(\mathbf{x}). \quad (5.10)$$

De forma similar, ao realizado para o voto majoritário simples, pode-se inferir qual o índice da classe $C(\mathbf{x})$ mais votada de um comitê implementando-se a seguinte regra de decisão:

$$C(\mathbf{x}) = \arg \max_{k=1,\dots,m} \{v_k(\mathbf{x})\} \quad (5.11)$$

em que $v_k(\mathbf{x})$ é o k -ésimo elemento do vetor $\mathbf{v}(\mathbf{x})$.

Média Ponderada: similar à média simples. No entanto, calcula-se a média ponderada

$$\mathbf{v}(\mathbf{x}) = \frac{1}{l} \sum_{i=1}^l \mathbf{w}_i \mathbf{y}_l(\mathbf{x}), \quad (5.12)$$

em que \mathbf{w}_i representa o peso do i -ésimo componente-base. De forma similar, pode-se inferir

qual a classe $C(\mathbf{x})$ mais votada de um comitê implementando-se a seguinte regra de decisão:

$$C(\mathbf{x}) = \arg \max_{k=1,\dots,m} \{v_k(\mathbf{x})\}. \quad (5.13)$$

Diversos outros métodos levam em consideração o desempenho dos componentes na tomada de decisão e pode-se citar: métodos bayesianos (FRENCH, 1985; LEE, 1997), supra-bayesianos (JACOBS, 1995), método BKS (*Behavior-Knowledge Space*) (HUANG; SUEN, 1995; WERNECKE, 1992), Método Dempster-Shafer (XU et al., 1992), e método de empilhamento (LEBLANC; TIBSHIRANI, 1996).

5.4 Simulações Computacionais

Antes do treinamento de todos os classificadores foram criados três conjuntos a partir do conjunto T com 310 padrões disponíveis: um conjunto W livre de *outliers*, um conjunto N contendo somente *outliers*, e um terceiro conjunto S relacionado ao primeiro e ao segundo conjunto da seguinte forma $S = W \cup N_p$, em que N_p representa um conjunto contendo $P\%$ dos *outliers* de N . Vale observar que $T = W \cup N$ e que $S \subset T$. Os exemplos de treinamento e teste são selecionados do conjunto S . Os atributos são normalizados para a faixa 0 e 1. A base de dados utilizada neste estudo sobre ensembles é sempre a PCV-3C.

Para cada classificador são realizadas 50 rodadas de treinamento/teste. Os padrões de treinamento são selecionados aleatoriamente do conjunto S , sendo as restantes usadas para teste. Inicialmente, trabalha-se com $P = 0\%$, resultando em um conjunto S igual ao conjunto W , ou seja, livre de *outliers*. O valor de P é então incrementado em 20 unidades, até atingir $P = 100\%$, quando então todos os *outliers* são adicionados ao conjunto S . Após as 50 rodadas de treinamento/teste, correspondentes a um valor de P específico, são determinadas a taxa média de acerto (acurácia) e o desvio-padrão das taxas de acerto observadas durante os testes.

A remoção remoção de *outliers* foi realizada calculando-se a distância de Mahalanobis (WEBB, 2002) de cada exemplo de uma classe ao centróide da respectiva classe. Se a distância for maior que certo limiar, o exemplo é considerado um outlier. Matematicamente, considera-se que $\mathbf{x} \in \mathbb{R}^n$ é um outlier da i -ésima classe, se a seguinte condição for satisfeita:

$$\sqrt{(\mathbf{x} - \mu_i)^T \mathbf{C}_i^{-1} (\mathbf{x} - \mu_i)} > \beta K, \quad (5.14)$$

em que $0 < \beta \leq 1$ é uma constante arbitrária, K é o valor crítico da distribuição Qui-quadrado com n graus de liberdade e nível de significância α , μ_i é o vetor de médias e \mathbf{C}_i é a matriz de covariância. Os valores utilizados foram $\beta = 0,3$, $\alpha = 0,95$ e $n = 6$.

A Tabela 5.1 apresenta o número de padrões contidos em S para cada percentagem de padrões discrepantes adicionada. Pode-se observar nesta tabela que para ausência de padrões discrepantes (0%) o número de padrões é igual a 195, para 100% igual a 310 padrões, enquanto para os conjuntos intermediários acréscimos de 23, 46, 69 e 92 padrões respectivamente para 20%, 40%, 60% e 80%. O número de padrões por classe para um dado conjunto S não é fixo, visto que apesar do número acrescentado ser fixo (por exemplo 23 padrões para 20% de *outliers*) os padrões são adicionados aleatoriamente do conjunto N .

<i>Outliers</i>	<i>Total de Padrões</i>
0%	195
20%	218
40%	241
60%	264
80%	287
100%	310

Tabela 5.1: Número de padrões do conjunto S por percentagem de *outliers*.

Além das taxas médias de acerto (acurácia) e seus desvios-padrão, os classificadores são também comparados entre si com base no número de falsos negativos e falsos positivos gerados. O estudo de falso negativo e falso positivo é feito normalmente via matriz de confusão que relaciona a classificação dada ou predita com a situação do indivíduo (saudável, herniado ou com espondilolistese). Como o objetivo é diferenciar indivíduos que apresentam-se sadios dos que estão com espondilolistese ou hérnia de disco, uma matriz de confusão para este fim deve conter logicamente 3 linhas e 3 colunas. Assim, pode-se assumir que, as linhas da matriz relacionam-se a situação do indivíduo e as colunas relacionam-se a predição realizada por um classificador. Os valores na primeira linha representam indivíduos com hérnia de disco, na segunda indivíduos com espondilolistese e na última indivíduos saudáveis. As colunas seguem a mesma ordem de distribuição contudo relacionam-se a predição realizada pelos classificadores.

	<i>diagnóstico 1</i>	<i>diagnóstico 2</i>	<i>diagnóstico 3</i>
<i>diagnóstico 1</i>	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$
<i>diagnóstico 2</i>	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$
<i>diagnóstico 3</i>	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$

Tabela 5.2: explanação da matriz de confusão.

Com base na descrição feita anteriormente em conjunto com a Tabela 5.2, pode-se afirmar que os elementos da diagonal principal da matriz de confusão, $a_{1,1}, a_{2,2}$ e $a_{3,3}$, representam os diagnósticos corretos, ou seja, a classificação inferida é igual a situação do indivíduo. Já os

falso-negativos são representados pelos elementos $a_{1,3}$ e $a_{2,3}$, pois os indivíduos herniados e/ou com espondilolistese são dados como saudáveis; enquanto, os falsos positivos são representados pelos elementos $a_{3,1}$ e $a_{3,2}$, quando, indivíduos saudáveis são diagnosticados como patológicos.

5.4.1 Classificadores Considerados Isoladamente

Para os classificadores SVM, MLP e GRNN separou-se aleatoriamente o conjunto S na proporção 80% para treino e 20% para teste (*hold out*). A rede MLP foi configurada com apenas 1 camada oculta contendo 12 neurônios, definida por experimentação, enquanto a camada de saída possui 3 neurônios, um para cada classe, com base na estratégia *one-out-of-m encoding*. Logo, o rótulo numérico da classe de indivíduos normais é [1 0 0], para classe de indivíduos com hérnia de disco é [0 1 0] e para a classe com espondilolistese o rótulo é [0 0 1]. A função de ativação logística é utilizada por cada um dos neurônios da rede MLP. Os pesos e os limiares de ativação são ajustados através do algoritmo backpropagation padrão por 2500 épocas, com taxa de aprendizagem igual a 0,01.

No tocante ao classificador SVM, utilizou-se o algoritmo *Kernel* Adatron (CAMPBEL; CRISTIANINI, 1998) para determinação do viés e dos multiplicadores de Lagrange. O kernel KMOD (AYAT et al., 2002) foi utilizado, com os seguintes parâmetros $\lambda = 8,0$ e $\sigma = 2,5$, os quais foram determinados experimentalmente. O parâmetro de regularização C é fixado em 0,05 e o número de épocas em 500, enquanto, o raio da i -ésima função de base da rede GRNN é calculado como sendo igual à média das distâncias do centro da i -ésima função de base aos cinco centros mais próximos. No classificador GRNN, são utilizados todos os padrões de treinamento como funções de base e, ressalta-se também que neste tipo de classificador não há treinamento, no sentido de serem necessárias várias épocas de apresentação dos dados. Os valores obtidos para a taxa de acerto médio no teste e os respectivos desvios-padrões, em função de P , são apresentados na Tabela 5.3.

P(%)	SVM	MLP	GRNN
0	$96,51 \pm 3,65$	$98,70 \pm 1,76$	$96,51 \pm 3,46$
20	$88,68 \pm 7,39$	$93,39 \pm 3,90$	$90,00 \pm 4,94$
40	$86,33 \pm 7,25$	$90,02 \pm 4,27$	$84,79 \pm 5,72$
60	$85,47 \pm 7,03$	$87,90 \pm 4,63$	$80,94 \pm 5,18$
80	$83,47 \pm 5,06$	$83,53 \pm 5,99$	$78,98 \pm 4,70$
100	$82,16 \pm 4,95$	$83,03 \pm 5,70$	$75,41 \pm 5,58$

Tabela 5.3: Resultados para os classificadores individuais SVM, MLP e GRNN.

Ao analisar essa tabela, percebe-se que o desempenho dos três classificadores é máximo

quando *outliers* não estão presentes nos dados (i.e. $P = 0\%$). Como esperado, à medida que P aumenta o desempenho dos classificadores se deteriora. Verifica-se ainda que para $P = 100\%$ as taxas médias de acerto dos classificadores SVM e MLP são superiores a 82%, valores considerados bons pelo médico ortopedista consultado. No tocante ao classificador GRNN verifica-se ainda que o desempenho deste classificador é inferior, na média, ao desempenho dos classificadores MLP e SVM para qualquer porcentagem de *outliers* adicionada (exceção para $P = 20\%$, caso em que o classificador SVM é inferior). A Tabela 5.4 apresenta a melhor matriz de confusão gerada pelo classificador SVM. Os resultados mostrados nesta tabela correspondem aos obtidos na melhor dentre as 50 rodadas de treinamento/teste, para um valor fixo de $P = 100\%$. A obtenção da matriz de confusão para os demais classificadores segue o mesmo procedimento.

	Hérnia Disco	Espondilolistese	Normal
Hérnia Disco	7	0	5
Espondilolistese	1	32	2
Normal	1	0	14

Tabela 5.4: Melhor matriz de confusão - Classificador SVM.

Verifica-se na Tabela 5.4 a ocorrência de um falso positivo (indivíduo normal considerado pelo sistema como herniado) e de 7 falsos negativos (5 indivíduos herniados e 2 indivíduos com espondilolistese categorizados como normais). Ocorre ainda a classificação incorreta entre patologias, com um caso de indivíduo com espondilolistese sendo categorizado pelo classificador SVM como portador de hérnia de disco. A partir da análise da melhor matriz de confusão obtida para o classificador MLP (Tabela 5.5), verifica-se a ocorrência de 3 falsos positivos, 2 falsos negativos e um erro de classificação entre patologias, com o classificador inferindo hérnia de disco quando o indivíduo possui na verdade espondilolistese. Verifica-se ao se comparar as matrizes apresentadas na Tabela 5.4 e na Tabela 5.5 que há 5 falsos negativos a menos e 2 falsos positivos a mais para o classificador MLP.

	Hérnia Disco	Espondilolistese	Normal
Hérnia Disco	10	0	2
Espondilolistese	1	24	0
Normal	1	2	22

Tabela 5.5: Melhor matriz de confusão - Classificador MLP.

A Tabela 5.6, apresenta a melhor matriz de confusão obtida para o classificador GRNN. Verifica-se a ocorrência de 7 falsos positivos, 3 falsos negativos e a classificação incorreta de

5 indivíduos, categorizados como possuindo espondilolistese quando na verdade deveriam ter sido categorizados como portadores de hérnia de disco. Pode-se concluir, a partir desta tabela e das outras duas descritas anteriormente, que dentre os 3 classificadores analisados individualmente, o classificador GRNN foi o que obteve o pior resultado quanto ao número de falsos positivos e falsos negativos. Do ponto de vista estatístico (teste-t), o desempenho individual do classificador SVM é equivalente ao do classificador MLP.

	Hérnia Disco	Espondilolistese	Normal
Hérnia Disco	6	5	2
Espondilolistese	0	28	1
Normal	3	4	13

Tabela 5.6: Melhor matriz de confusão - Classificador GRNN.

5.4.2 Comitês de Classificadores (Proposta 5)

Resultados para Comitês Homogêneos

Os comitês de SVM, MLP e GRNN são construídos e avaliados com base nas seguintes proporções de conjunto de dados: 60% para treinamento, 20% para seleção e 20% para teste. Comitês homogêneos são compostos por $L = 5$ classificadores do mesmo tipo, escolhidos dentre os modelos obtidos das 50 rodadas treino/seleção. Os cinco escolhidos são aqueles com maiores taxas de acerto médio no conjunto de seleção. Quantidades variadas de classificadores para composição de comitês foram examinadas, porém, a composição com cinco classificadores foi a que apresentou os melhores resultados.

No caso do comitê de MLPs, foram avaliados diferentes números de neurônios ocultos, variando de 9 a 15 unidades e até mesmo MLPs com duas camadas ocultas; porém, os resultados nestes casos não se mostraram estatisticamente superiores aos dos comitês de MLPs com uma única camada oculta com 12 neurônios.

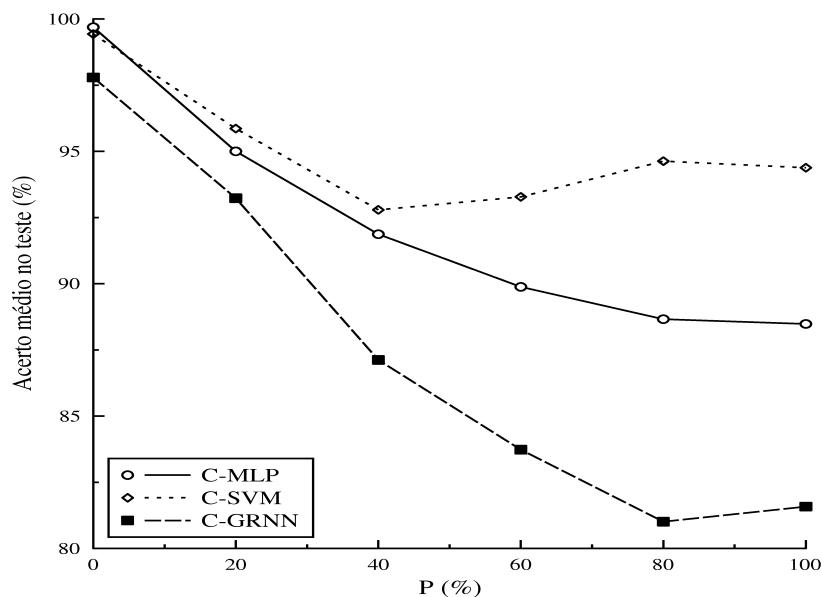
A Tabela 5.7 mostra os resultados obtidos para comitês homogêneos de SVM (C-SVM), MLP (C-MLP) e GRNN (C-GRNN). Os valores constantes nesta tabela são apresentados na Figura 5.1.

Na Figura 5.1 é possível notar que o comitê de SVMs apresenta-se com melhor desempenho de generalização dentre os comitês homogêneos, seguido pelo comitê de MLPs e pelo comitê de GRNNs, nesta ordem. Exceto para $P = 0\%$, com uma pequena diferença de 0,26%, todos os valores médios da taxa de acerto foram melhores para o comitê de SVM do que para o comitê

P(%)	C-SVM	C-MLP	C-GRNN
0	99,43 ± 1,07	99,69 ± 0,84	97,79 ± 1,93
20	95,86 ± 2,66	95,00 ± 3,24	93,22 ± 4,26
40	92,79 ± 2,98	91,87 ± 3,12	87,12 ± 4,77
60	93,28 ± 3,21	89,88 ± 3,65	83,73 ± 5,40
80	94,63 ± 2,41	88,66 ± 3,70	81,01 ± 4,71
100	94,38 ± 2,86	88,48 ± 3,93	81,58 ± 4,78

Tabela 5.7: Resultados para comitês homogêneos de $L = 5$ classificadores.

de MLP. Para o comitê de GRNNs, as taxas médias de acerto sempre se apresentam piores do que as obtidos para os comitês de MLPs e SVMs.

**Figura 5.1:** Curvas das taxas de acerto médio em função de P% para comitês homogêneos de $L = 5$ classificadores.

A Figura 5.2 apresenta os diagramas de caixa (*boxplots*) das taxas de acerto geradas pelos comitês homogêneos para $P = 100\%$, após 50 rodadas de treinamento/seleção/teste. Analisando os diagramas percebe-se uma maior dispersão das taxas de acerto para os comitês homogêneos de GRNNs e MLPs do que para o comitê de SVMs. Observa-se também que os valores da taxa de acerto para o comitê de SVMs encontram-se nos piores casos próximos a 89%, e nos melhores casos próximos a 100%, atingindo inclusive 100% de acerto algumas vezes. A grande maioria dos valores encontra-se na faixa entre 93,5% e 96,77%. Para o comitê de MLPs, os menores valores da taxa de acerto encontram-se próximos a 81%, os maiores próximos a 99%, e a maioria na faixa entre 86,28% e 91,93%. Por fim, o comitê de GRNNs apresenta os menores valores de taxa de acerto próximos a 71%, os maiores valores próximos a 92%, e a grande maioria dos resultados na faixa de 79,03% a 84,67%.

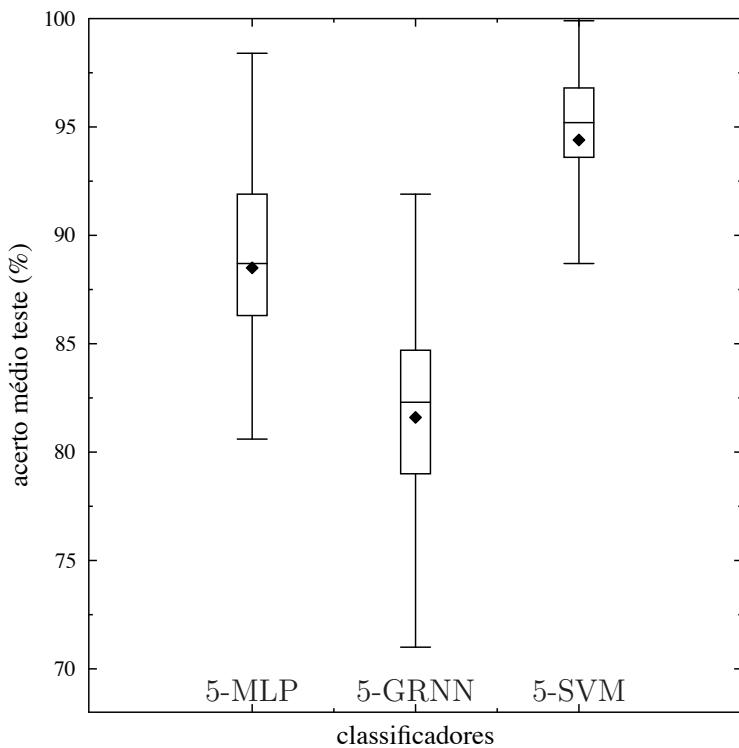


Figura 5.2: Diagrama de caixas (boxplots) das taxas de acerto de classificação para os comitês homogêneos ($P = 100\%$).

A matriz de confusão do comitê de SVMs apresentada na Tabela 5.8, para os mesmos exemplos de teste usados para gerar a Tabela 5.4, não contém falsos positivos e falsos negativos, diferentemente do que ocorre com o classificador individual. A matriz de confusão do comitê de MLPs apresentada na Tabela 5.9, para os mesmos exemplos de teste usados para gerar a Tabela 5.5, apresenta um caso a menos de falso positivo. Por fim, a matriz de confusão do comitê de GRNNs apresentada na Tabela 5.10, para os mesmos exemplos de teste usados para gerar a Tabela 5.6, apresenta um caso a menos de falso positivo e um caso a menos de falso negativo.

	Hérnia Disco	Espondilolistese	Normal
Hérnia Disco	12	0	0
Espondilolistese	2	33	0
Normal	0	0	15

Tabela 5.8: Matriz de confusão - Comitê C-SVM.

Em suma, a análise dos resultados dos comitês homogêneos revela que os resultados produzidos pelo comitê de SVMs são significativamente melhores que os resultados obtidos para os outros comitês homogêneos.

	Hérnia Disco	Espondilolistese	Normal
Hérnia Disco	10	0	2
Espondilolistese	1	24	0
Normal	1	1	23

Tabela 5.9: Matriz de confusão - Comitê C-MLP.

	Hérnia Disco	Espondilolistese	Normal
Hérnia Disco	6	5	2
Espondilolistese	0	29	0
Normal	2	4	14

Tabela 5.10: Matriz de confusão - Comitê C-GRNN.

Resultados para Comitês Heterogêneos

No tocante aos comitês heterogêneos vários resultados foram obtidos. Estes comitês são constituídos por cinco componentes variados (i.e. dois ou três tipos de classificadores diferentes). As configurações dos comitês heterogêneos testadas são combinações de classificadores MLP/GRNN, GRNN/SVM, MLP/SVM, ou ainda de MLP/GRNN/SVM. A Figura 5.3 mostra os resultados das melhores configurações de comitês heterogêneos. Os valores mostrados correspondem às taxas de acerto médio após 50 rodadas de treinamento/seleção/teste para diferentes valores de $P\%$. Uma rápida análise da figura permite verificar que o comitê com 1 classificador MLP e 4 classificadores SVM (1-MLP 4-SVM) apresenta o melhor desempenho de classificação.

A Figura 5.4 permite comparar os desempenhos dos comitês homogêneos com o desempenho do melhor comitê heterogêneo. A partir da análise desta figura, percebe-se que o desempenho do comitê homogêneo 5-SVM é muito similar ao do comitê heterogêneo 1-MLP/4-SVM. O comitê 5-SVM somente apresenta resultados ligeiramente inferiores ao comitê 1-MLP/4-SVM quando $P = 0\%$ e $P = 80\%$, porém estas diferenças não são significativas do ponto de vista estatístico. Assim, pela maior simplicidade na implementação (pois requer apenas um tipo de classificador), é dada preferência ao comitê 5-SVM.

Um fato interessante observado nas simulações é que para todas as configurações de comitês heterogêneos que envolveram classificadores SVM, os melhores resultados sempre foram obtidos quando o número de classificadores SVMs era dominante sobre os outros tipos de classificadores. Por exemplo, para a configuração GRNN/SVM, a melhor combinação foi 1-GRNN/4-SVM; para a configuração MLP/SVM, o melhor comitê encontrado foi 1-MLP/4-SVM; e, por fim, para a configuração MLP/GRNN/SVM a melhor combinação encontrada foi

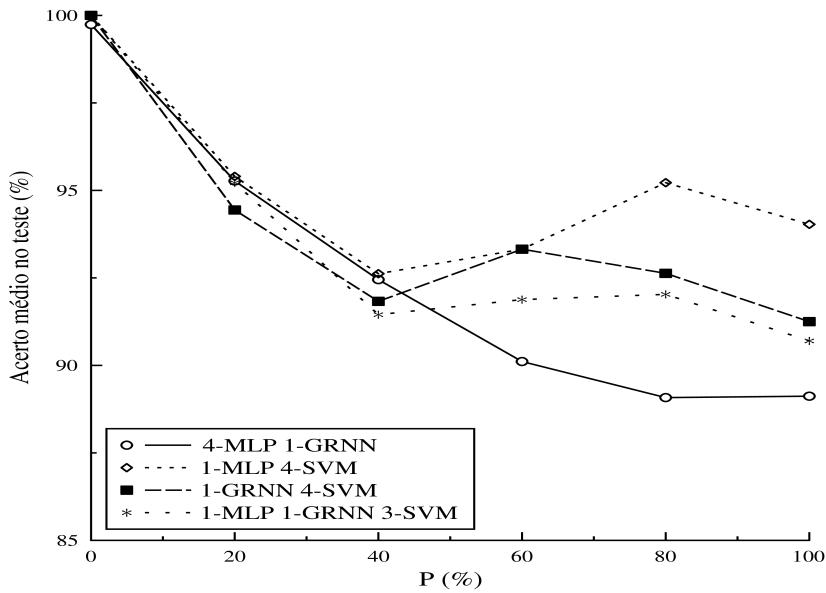


Figura 5.3: Gráfico do desempenho de classificação em função de P% para os melhores comitês heterogêneos com $L = 5$ classificadores-base.

1-MLP/1-GRNN/3-SVM.

Para fins de avaliação dos tipos de *kernel* que compõem o comitê C-SVM, realizou-se também o treinamento dos classificadores-base SVM utilizando os *kernels* RBF e linear para P=100%. A acurácia de 86,9% ($\pm 4,6\%$) foi obtida para o comitê C-SVM/RBF e de 86,6% ($\pm 3,4\%$) para o comitê C-SVM/Linear. Entretanto, os resultados foram inferiores aos obtidos para o comitê C-SVM que utiliza o *kernel* KMOD (veja Tabela 5.7). Vale ressaltar que a utilização *kernel* KMOD nos classificadores-base aumenta significativamente a acurácia dos comitês.

Para concluir a seção de resultados, a Figura 5.5 apresenta os gráficos de desempenho para os comitês homogêneos e para os classificadores individuais. Pode-se verificar que os resultados obtidos para os comitês 5-MLP, 5-GRNN e 5-SVM são melhores que os resultados obtidos para os classificadores individuais, respectivamente. Novamente, para o problema estudado, o comitê 5-SVM aparece com destaque, pois esta estratégia de aprendizado detém os melhores resultados quando comparada a todas as demais, com exceção do comitê 1-MLP/4-SVM cujo desempenho é equivalente.

A partir dos resultados apresentados, as seguintes conclusões podem ser feitas: (i) dentre os classificadores tomados isoladamente, a rede MLP apresentou o melhor desempenho individual; (ii) o comitê homogêneo 5-SVM apresentou melhor capacidade de generalização em relação aos classificadores analisados individualmente; (iii) comitês heterogêneos só tiveram desempenho superior quando o número de classificadores SVM na composição era maior que o dos demais, e

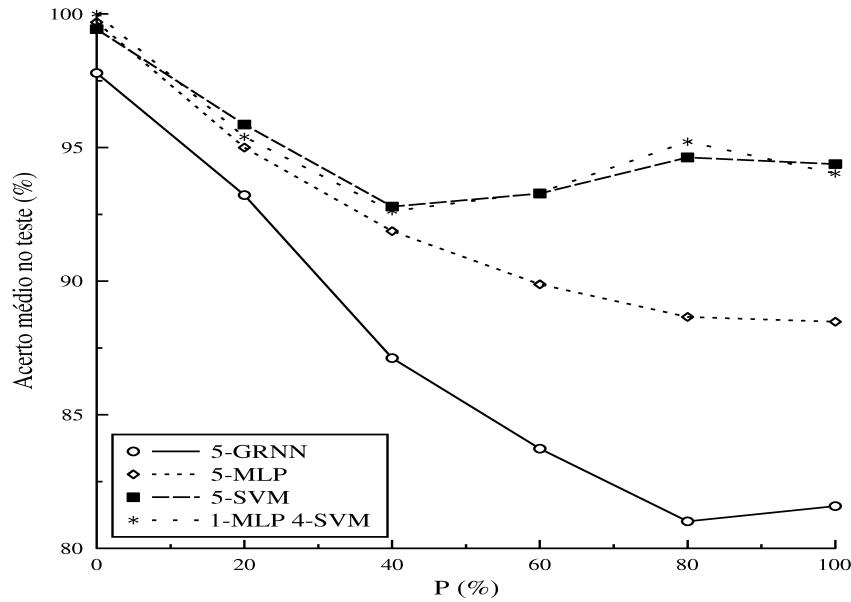


Figura 5.4: Gráfico do desempenho de classificação em função de P% para os comitês homogêneos e o melhor comitê heterogêneo, formado por $L = 5$ classificadores-base.

(iv) o comitê heterogêneo 1-MLP/4-SVM obteve resultados equivalentes ao comitê homogêneo 5-SVM, passando ambos a compor o módulo de diagnóstico semi-automático da plataforma SINPATCO.

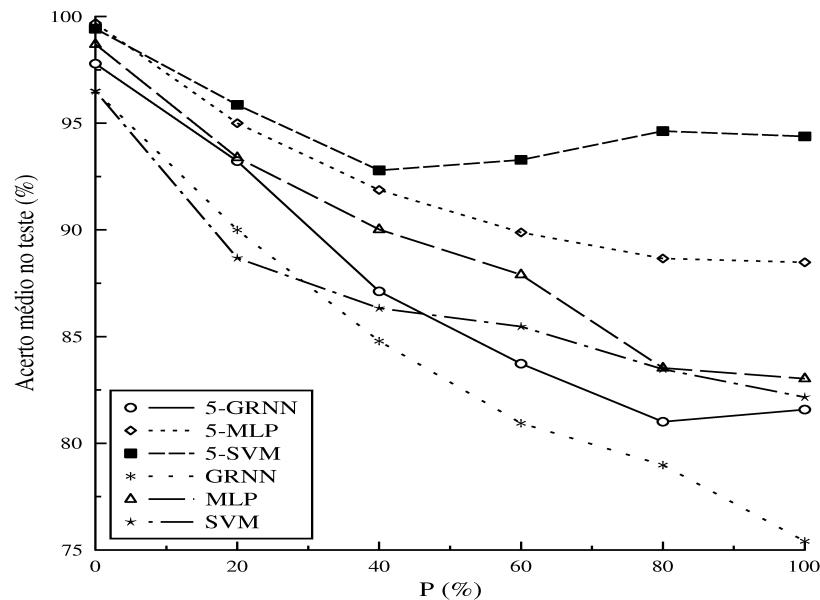


Figura 5.5: Gráfico do desempenho de classificação em função de P% para os comitês homogêneos e os classificadores individuais.

5.5 Conclusão

Este capítulo apresentou novos resultados referentes ao módulo de diagnóstico semi-automático da plataforma SINPATCO II. Este módulo é avaliado quanto à capacidade de categorizar casos clínicos na área de Ortopedia da coluna vertebral em uma das seguintes classes: Normal, Hérnia de Disco e Espondilolistese. Os classificadores MLP, SVM e GRNN tomados individualmente foram comparados com comitês formados a partir da combinação daqueles classificadores. De um modo geral, os comitês apresentaram melhor generalização que os classificadores individuais. Entre os comitês de classificadores, o comitê homogêneo composto de cinco classificadores SVM e o comitê heterogêneo com um classificador MLP e 4 classificadores SVM apresentou os melhores resultados no que se refere à taxa de acerto, número de falsos positivos e falsos negativos e robustez a *outliers*.

6 Conclusões e Trabalhos Futuros

Este último capítulo apresenta as conclusões, considerações finais, um resumo das contribuições científicas e identifica trabalhos futuros relacionados com os assuntos abordados nesta tese.

6.1 Conclusões

A plataforma SINPATCO é uma ferramenta de auxílio ao diagnóstico de patologias da coluna vertebral. O SINPATCO é composto por três módulos, a saber: interface gráfica, módulo de diagnóstico e de explanação de resultados.

Os classificadores que inicialmente pertenceram ao módulo de diagnóstico do SINPATCO, em sua primeira versão, foram *Naive Bayes*, k-NN, rede MLP, rede SOM e rede GRNN. O SINPATCO I é capaz de auxiliar no diagnóstico de patologias da coluna vertebral considerando o problema com 3 classes: hérnia de disco, espondilolistese e normal. O SINPATCO II, resultado deste trabalho, realiza a classificação considerando tanto o problema com 3 classes quanto com duas classes, normal e com patologia.

A partir dos resultados obtidos nesta tese de doutorado pode-se considerar a incorporação ao SINPATCO II de classificadores SVM (i.e., SVM/SMO/KMOD) e de comitês de classificadores. Vale destacar que o comitê 5-SVM apresenta melhor desempenho que os classificadores avaliados neste trabalho e melhor que os demais classificadores presentes no módulo de diagnóstico do SINPATCO I.

Com o intuito de reduzir o custo de uma classificação incorreta e aumentar a confiança e a qualidade no processo de tomada de decisão do SINPATCO I, pode-se adicionar também os classificadores que possuem capacidade de rejeição, como o rejoSVM e o SOM-2C. Estes classificadores apresentam acurácia muito próxima a 100% com taxas de rejeição relativamente baixas. Os casos rejeitados podem ser então separados para uma análise posterior de um especialista médico.

Para redução do custo computacional, foram propostos classificadores SVM e LSSVM com conjunto reduzido de vetores-suporte. Estes conjuntos reduzidos são obtidos com base na combinação dos classificadores SVM e LSSVM com o método *Opposite Maps* (ou *Generalized Opposite Maps*, sua versão estendida).

6.2 Resumo das Contribuições Científicas

De uma forma geral, são apresentadas neste trabalho contribuições relacionadas aos seguintes temas: máquinas de vetores-suporte, obtenção de conjuntos reduzidos em máquinas de vetores-suporte, classificação com opção de rejeição e em comitês de classificadores.

No Capítulo 2 deste trabalho é apresentado um estudo detalhado sobre classificadores SVM e LSSVM, tem-se também a proposição de um algoritmo de treinamento para os classificadores LSSVM com base no algoritmo de Levenberg-Marquardt (Proposta 1). Neste capítulo é mostrada ainda uma avaliação geral de desempenho de classificadores SVM e LSSVM para o problema PCV-2C e PCV-3C. Nesta avaliação são utilizados *kernels* linear, RBF e KMOD, e os algoritmos de treinamento SMO, *Kernel* Adatron para os classificadores SVM, e treinamento pela Matriz Inversa e por Levenberg-Marquardt para os classificadores LSSVM.

No Capítulo 3 é apresentado um novo método para obtenção de conjuntos reduzidos em classificadores SVM e LSSVM, denominado *Opposite Maps* (Proposta 2). Este método proposto baseia-se na rede de Kohonen e é generalizado para a obtenção de conjuntos reduzidos a partir de outros algoritmos de quantização vetorial (*Generalized Opposite Maps*). Com base no *Opposite Maps* são propostos diversos diversos classificadores, tais como GOM-SVM/KM e GOM-LSSVM/KM (derivados do K-Médias), SOM-SVM e SOM-LSSVM (derivados da rede SOM), GOM-SVM/GNG e GOM-LSSVM/GNG (derivados da rede GNG). Além disto, a obtenção de conjuntos reduzidos pelo método *Generalized Opposite Maps* foi estendida para trabalhar no espaço de características usando o algoritmo *Kernel* K-Médias. Estes classificadores são denotados GOM-SVM/K²M e GOM-LSSVM/K²M.

No Capítulo 4, é apresentada um análise comparativa de estratégias de classificação com opção de rejeição. Neste estudo são descritos resultados obtidos para estratégias um classificador padrão, dois classificadores independentes e classificador com rejeição embutida. São avaliados classificadores MLP e SVM com base nestas estratégias e, assim, tem-se os classificadores SVM-1C e MLP-1C (baseados na estratégia um classificador binário), SVM-2C, MLP-2C (baseados na estratégia dois classificadores independentes), são também avaliados os classificadores Fumera e rejoSVM (baseados na incorporação da capacidade de rejeição no

treinamento). Além disto, os classificadores SOM-1C (Proposta 3) e SOM-2C (Proposta 4) baseados na rede SOM são comparados com as demais estratégias. Os classificadores SOM-1C e SOM-2C apresentam-se como novidades visto que não há, até onde se tem conhecimento, técnicas para classificação com opção de rejeição que se baseiam em algoritmos de quantização vetorial. Ressalta-se que os classificadores SOM-1C e SOM-2C apresentaram melhores desempenhos que os classificadores MLP-1C, MLP-2C e SVM-2C e, mais especificamente, o classificador SOM-2C apresentou desempenho similar ao rejoSVM. Os classificadores rejoSVM e SOM-2C são superiores em desempenho aos demais classificadores com capacidade de rejeição avaliados.

Por último, no Capítulo 5, são avaliados comitês homogêneos e heterogêneos quando aplicados ao diagnóstico de patologias da coluna vertebral com 3 classes. Uma série abrangente de experimentos computacionais com vários classificadores clássicos, arranjados em comitês homogêneos/heterogêneos, e usando diferentes funções kernels (no caso do classificador SVM), atestam o desempenho superior do comitê 5-SVM (Proposta 5). Os desempenhos obtidos foram os que mais se destacam dentre todos os realizados neste trabalho, visto que a acurácia encontra-se bastante superior a 90%, quando não são consideradas rejeição ou filtragem de amostras discrepantes.

6.3 Trabalhos Futuros

Como trabalho futuro, pode-se citar a possibilidade de obtenção de conjuntos reduzidos com base em informações de erro obtidas no processo de resolução das LSSVMs com base no algoritmo de Levenberg-Maquardt. Há na literatura trabalhos que utilizam informação de erro em algoritmos de treinamento que utilizam informação de primeira ordem. Com a utilização do erro obtido pelo método de Levenberg-Maquardt há informação de segunda ordem.

Outra possibilidade de trabalho futuro consiste em aprimorar o método *Opposite Maps*. Este método pode ter seu conjunto reduzido diminuído por algumas alterações como a escolha dos padrões mais próximos aos protótipos ao invés de todos os padrões associados. Uma alteração importante pode levar em consideração a execução do algoritmo *Opposite Maps* de forma recursiva a fim de permitir utilizar mapas com quantidades de neurônios bastante inferiores.

Em termos de classificação com opção de rejeição, pode-se considerar a generalização do método SOM-1C e SOM-2C para outras algoritmos de quantização vetorial no espaço de entrada ou de características. Exemplos de algoritmos seriam K-Médias, *Kernel K-Médias*, a rede GNG, dentre outros.

APÊNDICE A – Análise Preliminar dos Dados

Neste apêndice são descritos outros resultados obtidos para o conjunto de dados de patologias da coluna vertebral. São apresentados os diagramas de caixa para os atributos biomecânicos, os gráficos de dispersão dos atributos tomados dois-a-dois, análise de componentes principais e análise da generalização com base nas diversas combinações possíveis dos atributos. Além disso, são apresentados resultados obtidos para classificação de patologias da coluna vertebral a partir de um conjunto de dados modificado, em que houve um aumento da quantidade de atributos com base em um mapeamento quadrático.

A.1 Diagramas de Caixa dos Atributos Biomecânicos

Nas figuras A.1-A.3 são apresentados os diagramas de caixa para cada uma das seis variáveis: incidência pélvica, versão pélvica, ângulo de lordose, declive sacral, raio pélvico e grau de deslizamento. Em cada diagrama de caixa são mostrados o menor valor, o 25º percentil , a mediana, o 75º percentil e o maior valor para cada uma das três classes: hérnia de disco, espondilolistese e normal.

Pode-se verificar, ao se analisar individualmente os atributos, que em geral há uma grande sobreposição entre os dados das classes. O atributo grau de deslizamento é o que se apresenta com maior capacidade de discriminação, visto que este apresenta os dados da classe espondilolistese menos sobrepostos aos dados das outras duas classes do que os dos outros atributos. Vale destacar que esta análise apresenta os dados em uma única dimensão, e que ao se considerar as diversas dimensões pode-se obter um problema não complexo. Porém, esta análise inicial pode permitir entender quais atributos possuem maior capacidade de discriminação ou mesmo mostrar que um determinado problema é simples, desde que se consiga observar uma baixa sobreposição dos dados.

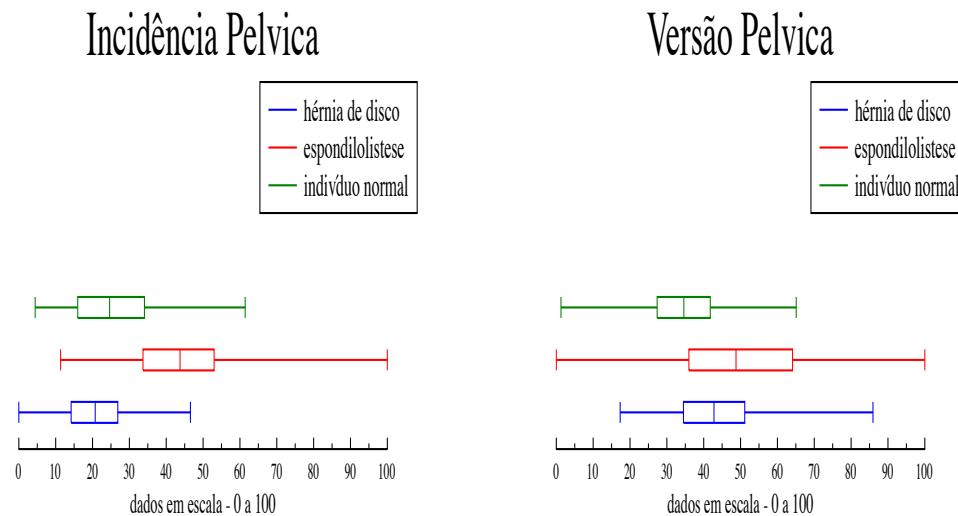


Figura A.1: (E) Diagrama de Caixa do atributo incidência pélvica. (D) Diagrama de caixa do atributo versão pélvica.

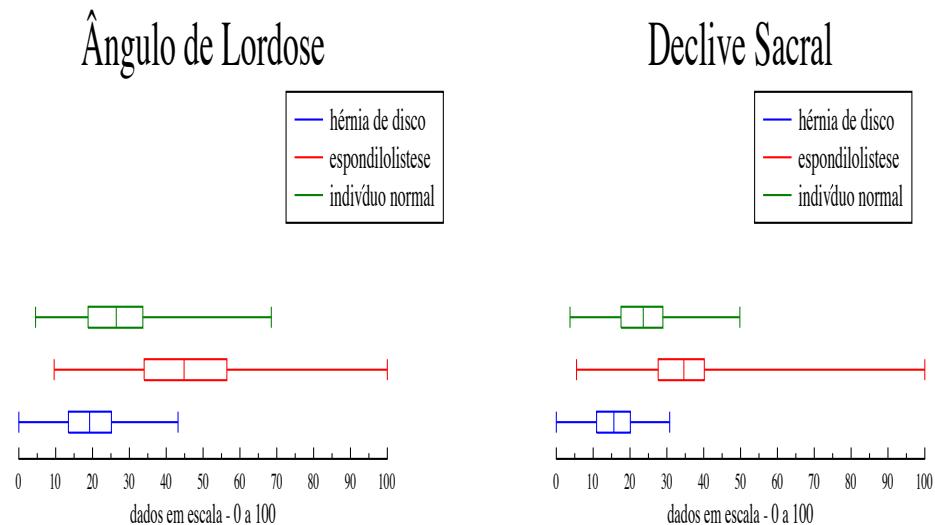


Figura A.2: (E) Diagrama de Caixa do atributo ângulo de lordose. (D) Diagrama de caixa do atributo declive sacral.

A.2 Gráficos de Dispersão e Curvas de Nível

Nas figuras A.4 e A.5 pode-se visualizar os melhores resultados das combinações de atributos tomados dois-a-dois. As figuras demonstram, ao se analisar bi-dimensionalmente a disposição dos dados, que há uma grande sobreposição entre as classes normal e hérnia de disco, enquanto os dados da classe espondilolistese apresentam-se menos sobrepostos aos dados das outras duas classes. A questão da dimensionalidade citada anteriormente também é válida.

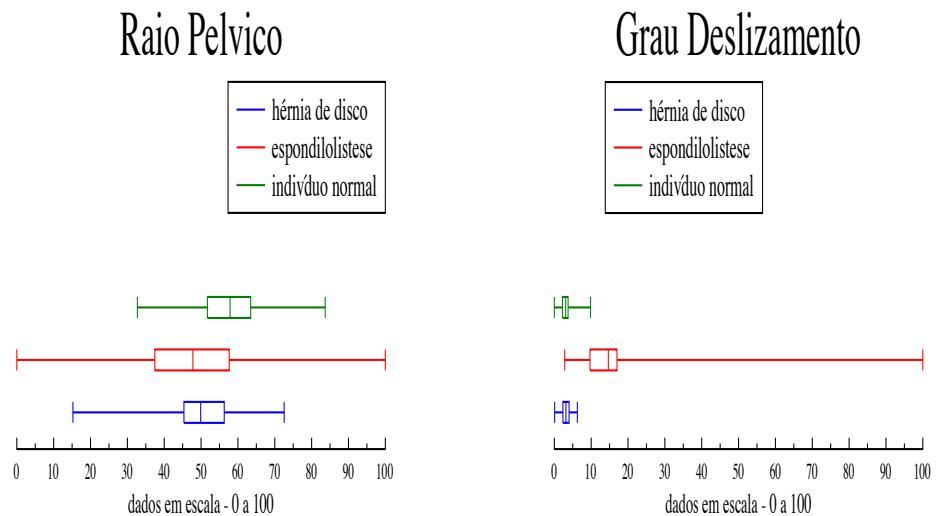


Figura A.3: (E) Diagrama de Caixa do atributo raio pélvico. (D) Diagrama de caixa do atributo grau de deslizamento.

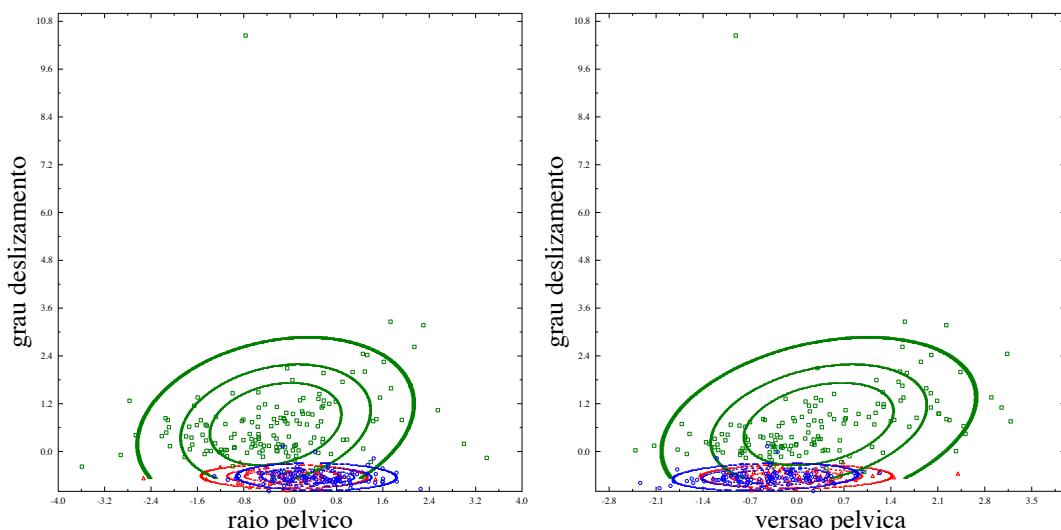


Figura A.4: (E) Gráficos de dispersão e curvas de nível dos atributos raio pélvico e grau de deslizamento. (D) Gráficos de dispersão e curvas de nível dos atributos versão pélvica e grau de deslizamento.

A.3 Análise de Componentes Principais (PCA)

Na Tabela A.1 são apresentados os autovetores referentes aos atributos: ângulo de incidência pélvica (IP), versão pélvica (VP), ângulo de lordose (AL), declive sacral (DS), raio pélvico (RP) e grau de deslizamento (GD). Os autovalores associados aos autovalores apresentados na Tabela A.1 estão descritos na Tabela A.2. Enquanto, o percentual de informação obtido a partir dos autovalores estão apresentados na Tabela A.3.

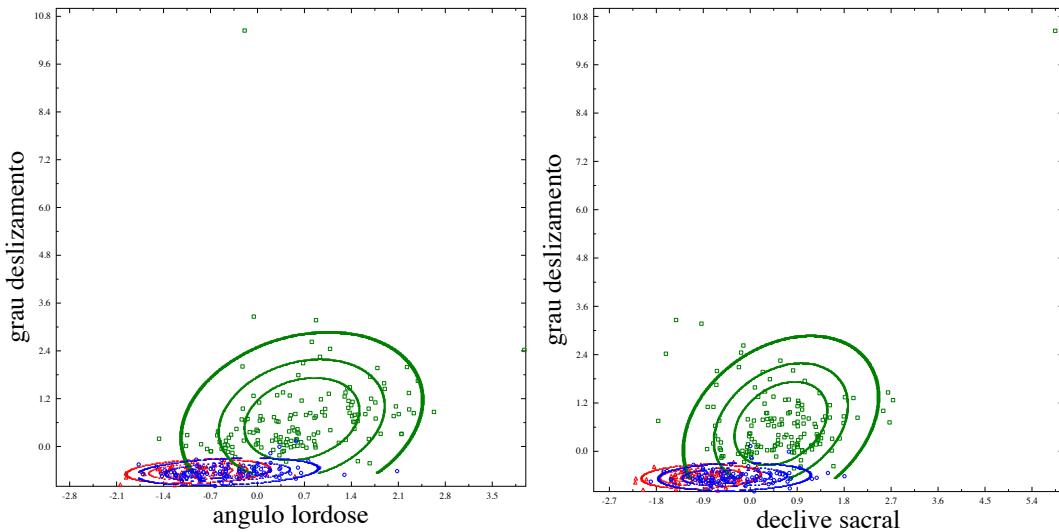


Figura A.5: (E) Gráficos de dispersão e curvas de nível dos atributos ângulo de lordose e grau de deslizamento. (D) Gráficos de dispersão e curvas de nível dos atributos declive sacral e grau de deslizamento.

	IP	VP	AL	DS	RP	GD
-0.717	-0.423	0.103	-0.096	0.002	0.535	
0.416	-0.151	0.006	-0.649	-0.528	0.324	
0.000	0.677	0.548	0.152	-0.093	0.458	
0.559	-0.432	0.127	0.360	0.396	0.446	
0.000	-0.276	0.174	0.586	-0.728	-0.143	
-0.000	0.279	-0.802	0.271	-0.163	0.424	

Tabela A.1: Autovetores para cada um dos atributos biomecânicos.

	IP	VP	AL	DS	RP	GD
0.000	0.326	0.473	0.761	1.195	3.246	

Tabela A.2: Autovalores associados a cada um dos autovetores apresentados anteriormente.

	IP	VP	AL	DS	RP	GD
0.0%	5.43%	7.88%	12.68%	19.91%	54.10%	

Tabela A.3: Percentual de informação contida em cada uma das componentes principais.

A.4 Análise de Generalização por Combinação dos Atributos

Nestas simulações utilizou-se uma rede MLP contendo $X - 12 - 3$ neurônios, em que X representa o número de atributos na combinação. Para cada valor de X são executadas 50 realizações. O conjunto de dados é separado em treinamento-teste com 80-20% do total de amostras, respectivamente. A rede é treinada utilizando uma taxa de aprendizado constante de

0,05 durante 1000 épocas. Os valores 1,2,3,4,5 e 6 apresentados nas tabelas a seguir correspondem aos atributos incidência pélvica, versão pélvica, ângulo de lordose, declive sacral, raio pélvico e grau de deslizamento, respectivamente. Assim, no campo atributo de cada tabela são apresentados os atributos utilizados na combinação. Os resultados são apresentados nas tabelas A.4-A.8.

Atributos	Média (%)	Desvio Padrão
1,2,3,4,5 e 6	84,45	4,18

Tabela A.4: Combinação dos atributos tomados seis-a-seis. Ou seja, utilizando todos os atributos.

Atributos	Média (%)	Desvio Padrão
1, 2, 3, 4 e 5	76,16	4,94
1, 2, 3, 4 e 6	82,80	3,69
1, 2, 3, 5 e 6	84,80	3,82
1, 2, 4, 5 e 6	84,45	3,89
1, 3, 4, 5 e 6	84,54	4,01
2, 3, 4, 5 e 6	84,55	4,24

Tabela A.5: Combinação dos atributos tomados cinco-a-cinco.

Pode-se verificar na Tabela A.5 que a acurácia diminui significativamente quando o atributo 6 (grau de deslizamento) é eliminado. Observa-se também que a acurácia apresenta-se um pouco superior ao resultado obtido quando se utilizam todos os atributos, bem como o desvio padrão apresenta-se menor.

Pode-se verificar na Tabela A.6 que as maiores taxas de acerto são obtidas para as combinações que envolvem o atributo grau de deslizamento (6). Pode-se verificar também que a acurácia em 86,03% para a combinação com os atributos 1,4,5 e 6 (incidência pélvica, declive sacral, raio pélvico e grau de deslizamento) é superior a de todos os outros resultados, seja nas combinações dos atributos tomados cinco-a-cinco ou mesmo com a utilização de todos as características.

Na Tabela A.7 pode-se verificar uma significativa degradação da generalização para os modelos gerados por bases de dados utilizando atributos combinados três-a-três. Pode-se verificar que a maior taxa de acerto é de 83,67%, um valor bastante inferior a 86,03% obtidos para os atributos: incidência pélvica, declive sacral, raio pélvico e grau de deslizamento. O mesmo pode ser dito para os resultados obtidos e apresentados na Tabela A.8.

Atributos	Média (%)	Desvio Padrão
1, 2, 3 e 4	68,03	5,10
1, 2, 3 e 5	75,77	5,06
1, 2, 3 e 6	81,77	4,48
1, 2, 4 e 5	74,67	5,53
1, 2, 4 e 6	81,64	4,75
1, 2, 5 e 6	85,67	4,54
1, 3, 4 e 5	75,45	5,58
1, 3, 4 e 6	81,74	4,70
1, 3, 5 e 6	82,41	4,95
1, 4, 5 e 6	86,03	3,97
2, 3, 4 e 5	75,58	5,11
2, 3, 4 e 6	82,35	5,09
2, 3, 5 e 6	84,58	3,92
2, 4, 5 e 6	85,12	3,85
3, 4, 5 e 6	85,51	3,97

Tabela A.6: Combinação dos atributos tomados quatro-a-quatro.

Atributos	Média (%)	Desvio Padrão
1, 2 e 3	69,80	4,45
1, 2 e 4	64,87	5,13
1, 2 e 5	73,80	4,95
1, 2 e 6	82,41	4,73
1, 3 e 4	66,67	5,04
1, 3 e 5	70,67	6,38
1, 3 e 6	79,51	5,23
1, 4 e 5	69,51	5,38
1, 4 e 6	81,93	4,49
1, 5 e 6	80,90	4,88
2, 3 e 4	67,74	5,86
2, 3 e 5	73,48	5,40
2, 3 e 6	81,38	4,16
2, 4 e 5	75,41	4,83
2, 4 e 6	82,61	4,06
2, 5 e 6	79,93	5,13
3, 4 e 5	71,74	4,82
3, 4 e 6	81,19	4,86
3, 5 e 6	82,41	4,58
4, 5 e 6	83,67	3,53

Tabela A.7: Combinação dos atributos tomados três-a-três.

Na Tabela A.9 é apresentado um resumo dos melhores resultados obtidos desde o uso de todos os atributos até as combinações de atributos tomados dois-a-dois. Com base nos resultados

Atributos	Média (%)	Desvio Padrão
1 e 2	64,03	5,87
1 e 3	62,54	4,81
1 e 4	64,54	5,44
1 e 5	65,32	6,21
1 e 6	77,25	4,89
2 e 3	64,70	5,57
2 e 4	62,70	5,05
2 e 5	62,54	5,14
2 e 6	79,19	4,75
3 e 4	63,64	5,03
3 e 5	69,70	5,44
3 e 6	79,87	4,49
4 e 5	67,64	6,51
4 e 6	81,16	4,25
5 e 6	78,80	4,89

Tabela A.8: Combinação dos atributos tomados dois-a-dois.

apresentados, pode-se inferir que os atributos 4,5,6 (declive sacral, raio pélvico e grau de deslizamento) são os mais importantes para fins de discriminação entre as classes. Visto que todos os melhores resultados apresentados envolvem estes três atributos, exceto quando se considera o melhor resultado da combinação de atributos tomados cinco-a-cinco. Porém, o segundo melhor resultado em que as acurárias são bastante próximas pode-se verificar também a presença destes três atributos.

Atributos	Média (%)	Desvio Padrão
4 e 6	81,16	4,25
4, 5 e 6	83,67	3,53
1, 4, 5 e 6	86,03	3,97
1, 3, 4, 5 e 6	84,54	4,01
1, 2, 3, 5 e 6	84,80	3,82
1, 3, 4, 5 e 6	84,54	4,01

Tabela A.9: Resumo dos resultados obtidos pela combinação de atributos.

A.5 Mapeamento Quadrático

Na Tabela A.10 são analisados também casos em que há um aumento na quantidade de atributos com base em um mapeamento quadrático. Nesta tabela podem ser observados resultados para diversas arquiteturas de redes neurais, visto que são analisadas redes com diversos

números de neurônios na camada oculta. A partir deste mapeamento uma nova base de dados pode ser obtida contendo no total 27 atributos. Neste contexto, são consideradas as seguintes combinações entre os atributos, a saber:

$$\begin{array}{ccccccc}
 x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \\
 x_1.x_1 & x_1.x_2 & x_1.x_3 & x_1.x_4 & x_1.x_5 & x_1.x_6 \\
 & x_2^2 & x_2.x_3 & x_2.x_4 & x_2.x_5 & x_2.x_6 \\
 & & x_3^2 & x_3.x_4 & x_3.x_5 & x_3.x_6 \\
 & & & x_4^2 & x_4.x_5 & x_4.x_6 \\
 & & & & & x_6^2
 \end{array}$$

MLP (IN-HL-IL)	Média (%)	Desvio Padrão
6-12-3	84,45	4,18
27-12-3	85,61	4,01
27-18-3	86,54	3,75
27-24-3	85,90	3,96
27-30-3	84,93	3,95
27-36-3	84,70	3,95
27-45-3	85,41	4,19
27-48-3	86,32	3,87
27-54-3	85,74	4,16
27-60-3	85,06	4,32

Tabela A.10: Resultados obtidos para diversas configurações da rede MLP.

APÊNDICE B – Rede GNG, Algoritmos K-Médias e Kernel K-Médias

B.1 K-Médias

Considere um conjunto de dados, consistindo de n amostras, dados por $D = x_1, x_2, \dots, x_n$. O algoritmo K-Médias (*K-Means*) partitiona o espaço de entrada com n amostras em K agrupamentos, $\{\mathbf{C}_j\}_{j=1}^K$, e então retorna o centróide cada agrupamento, $\{\mathbf{c}_j\}_{j=1}^K$. Este conjunto de centróides são os vetores que representam o conjunto de dados. Assim um conjunto de dados com n amostras pode ser compactado para este conjunto (*code book*) de k vetores. O algoritmo K-Médias em lote que utiliza a distância Euclidiana

$$d(\mathbf{x}_i, \mathbf{c}_j) = ||\mathbf{x}_i - \mathbf{c}_j||^2. \quad (\text{B.1})$$

é descrito como segue:

PASSO 1 - Atribuir um valor à K , em que representa a quantidade de centróides: $\{\mathbf{c}_j\}_{j=1}^K$.

Iniciar os K centróides com base em K amostras escolhidas aleatoriamente do conjunto de treinamento;

PASSO 2 - Associar cada amostra $\{\mathbf{x}_i\}_{i=1}^n$ ao centróide mais próximo para criar K agrupamentos. Isto é, calcule o valor da função indicadora $\{\delta(\mathbf{x}_i, \mathbf{C}_j)\}_{i=1, j=1}^{n, K}$, tal que

$$\delta(\mathbf{x}_i, \mathbf{C}_j) = \begin{cases} 1 & d(\mathbf{x}_i, \mathbf{c}_m) < d(\mathbf{x}_i, \mathbf{c}_j) \quad \forall j \neq m, \\ 0 & \text{caso contrário;} \end{cases} \quad (\text{B.2})$$

PASSO 3 - Atualizar o centróide \mathbf{c}_j para cada agrupamento \mathbf{C}_j

$$\mathbf{c}_j = \frac{1}{|\mathbf{C}_j|} \sum_{i=1}^n \delta(\mathbf{x}_i, \mathbf{C}_j) \mathbf{x}_i \quad (\text{B.3})$$

em que $|\mathbf{C}_k|$ é o número de vetores associados ao centróide \mathbf{C}_k .

PASSO 4 - Repetir os PASSOs 2 e 3 até a convergência.

PASSO 5 - Retornar os centróides $\{\mathbf{c}_j\}_{j=1}^K$.

O *K*-Médias é geralmente utilizado para descoberta de agrupamentos em dados. Porém, pode-se adaptar o algoritmo para possibilitar a sua utilização em problemas de classificações de padrões. Após o treinamento não-supervisionado, os padrões de treinamento são associados aos centróides mais próximos, de tal maneira que ao final deste processo estão associados k padrões a um determinado centróide, em seguida cada centróide recebe o rótulo da classe com mais ocorrências dentre os seus k padrões associados. Quando um exemplo não visto é apresentado, o *K*-Médias verifica qual o centróide mais próximo e então atribui a esta amostra o rótulo deste centróide.

B.2 Kernel K-Médias

O *Kernel K*-Médias (*Kernel K-Means* - K^2M) é uma extensão do algoritmo *K-Means*, tal que as operações até então realizadas no espaço de entrada são agora realizadas no espaço de características. Então considere cada centróide no espaço de características $\{\Phi(\mathbf{c}_j)\}_{j=1}^K$ como uma função de vetores de características em um espaço transformado, de tal forma que

$$\Phi(\mathbf{c}_j) = \frac{1}{|\mathbf{C}_j|} \sum_{k=1}^n \delta(\mathbf{x}_k, \mathbf{C}_j) \Phi(\mathbf{x}_k). \quad (\text{B.4})$$

A distância entre dois vetores característicos é expresso como

$$\begin{aligned} d^2(\Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j)) &= K(\Phi(\mathbf{x}_i), \Phi(\mathbf{x}_i)) - 2K(\Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j)) + K(\Phi(\mathbf{x}_j), \Phi(\mathbf{x}_j)) \\ d^2(\Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j)) &= K_{ii} - 2K_{ij} + K_{jj} \end{aligned} \quad (\text{B.5})$$

e a distância entre o vetor de características $\Phi(\mathbf{x}_i)$ e o centróide $\Phi(\mathbf{c}_j)$ no espaço de características pode ser expresso como

$$\begin{aligned} d^2(\Phi(\mathbf{x}_i), \Phi(\mathbf{c}_j)) &= ||\Phi(\mathbf{x}_i) - \Phi(\mathbf{c}_j)|| \\ &= ||\Phi(\mathbf{x}_i) - \frac{1}{|\mathbf{C}_j|} \sum_{k=1}^n \delta(\mathbf{x}_k, \mathbf{C}_j) \Phi(\mathbf{x}_k)|| \\ &= f(\mathbf{x}_i, \mathbf{x}_i) + g(\mathbf{x}_i, \mathbf{C}_j) + h(\mathbf{C}_j) \end{aligned} \quad (\text{B.6})$$

em que

$$f(\mathbf{x}_i, \mathbf{x}_i) = \Phi(\mathbf{x}_i)\Phi(\mathbf{x}_i) \quad (\text{B.7})$$

$$g(\mathbf{x}_i, \mathbf{C}_j) = -2 \frac{1}{|\mathbf{C}_j|} \sum_{k=1}^n \delta(\mathbf{x}_k, \mathbf{C}_j) \Phi(\mathbf{x}_k) \Phi(\mathbf{x}_i) \quad (\text{B.8})$$

$$h(\mathbf{C}_j) = \frac{1}{|\mathbf{C}_j|^2} \sum_{k=1}^n \sum_{m=1}^n \delta(\mathbf{x}_k, \mathbf{C}_j) \delta(\mathbf{x}_k, \mathbf{C}_j) \Phi(\mathbf{x}_k) \Phi(\mathbf{x}_k) \quad (\text{B.9})$$

(B.10)

PASSO 1 - Atribuir valores iniciais à $\{\delta(\mathbf{x}_i, \mathbf{C}_j)\}_{i=1, j=1}^{n, K}$, de tal maneira que são formados K agrupamentos iniciais $\{\mathbf{C}_j\}_{j=1}^K$;

PASSO 2 - Para cada agrupamento $\{\mathbf{C}_j\}_{j=1}^K$, calcule $|\mathbf{C}_j|$ e $h(\mathbf{C}_j)$;

PASSO 3 - Para cada vetor de entrada $\{\mathbf{x}_i\}_{i=1}^n$ e agrupamento $|\mathbf{C}_j|$, calcule $g(\mathbf{x}_i, \mathbf{C}_j)$ usando a Equação B.8. Depois disto, associar \mathbf{x}_i ao agrupamento mais próximo com base em

$$\delta(\mathbf{x}_i, \mathbf{C}_j) = \begin{cases} 1 & g(\mathbf{x}_i, \mathbf{C}_m) + h(\mathbf{C}_m) < g(\mathbf{x}_i, \mathbf{C}_j) + h(\mathbf{C}_j) \quad \forall j \neq m, \\ 0 & \text{caso contrário;} \end{cases} \quad (\text{B.11})$$

PASSO 4 - Repetir os PASSOs 2 e 3 até a convergência.

PASSO 5 - Para cada agrupamento $\{\mathbf{C}_j\}_{j=1}^K$, selecione como representante do centróide \mathbf{C}_j o vetor de características que está mais próximo ao centróide.

O processo de classificação de exemplos não vistos utilizando o *Kernel K-Means* é realizado de forma semelhante do *K-Means*.

B.3 Growing Neural Gas (GNG)

A rede GNG aumenta durante o processo de auto-organização (FRITZKE, 1995). O mecanismo de crescimento das GCSs (FRITZKE, 1994) e a geração de topologia do aprendizado competitivo hebbiano (MARTINETZ; SCHULTEN, 1991) são combinados neste novo modelo. Entre os pares de neurônios da rede há conexões que visam definir a estrutura topológica da rede. O processo de aprendizado tem seu início com poucos protótipos. Em seguida novas unidades são inseridas. Para determinar onde os novos protótipos devem ser adicionados, erros locais são coletadas durante o processo de treinamento/adaptação. Assim, cada novo protótipo é inserido próximo da unidade que possui o maior erro acumulado. O algoritmo de treinamento da rede GNG é descrito a seguir:

PASSO 1 - Iniciar com dois protótipos (unidades) a e b aleatoriamente nas posições \mathbf{w}_a e

\mathbf{w}_b em \mathbb{R}^n .

PASSO 2 - Selecionar de forma aleatória um padrão de entrada \mathbf{x} .

PASSO 3 - Encontrar a partir de \mathbf{x} os índices da primeira e da segunda unidade vencedora, $s1$ e $s2$, respectivamente.

PASSO 4 - Incrementar a idade de todas as conexões do protótipo com índice igual a $s1$.

PASSO 5 - Acumular o erro local do protótipo com índice igual a $s1$:

$$\Delta error(s1) = ||\mathbf{w}_{s1} - \mathbf{x}||^2 \quad (\text{B.12})$$

PASSO 6 - Mover o protótipo com índice igual a $s1$ e seu vizinhos topologicamente diretos na direção de \mathbf{x} proporcionalmente a ξ_s e ξ_n , respectivamente, segundo:

$$\Delta \mathbf{w}_{s1} = \xi_s (\mathbf{x} - \mathbf{w}_{s1}) \quad (\text{B.13})$$

$$\Delta \mathbf{w}_{sn} = \xi_n (\mathbf{x} - \mathbf{w}_n), \forall n \in N_{s1} \quad (\text{B.14})$$

em que N_{s1} é o conjunto de todos os protótipos conectados ao protótipo com índice igual a $s1$.

PASSO 7 - Se os protótipos com índices iguais a $s1$ e $s2$ estão conectados, então atribua o valor zero a esta conexão. Se tal conexão não existe, crie.

PASSO 8 - Remover conexões com idade maior que um determinado valor a_{max} . Se esta remoção resultar em neurônios não conectados, remova-os.

PASSO 9 - Se a quantidade de padrões de entrada apresentados até o momento for um múltiplo de um parâmetro λ , adicione um novo protótipo como descrito a seguir:

- Determinar o protótipo com o maior erro acumulado, tal que q representa o índice deste protótipo. Em seguida determine um outro protótipo o qual é vizinho ao protótipo com índice q e possui o maior erro acumulado (f representa o índice deste vizinho);
- Adicionar um novo protótipo na metade da distância entre os protótipos com índices q e f , tal que:

$$\mathbf{w}_r = 0,5(\mathbf{w}_q + \mathbf{w}_f). \quad (\text{B.15})$$

- Inserir conexões entre o protótipo com índice r e os protótipos com índices q e f , em seguida remover a conexão original entre os protótipos com índices q e f ;
- Decrementar a variável de erro dos protótipos com índices q e f multiplicando por uma constante α .

$$\Delta error(q) = -\alpha error(q) \quad (\text{B.16})$$

$$\Delta error(f) = -\alpha \ error(f) \quad (\text{B.17})$$

- Atribuir um valor ao erro do protótipo com índice r , a saber:

$$error(r) = (error(q) + error(f))/2; \quad (\text{B.18})$$

PASSO 10 - Decrementar o erro de todos os protótipos, multiplicando-os por uma constante β .

$$\Delta error(i) = -\beta \ error(i), \forall i \in N, \quad (\text{B.19})$$

em que N representa todo o conjunto de protótipos.

PASSO 11 - Se nenhum critério de parada (exemplo, tamanho da rede ou alguma medida de desempenho) for atingido retornar ao [PASSO 2].

O processo de classificação realizado por esta rede ocorre da mesma maneira que o processo de classificação do algoritmo K-Médias.

Referências Bibliográficas

- ALI, K. M.; PAZZANI, M. J. Error reduction through learning multiple descriptions. *Mach. Learn.*, Kluwer Academic Publishers, Hingham, MA, USA, v. 24, p. 173–202, September 1996.
- ANLAUF, J. K.; BIEHL, M. The adatron: An adaptive perceptron algorithm. *EPL (Europhysics Letters)*, v. 10, n. 7, p. 687, 1989.
- ANTANI, S. et al. Vertebra shape classification using MLP for content-based image retrieval. In: *Proceedings of the IEEE-INNS International Joint Conference on Neural Networks (IJCNN'03)*. [S.l.: s.n.], 2003. p. 160–165.
- AYAT, N. E.; CHERIET, M.; SUEN, C. Y. Kmod - a two parameter svm kernel for pattern recognition. In: *In Proceedings of the 16th International Conference on Pattern Recognition*. [S.l.: s.n.], 2002. v. 3, p. 331–334.
- BARTLETT, P.; SHawe-Taylor, J. Generalization performance of support vector machines and other pattern classifiers. In: _____. Cambridge, MA, USA: MIT Press, 1999. p. 43–54.
- BAUER, E.; KOHAVI, R. An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Mach. Learn.*, Kluwer Academic Publishers, Hingham, MA, USA, v. 36, p. 105–139, July 1999.
- BAZARAA, M.; SHERALI, H.; SHETTY, C. *Nonlinear Programming Theory and Algorithms*. [S.l.]: Wiley, 1992.
- BENNETT, K. P.; DEMIRIZ, A.; MACLIN, R. Exploiting unlabeled data in ensemble methods. In: *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: [s.n.], 2002. (KDD '02), p. 289–296. ISBN 1-58113-567-X.
- BERTHONNAUD, E. et al. Analysis of the sagittal balance of the spine and pelvis using shape and orientation parameters. *Journal of Spinal Disorders & Techniques*, v. 18, n. 1, p. 40–47, 2005.
- BI, C.; BECKER, M.; LEEDER, S. Derivation of minimum best sample size from microarray data sets: A monte carlo approach. In: *IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*. [S.l.: s.n.], 2011. p. 1 –6.
- BOSER, B. E.; GUYON, I.; VAPNIK, V. A training algorithm for optimal margin classifiers. In: *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*. [S.l.]: ACM Press, 1992. p. 144–152.

- BOUNDS, D. G.; LLOYD, P. J. A multi layer perceptron network for the diagnosis of low back pain. In: *Proceedings of the IEEE International Conference on Neural Networks* (San Diego, CA). [S.l.: s.n.], 1998. II, p. 481–489.
- BOUNSIAR, A.; BEAUSEROY, P.; GRALL-MAËS, E. General solution and learning method for binary classification with performance constraints. *Pattern Recognition Letters*, Elsevier, v. 29, p. 1455–1465, July 2008.
- BOUNSIAR, A.; GRALL, E.; BEAUSEROY, P. A kernel based rejection method for supervised classification. *International Journal of Computational Intelligence*, v. 3, n. 4, p. 312–321, 2007.
- BRAUSE, R. Revolutionieren neuronale netze unsere vorhersagefähigkeiten. *Zentralblatt für Chirurgie*, p. 692–698, 1999.
- BRAUSE, R. Medical analysis and diagnosis by neural networks. In: *Proceedings of the Second International Symposium on Medical Data Analysis*. [S.l.]: Springer-Verlag, 2001. (ISMDA '01), p. 1–13.
- BREIMAN, L. Stacked regressions. In: *Machine Learning*. [S.l.: s.n.], 1995. p. 49–64.
- BREIMAN, L. Bagging predictors. *Machine Learning*, v. 24, p. 123–140, 1996.
- BROWN, G. *Diversity in Neural Network Ensemble*. Tese (Doutorado) — University of Birmingham, 2004.
- BRUZZONE, L.; COSSU, R.; VERNAZZA, G. Detection of land-cover transitions by combining multilate classifiers. *Pattern Recognition Letters*, v. 25, n. 13, p. 1491–1500, 2004.
- BURBIDGE, R.; BUXTON, B. 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*. [S.l.: s.n.], 2001. p. 3–15.
- BURGES; CHRISTOPHER. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, v. 2, n. 2, p. 121–167, 1998.
- BURGES, C. J.; SCHÖLKOPF, B. Improving the accuracy and speed of support vector machines. In: *Advances in Neural Information Processing Systems 9*. [S.l.]: MIT Press, 1997. p. 375–381.
- BURGES, C. J. C. Simplified support vector decision rules. In: *Proc. 13th International Conference on Machine Learning*. [S.l.]: Morgan Kaufmann, 1996. p. 71–77.
- CAMPBEL, C.; CRISTIANINI, N. *Simple Learning Algorithms for Training Support Vector Machines*. [S.l.], 1998.
- CARDOSO, J. S.; COSTA, J. F. P. da. Learning to classify ordinal data: the data replication method. *Journal of Machine Learning Research*, v. 8, p. 1393–1429, 2007.
- CARVALHO, B. P. R.; BRAGA, A. P. *Novas Estratégias para Detecção Automática de Vetores de Suporte em Least Squares Support Vector Machines*. Dissertação (Mestrado) — Universidade Federal de Minas Gerais (UFMG), 2005.

- CARVALHO, B. P. R.; BRAGA, A. P. Ip-lssvm: A two-step sparse classifier. *Pattern Recogn. Lett.*, Elsevier Science Inc., v. 30, p. 1507–1515, 2009.
- CHANG, C.; LIN, C. *LIBSVM: a Library for Support Vector Machines*. 2001.
- CHERUKURI, M. et al. Anterior osteophyte discrimination in lumbar vertebrae using size-invariant features. *Computerized Medical Imaging and Graphics*, v. 28, n. 1-2, p. 99–108, 2004.
- CHINDARO, S.; SIRLANTZIS, K.; FAIRHURST, M. Modelling multiple-classifier relationships using bayesian belief networks. In: *Proceedings of the 7th international conference on Multiple classifier systems*. Berlin, Heidelberg: Springer-Verlag, 2007. (MCS'07), p. 312–321.
- CHOW, C. An optimum character recognition system using decision functions. *IRE Trans. Electronic Computers*, EC-B, p. 247–254, Dec 1957.
- CHOW, C. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*,, v. 16, n. 1, p. 41–46, jan 1970.
- CIOS, K. J.; KURGAN, L.; REFORMAT, M. Machine learning in the life sciences. *Engineering in Medicine and Biology Magazine, IEEE*, v. 26, n. 2, p. 14 –16, 2007.
- CORDELLA, L. et al. A method for improving classification reliability of multilayer perceptrons. *IEEE Trans. Neural Networks*, v. 6, p. 1140–1147, 1995.
- CORTES, C.; VAPNIK, V. Support vector networks. *MLearn*, v. 20, p. 273–297, 1995.
- CRAMMER, K.; SINGER, Y. On the algorithmic implementation of multiclass kernel-based vector machines. *IEEE Transactions on Latin America*, v. 2, n. 12, p. 265–292, Dec. 2001.
- DASARATHY, B.; SHEELA, B. Composite classifier system design: Concepts and methodology. In: *Proceedings of the IEEE*. [S.l.: s.n.], 1979. v. 67, n. 5, p. 708–713.
- DIETTERICH, G. B. T. Solving multiclass problem via error-correcting output code. *Journal of Artificial Intelligence Research*, v. 2, p. 263–286, 1995.
- DIETTERICH, T. G. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning*, v. 40, p. 139–157, August 2000.
- DIETTERICH, T. G. Ensemble learning. In: ARBIB, M. A. (Ed.). *The Handbook of Brain Theory and Neural Networks*. 2nd. ed. [S.l.]: MIT Press, 2002.
- DOBROWOLSKI, A.; WIERZBOWSKI, M.; TOMCZYKIEWICZ, K. Wavelet analysis for support vector machine classification of motor unit action potentials. In: *International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. [S.l.: s.n.], 2010. p. 4632 –4635.
- DOBROWOLSKI, A. P.; JAKUBOWSKI, J.; TOMCZYKIEWICZ, K. Linear discriminant analysis of muap scalograms. In: *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE*. [S.l.: s.n.], 2008. p. 1100 –1103.

- DOMINGOS, P. Proceedings of the aaai-96 workshop on integrating multiple learned models. In: *In Proceedings of the AAAI-96 Workshop on Integrating Multiple Learned Models*, AAAI Press. [S.l.: s.n.], 1996. p. 29–34.
- DOWNS, T.; GATES, K. E.; MASTERS, A. Exact simplification of support vector solutions. *Journal of Machine Learning Research*, v. 2, p. 293–297, 2002.
- DUAN, K.; KEERTHI, S. S. Which is the best multiclass svm method? an empirical study. In: *Proceedings of the Sixth International Workshop on Multiple Classifier Systems*. [S.l.: s.n.], 2005. p. 278–285.
- EFRON, B. *The Jackknife, the Bootstrap and Other Resampling Plans*. Philadelphia: Society for industrial and applied mathematics, 1982. 92 p. (Regional conference series in applied mathematics, v. 38).
- FIÉRE, V.; DA MOTA, H. Discal herniation pelvic incidence and spinopelvic balance: a correlation study. *European Spine Journal*, v. 1, p. 45, 2001.
- FLETCHER, R. *Practical Methods of Optimization*. second. New York: John Wiley & Sons, 1987.
- FRENCH, S. Group consensus probability distributions: A critical survey. In: BERNARDO, J. et al. (Ed.). *Bayesian Statistics 2*. [S.l.]: Elsevier Science Publishers B.V., 1985. p. 183–202.
- FREUND, Y.; SCHAPIRA, R. E. Experiments with a new boosting algorithm. In: *Proceedings of the Thirteenth International Conference inn Machine Learning*. [S.l.: s.n.], 1996. p. 325–332.
- FRIEDEL, C. C.; RUCKERT, U.; KRAMER, S. Cost curves for abstaining classifiers. In: *Proceedings of the ICML 2006 workshop on ROC Analysis in Machine Learning*. Pittsburgh, PA,: [s.n.], 2006. p. 33–40.
- FRIEß, T.; CRISTIANINI, N.; CAMPBELL, C. The kernel-adatron algorithm: a fast and simple learning procedure for support vector machines. In: *Machine Learning: Proceedings of the Fifteenth International Conference*. [S.l.]: Morgan Kaufmann Publishers, 1998.
- FRITZKE, B. Growing cell structures—a self-organizing network for unsupervised and supervised learning. *Neural Networks*, v. 7, p. 1441–1460, 1994.
- FRITZKE, B. A growing neural gas network learns topologies. In: *Advances in Neural Information Processing Systems 7*. Cambridge MA: MIT Press, 1995. p. 625–632.
- FUMERA, G. *Advanced Methods for Pattern Recognition with the Reject Option*. Tese (PhD Thesis) — Cagliari (Italy), 2002.
- FUMERA, G.; ROLI, F. Support vector machines with embedded reject option. In: *Proceedings of the Int. Workshop on Pattern Recognition with Support Vector Machines (SVM2002)*, Niagara Falls. [S.l.]: Springer, 2002. p. 68–82.
- FUMERA, G.; ROLI, F.; GIACINTO, G. Reject option with multiple thresholds. *Pattern Recognition*, v. 33, n. 12, p. 2099–2101, 2000.

- FUNG, G.; MANGASARIAN, O. L. Proximal support vector machine classifiers. *Knowledge Discovery and Data Mining*, p. 77–86, 2001.
- GAVRISHCHAKA, V. V.; KOEPKE, M. E.; ULYANOVA, O. N. Ensemble learning frameworks for the discovery of multi-component quantitative models in biomedical applications. In: *Second International Conference on Computer Modeling and Simulation, 2010. ICCMS '10*. [S.l.: s.n.], 2010. v. 4, p. 329 –336.
- GEY, S.; POGGI, J.-M. Boosting and instability for regression trees. *Computational Statistics & Data Analysis*, v. 50, n. 2, p. 533–550, 2006.
- GHORAI, S. et al. Multicategory cancer classification from gene expression data by multiclass nppc ensemble. In: *International Conference on Systems in Medicine and Biology (ICSMB)*. [S.l.: s.n.], 2010. p. 41 –48.
- GRIGSBY, J.; KOOKEN, R.; HERSHBERGER, J. Simulated neural networks to predict outcomes, costs, and length of stay among orthopedic rehabilitation patients. *Archives of Physical Medicine and Rehabilitation*, v. 75, n. 10, p. 1077–1081, 1994.
- GUIMERA-TOMAS, J. et al. Classifier of intestinal contractile activity degree based on internal electroenterogram recording. In: *International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. [S.l.: s.n.], 2010. p. 622 –625.
- HALL, S. J. *Biomecânica Básica*. [S.l.]: Guanabara Koogan, Rio de Janeiro, 2000.
- HAN, J. *Data Mining: Concepts and Techniques*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2005. ISBN 1558609016.
- HANCZAR, B.; DOUGHERTY, E. R. Classification with reject option in gene expression data. *Bioinformatics*, v. 24, n. 17, p. 1889–1895, 2008.
- HANSEN, L. K.; SALAMON, P. Neural network ensembles. *IEEE Trans. Pattern Anal. Mach. Intell.*, v. 12, p. 993–1001, October 1990.
- HO, T. K.; HULL, J. J.; SRIHARI, S. N. Decision combination in multiple classifier systems. *IEEE Trans. Pattern Anal. Mach. Intell.*, IEEE Computer Society, v. 16, p. 66–75, January 1994.
- HOEGAERTS, L. et al. A comparison of pruning algorithms for sparse least squares support vector machines. In: *Proceedings of the 11th International Conference on Neural Information Processing (11th ICONIP)*. Calcutta, India: [s.n.], 2004. p. 22–25.
- HUANG, Y.; SUEN, C. A method of combining multiple experts for the recognition of unconstrained handwritten numerals. *Pattern Analysis and Machine Intelligence*, v. 17, n. 1, p. 90–94, January 1995.
- HUSSAIN, A. et al. Reduced set support vector machines: Application for 2-dimensional datasets. In: *Proceedings of the Second International Conference on Signal Processing and Communication Systems (ICSPCS 2008)*. [S.l.: s.n.], 2008.
- JACOBS, R. A. Methods for combining experts' probability assessments. *Neural Computation*, v. 7, p. 867–888, 1995.

- KAUFMAN, L. Solving the quadratic programming problem arising in support vector classification. In: *Advances in kernel methods*. Cambridge, MA, USA: MIT Press, 1999. p. 147–167.
- KROGH, A.; VEDELSBY, J. Neural network ensembles, cross validation, and active learning. In: *Advances in Neural Information Processing Systems*. [S.l.]: MIT Press, 1995. p. 231–238.
- LABELLE, H.; ROUSSOULY, P.; BERTHONNAUD, E. The importance of spino pelvic balance in l5 s1 developmental spondylolisthesis. *SPINE*, v. 30, n. 6, p. 27–34, 2005.
- LAM, L. Classifier combinations: implementations and theoretical issues. in multiple classifier systems. *Lecture Notes in Computer Science, Cagliari, Italy, Springer*, v. 1857, p. 78–86, 2000.
- LEBLANC, M.; TIBSHIRANI, R. Combining estimates in regression and classification. *Journal of the American Statistical Association*, v. 91, n. 436, p. 1641, 1996.
- LECUN L. BOTOU, L. J. H. D. C. C. J. D. I. G. U. M. E. S. P. S. V. V. Y. Learning algorithms for classification: A comparison on handwritten digit recognition. In: *Neural Networks*. [S.l.]: World Scientific, 1995. p. 261–276.
- LEE, B. I. et al. Separation of factor images for blood flow estimation in positron emission tomography using ensemble independent component analysis. In: *Enterprise networking and Computing in Healthcare Industry, 2005. HEALTHCOM 2005. Proceedings of 7th International Workshop on*. [S.l.: s.n.], 2005. p. 460 – 462.
- LEE, J. et al. Arrhythmia classification with reduced features by linear discriminant analysis. In: *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*. [S.l.: s.n.], 2005. p. 1142 –1144.
- LEE, P. M. *Bayesian Statistics*. [S.l.]: Wiley, New York, 1997.
- LEE, Y.; MANGASARIAN, O. Rsvm: Reduced support vector machines. In: *SIAM International Conference on Data Mining*. [S.l.: s.n.], 2001. p. 00–07.
- LEE, Y.-J.; MANGASARIAN, O. L. RSVM: Reduced support vector machines. In: *Proceedings of the first SIAM International Conference on Data Mining*. [S.l.: s.n.], 2001.
- LEE, Y.-J.; MANGASARIAN, O. L. SSVM: A smooth support vector machine. *Computational Optimization and Applications*, v. 20, p. 5–22, 2001.
- LEVENBERG, K. A method for the solution of certain problems in least squares. *Quart. Appl. Math.*, v. 2, p. 164–168, 1944.
- LI, W. et al. Cox-2 activity prediction in chinese medicine using neural network based ensemble learning methods. In: *IEEE International Joint Conference on Neural Networks, (2008 IJCNN'2008). (IEEE World Congress on Computational Intelligence)*. [S.l.: s.n.], 2008. p. 1853 –1858.
- LI, Y.; LIN, C.; ZHANG, W. Letters: Improved sparse least-squares support vector machine classifiers. *Neurocomputing*, v. 69, p. 1655–1658, 2006.
- LINDA, O.; MANIC, M. Gng-svm framework - classifying large datasets with support vector machines using growing neural gas. In: *International Joint Conference on Neural Networks, 2009. (IJCNN 2009)*. [S.l.: s.n.], 2009. p. 1820 –1826.

- LIU, C. et al. Handwritten digit recognition: benchmarking of state-of-the-art techniques. In: *Pattern Recognition*. [S.l.: s.n.], 2003. v. 36, p. 2271–2285.
- LIU, C. lin; SAKO, H.; FUJISAWA, H. Performance evaluation of pattern classifiers for handwritten character recognition. *International Journal on Document Analysis and Recognition*, p. 191–204, 2002.
- MANGASARIAN, O. L. Generalized support vector machines. In: SMOLA P. BARTLETT, B. S. A. J.; SCHUURMANS, D. (Ed.). *Advances in Large Margin Classifiers*. [S.l.]: MIT Press, 2000. p. 135–146.
- MANGIAMELI, P.; WEST, D.; RAMPAL, R. Model selection for medical diagnosis decision support systems. *Decision Support Systems*, v. 36, p. 247–259, January 2004. ISSN 0167-9236.
- MARQUARDT, D. An algorithm for least-squares estimation of nonlinear parameters. *SIAM J. Appl. Math.*, v. 11, p. 431–441, 1963.
- MARTINETZ, T.; SCHULTEN, K. A neural gas network learns topologies. In: KOHONEN, T. et al. (Ed.). *Artificial Neural Networks*. Amsterdam: Elsevier, 1991. p. 397–402.
- MENAHEM, E. et al. Improving malware detection by applying multi-inducer ensemble. *Computational Statistics & Data Analysis*, v. 53, n. 4, p. 1483–1494, 2009.
- MERCER, J. Functions of positive and negative types and their connection with the theory of integral equations. In: *Transactions of the London Philosophical Society*. [S.l.: s.n.], 1909. (209, 415–446).
- MERKWIRTH, C. et al. Ensemble methods for classification in cheminformatics. *Journal of Chemical Information and Modeling*, v. 44, n. 6, p. 1971–1978, 2004.
- MITCHELL, T. *Machine Learning*. [S.l.]: McGraw-Hill, 1997.
- MOORE, E. H. On the reciprocal of the general algebraic matrix. *Bull. Amer. Math. Soc.*, v. 26, p. 394–395, 1920.
- MOSKOVITCH, R.; ELOVICI, Y.; ROKACH, L. Detection of unknown computer worms based on behavioral classification of the host. *Comput. Stat. Data Anal.*, v. 52, p. 4544–4566, May 2008. ISSN 0167-9473.
- NGUYEN, M. H. *Cooperative Coevolutionary Mixture of Experts - A Neuro Ensemble Approach for Automatic Decomposition Of Classification Problems*. Tese (Doutorado) — School of Information Technology and Electrical Engineering University of New South Wales, 2006.
- OHNO-MACHADO, L.; ROWLAND, T. Neural network applications in physical medicine and rehabilitation. *American Journal of Physical Medicine & Rehabilitation*, v. 78, n. 4, p. 392–398, 1999.
- OPITZ, D. W.; SHAVLIK, J. W.; SHAVLIK, O. Actively searching for an effective neural-network ensemble. *Connection Science*, v. 8, p. 337–353, 1996.
- PAPIK, K. et al. Application of neural networks in medicine - a review. *Medical Science Monitor*, v. 4, n. 3, p. 538–546, 1998.

- PENROSE, R. A generalized inverse for matrices. *Proceedings of the Cambridge Philosophical Society*, v. 51, p. 406–413, 1955.
- PERRONE, M. P.; COOPER, L. N. When networks disagree: Ensemble method for neural networks. In: *Artificial Neural Networks for Speech and Vision*. [S.l.: s.n.], 1993. p. 126–142.
- PLATT, J. C. *Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines*. 1998.
- PLATT, J. C. Fast training of support vector machines using sequential minimal optimization. In: *Advances in Kernel Methods - Support Vector Learning*. Cambridge, MA, USA: MIT Press, 1999.
- PLATT, J. C.; CRISTIANINI, N.; SHawe-Taylor, J. Large margin dags for multiclass classification. In: *Advances in Neural Information Processing Systems*. [S.l.]: MIT Press, 2000. p. 547–553.
- POLIKAR, R. Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, 2006.
- PRATI, R.; BATISTA, G.; MONARD, M. Evaluating classifiers using roc curves. *IEEE Latin America Transactions*, v. 6, n. 2, p. 215 –222, june 2008.
- QUEVEDO, J. et al. Disease liability prediction from large scale genotyping data using classifiers with a reject option. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, n. 99, p. 1, 2011.
- QUINLAN, J. Bagging, boosting, and c4.5. *Proceedings of the Thirteenth National Conference on Artificial Intelligence and the Eighth Innovative Applications of Artificial Intelligence Conference*, p. 725–730, 1996.
- RAHMAN, M.; ANTANI, S.; THOMA, G. A learning-based similarity fusion and filtering approach for biomedical image retrieval using svm classification and relevance feedback. *IEEE Transactions on Information Technology in Biomedicine*, v. 15, n. 4, p. 640 –646, july 2011.
- RAMESH, A. N. et al. Artificial intelligence in medicine. *Annals of the Royal Colege of Surgeons of England*, v. 86, n. 5, p. 334–338, 2004.
- RAMOSER, H. et al. Leukocyte segmentation and classification in blood-smear images. In: *27th Annual International Conference of the Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005*. [S.l.: s.n.], 2005. p. 3371 –3374.
- RANGANATHAN, A. *The Levenberg-Marquardt Algorithm*. 2004.
- REGGIA, J. A. Neural computation in medicine. *Artificial Intelligence in Medicine*, v. 5, n. 2, p. 143–157, 1993.
- RIDGEWAY, G. Looking for lumps: boosting and bagging for density estimation. *Computational Statistics & Data Analysis*, v. 38, n. 4, p. 379–392, Feb 2002.
- ROBERT, C. et al. Bibliometric overview of the utilization of artificial neural networks in medicine and biology. *Scientometrics*, v. 59, n. 1, p. 117–130, 2004.

- ROCHA-NETO, A. R. *SINPATCO - Sistema Inteligente para o Diagnóstico de Patologias da Coluna Vertebral*. Dissertação (Mestrado) — Universidade Federal do Ceará, 2006.
- ROCHA-NETO, A. R.; BARRETO, G. A. On the application of ensembles of classifiers to the diagnosis of pathologies of the vertebral column: A comparative analysis. *IEEE Transactions on Latin America*, v. 7, n. 4, p. 487–496, Aug. 2009. ISSN 1548-0992.
- ROCHA-NETO, A. R.; BARRETO, G. A. A Novel Heuristic for Building Reduced-Set SVMs using the Self-Organizing Map. In: Cabestany, J.; Rojas, I.; Caparrós, G. J. (Ed.). *International Conference on Artificial Neural Networks (IWANN'2011)*. [S.l.]: Springer, 2011. (Lecture Notes in Computer Science, v. 6691), p. 97–104.
- ROCHA-NETO, A. R. et al. Diagnostic of pathology on the vertebral column with embedded reject option. In: *Proceedings of Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA)*. [S.l.]: Springer-Verlag, 2011. (Lecture Notes in Computer Science. Berlin Heidelberg, v. 6669), p. 588–595.
- ROKACH, L. Mining manufacturing data using genetic algorithm-based feature set decomposition. *Intelligent Systems Technologies and Applications*, v. 4, n. 1/2, p. 57–78, 2008.
- ROKACH, L. Ensemble-based classifiers. *Artificial Intelligence Review*, v. 33, p. 1–39, February 2010.
- ROKACH, L.; MAIMON, O. Data mining for improving the quality of manufacturing: a feature set decomposition approach. *Journal of Intelligent Manufacturing*, v. 17, n. 3, p. 285–299, 2006.
- SCHAPIRE, R. The strength of weak learnability. *Machine Learning*, v. 5, n. 2, p. 197, 227 1990.
- SCHÖLKOPF, B. et al. Input space versus feature space in kernel-based methods. *IEEE Transactions on Neural Networks*, v. 10, n. 5, p. 1000–1017, 1999.
- SCHOLKOPF, B.; SMOLA, A. J. Learning with kernels: Support vector machines, regulation, optimization, and beyond. In: _____. Cambridge MA: MIT Press, 2002. cap. 18.
- SCHOLLHORN, W. I. Applications of artificial neural nets in clinical biomechanics. *Clinical Biomechanics*, v. 19, n. 9, p. 876–898, 2004.
- SCHÖLKOPF, B. et al. Fast approximation of support vector kernel expansions, and an interpretation of clustering as approximation in feature spaces. In: LEVI, P. et al. (Ed.). *Mustererkennung 1998–20. DAGM-Symposium*. Berlin: Springer, 1998. (Informatik aktuell), p. 124–132.
- SHARKEY, A. On combining artificial neural nets. *Connection Science*, v. 8, p. 299–313, 1996. Disponível em: <citeseer.nj.nec.com/sharkey96combining.html>.
- SIERMALA, M.; JUHOLA, M.; KENTALA, E. Neural network classification of otoneurological data and its visualization. *Computers in Biology and Medicine*, v. 38, p. 858–866, August 2008. ISSN 0010-4825.
- SMOLA, A.; SCHÖLKOPF, B. *A Tutorial on Support Vector Regression*. [S.l.], October 1998.

- SOUSA, R.; MORA, B.; CARDOSO, J. S. An ordinal data method for the classification with reject option. In: *Proceedings of The Eighth International Conference on Machine Learning and Applications (ICMLA 2009)*. [S.l.: s.n.], 2009.
- SOUSA, R.; MORA, B.; CARDOSO, J. S. An ordinal data method for the classification with reject option. In: *Proceedings of the 2009 International Conference on Machine Learning and Applications*. Washington, DC, USA: IEEE Computer Society, 2009. (ICMLA '09), p. 746–750.
- SPECHT, D. F. Probabilistic neural networks. *NNks*, v. 3, p. 109–118, 1990.
- SUYKENS, J. A. K.; LUKAS, L.; VANDEWALLE, J. Sparse approximation using least squares support vector machines. In: *Proceedings of 2000 IEEE International Symposium on Circuits and Systems*. Geneva, Switzerland: [s.n.], 2000. p. 757–760.
- SUYKENS, J. A. K.; LUKAS, L.; VANDEWALLE, J. Sparse least squares support vector machine classifiers. In: *In ESANN'2000 European Symposium on Artificial Neural Networks*. [S.l.: s.n.], 2000b. p. 37–42.
- SUYKENS, J. A. K.; VANDEWALLE, J. Least squares support vector machine classifiers. *Neural Processing Letters*, v. 9, n. 3, p. 293–300, 1999.
- SUYKENS, J. A. K.; VANDEWALLE, J. Least squares support vector machine classifiers. *Neural Processing Letters*, v. 9, n. 3, p. 293–300, 1999.
- SUYKENS, J. A. K.; VANDEWALLE, J. Multiclass least squares support vector machines. In: *IJCNN'99 International Joint Conference on Neural Networks*. Washington, DC: [s.n.], 1999.
- TAN, A. C.; GILBERT, D. Multi-class protein fold classification using a new ensemble machine learning approach. *Genome Informatics*, v. 14, p. 206–217, 2003.
- TANG, B.; MAZZONI, D. Multiclass reduced-set support vector machines. In: *Proceedings of the 23rd international conference on Machine learning (ICML 2006)*. New York, NY, USA: [s.n.], 2006. p. 921–928.
- TANIGUCHI, M.; TRESP, V. Averaging regularized estimators. *Neural Computation*, v. 9, n. 5, p. 1163–1178, 1997.
- TAO X. TANG, X. L. D.; WU, X. Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 28, n. 7, p. 1088–1099, Jul 2006.
- TSYMBAL, A. et al. Search strategies for ensemble feature selection in medical diagnostics. In: *Proceedings 16th IEEE Symposium Computer-Based Medical Systems, 2003*. [S.l.: s.n.], 2003. p. 124 – 129.
- TUKEY, J. *Exploratory data analysis*. [S.l.]: Addison-Wesley, 1977.
- TUMER, K.; GHOSH, J. Error correlation and error reduction in ensemble classifiers. *Connection Science*, v. 8, n. 3/4, p. 385–404, 1996.
- TUTZ, G.; BINDER, H. Boosting ridge regression. *Computational Statistics & Data Analysis*, v. 51, n. 12, p. 6044–6059, Aug 2007.

- TZALLAS, A. T.; TSIPOURAS, M. G.; FOTIADIS, D. I. Epileptic seizure detection in eegs using time-frequency analysis. *Trans. Info. Tech. Biomed.*, v. 13, p. 703–710, September 2009.
- UCI. *UCI Machine Learning Repository: Breast Cancer Wisconsin (Original)*. 2011. Disponível em: <[http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+\(Original\)](http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Original))>.
- UCI. *UCI Machine Learning Repository: Pima Indians Diabetes*. 2011. Disponível em: <<http://archive.ics.uci.edu/ml/datasets/Pima+Indians+Diabetes>>.
- UEDA, N.; NAKANO, R. Generalization error of ensemble estimators. In: *Proceedings of International Conference on Neural Networks*. [S.l.: s.n.], 1996. p. 90–95.
- VALIANT, L. G. A theory of the learnable. In: *Proceedings of the sixteenth annual ACM symposium on Theory of computing*. New York, NY, USA: [s.n.], 1984. (STOC '84), p. 436–445.
- VALYON, J. *Extended LS-SVM For System Modeling*. Tese (Doutorado) — Budapest University of Technology and Economics, 2007.
- VALYON, J.; HORVÁTH, G. A sparse least squares support vector machine classifier. In: *International Joint Conference on Neural Networks (IJCNN'2004)*. Hungary, Budapest.: [s.n.], 2004.
- VAPNIK, V. *Statistical learning theory*. [S.l.]: Wiley, 1998.
- VAPNIK, V. *The nature of statistical learning theory*. [S.l.]: Springer, 2000.
- VAPNIK, V.; LERNER, A. Pattern recognition using generalized portrait method. *Automation and Remote Control*, v. 24, 1963.
- VAPNIK, V. N. *Estimation of Dependences Based on Empirical Data*. [S.l.: s.n.], 1982. (Springer Series in Statistics).
- VAPNIK, V. N. *The Nature of Statistical Learning Theory*. New York: Springer-Verlag., 1995.
- VAPNIK, V. N.; CHERVONENKIS, A. Y. A note on one class of perceptrons. *Automation and Remote Control*, v. 25, 1964.
- VOLYANSKY, K. Y.; HADDAD, W. M.; BAILEY, J. M. Neuroadaptive output feedback control for nonlinear nonnegative dynamical systems with actuator amplitude and integral constraints. In: *Proceedings of the 2009 conference on American Control Conference*. [S.l.: s.n.], 2009. (ACC'09), p. 4494–4499. ISBN 978-1-4244-4523-3.
- WEBB, A. *Statistical Pattern Recognition*. 2nd. ed. [S.l.]: John Wiley & Sons, 2002.
- WERNECKE, K. A coupling procedure for the discrimination of mixed data. *Biometrics*, v. 48, n. 2, p. 497–506, 1992.
- WOLPERT, D. Stacked generalization. In: *Stacked generalization*. [S.l.: s.n.], 1992. p. 241–259.
- WU, X. et al. Top 10 algorithms in data mining. *Knowledge and Information Systems*, Springer-Verlag New York, Inc., New York, NY, USA, v. 14, p. 1–37, December 2007. ISSN 0219-1377.

- XIANG, C.; CHEN, M.; WANG, H. An ensemble method for medicine best selling prediction. In: *Sixth International Conference on Fuzzy Systems and Knowledge Discovery, 2009. (FSKD 2009)*. [S.l.: s.n.], 2009. v. 1, p. 100 –103.
- XU, K. *How Has the Literature on Gini's Index Evolved in the Past 80 Years?* [S.l.], 2004. Disponível em: <<http://econpapers.repec.org/RePEc:dal:wparch:howgini>>.
- XU, L.; KRZYZAKK, A.; SUEN, Y. C. Several methods for combining multiple classifiers and their applications in handwritten character recognition. *IEEE Transactions on System, Man and Cybernetics*, v. 22, n. 3, p. 418–435, 1992.
- Y.KIM; KOO, J.-Y. Inverse boosting for monotone regression functions. *Computational Statistics & Data Analysis*, v. 49, n. 3, p. 757–770, Jun 2005.
- ZHANG, J. Developing robust non-linear models through bootstrap aggregated neural networks. In: *Neurocomputing*. [S.l.: s.n.], 1999. p. 93–113.
- ZHANG, J. Inferential estimation of polymer quality using bootstrap aggregated neural networks. In: *Neural Networks*. [S.l.: s.n.], 1999. p. 927–938.
- ZHANG, R.; RUDNICKY, A. I. A large scale clustering scheme for kernel k-means. In: *Proceedings of the 16 th International Conference on Pattern Recognition (ICPR'02)*. Washington, DC, USA: IEEE Computer Society, 2002. (ICPR '02), p. 40289–40292.
- ZHOU, Z.-H.; JIANG, Y. Medical diagnosis with c4.5 rule preceded by artificial neural network ensemble. *IEEE Transactions on Information Technology in Biomedicine*, v. 7, n. 1, p. 37 –42, march 2003.