

**UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”  
INSTITUTO DE BIOCÊNCIAS LETRAS E CIÊNCIAS EXATAS DE S. J. DO RIO PRETO  
PROGRAMA DE PÓS GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO**

**Identificação e Caracterização de Pterígio  
Utilizando Floresta de Caminhos Ótimos e  
Técnicas de Otimização**

André Franco Pagnin

Orientador: Prof. Dr. João Paulo Papa  
Co-orientadora: Profa. Dra. Silvana Artioli Schellini

UNESP-São José do Rio Preto  
2011

**UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”  
INSTITUTO DE BIOCIÊNCIAS LETRAS E CIÊNCIAS EXATAS DE S. J. DO RIO PRETO  
PROGRAMA DE PÓS GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO**

**Identificação e Caracterização de Pterígio  
Utilizando Floresta de Caminhos Ótimos e  
Técnicas de Otimização**

André Franco Pagnin

Dissertação apresentada ao Instituto de Biociências,  
Letras e Ciências Exatas da UNESP de São José do  
Rio Preto como exigência para obtenção do título de  
Mestre em Ciência da Computação, área de concen-  
tração: Sistemas de Computação.

Orientador: Prof. Dr. João Paulo Papa  
Co-orientadora: Profa. Dra. Silvana Artioli Schellini

UNESP-São José do Rio Preto  
2011

Pagnin, André Franco.

Identificação e caracterização de pterígio utilizando floresta de caminhos ótimos e técnicas de otimização / André Franco Pagnin. - São José do Rio Preto : [s.n.], 2011.

70 f. : il. ; 30 cm.

Orientador: João Paulo Papa

Coorientador: Silvana Artioli Schellini

Dissertação (mestrado) - Universidade Estadual Paulista, Instituto de Biociências, Letras e Ciências Exatas

1. Computação. 2. Inteligência artificial. 3. Otimização de caminhos. 4. Pterígio. 5. Oftalmologia. I. Papa, João Paulo. II. Schellini, Silvana Artioli. III. Universidade Estadual Paulista, Instituto de Biociências, Letras e Ciências Exatas. IV. Título.

CDU – 004.8:617.7

Ficha catalográfica elaborada pela Biblioteca do IBILCE  
Campus de São José do Rio Preto - UNESP

André Franco Pagnin

**Identificação e Caracterização de Pterígio  
Utilizando Floresta de Caminhos Ótimos e  
Técnicas de Otimização**

Dissertação apresentada ao Instituto de Biociências, Letras e Ciências Exatas da UNESP de São José do Rio Preto como exigência para obtenção do título de Mestre em Ciência da Computação, área de concentração: Sistemas de Computação.

Banca Examinadora:

Prof. Dr. João Paulo Papa - DCo - FC/UNESP - Bauru

Prof. Dr. Alexandre Luis Magalhães Levada - DCo - UFSCar - São Carlos

Profa. Dra. Silke Anna Theresa Weber - DOOCCP - FM/UNESP - Botucatu

UNESP-São José do Rio Preto  
2011

*“À DEUS, ao Grupo Recogna e à todos que  
contribuíram direta ou indiretamente para a realização  
desse trabalho, em especial, meu orientador João Paulo,  
o meu muito obrigado!!!”*

*“À minha Família, e em  
especial, meu filho Diogo, que  
com seu sorriso puro e verdadeiro,  
me ensinou o verdadeiro  
sentido da vida.”*

*“Falta de tempo é desculpa daqueles que perdem  
tempo por falta de métodos.”  
(Albert Einstein)*

# Resumo

Neste trabalho abordamos o problema de identificação e caracterização do pterígio, uma neoformação conjuntival triangular ou trapezoidal benigna, com causa exata ainda não definida, que potencialmente pode acarretar cegueira, como um primeiro passo para a criação de um sistema especialista para auxílio ao diagnóstico utilizando a técnica de reconhecimento de padrões denominada Floresta de Caminhos Ótimos (OPF). Para a caracterização da doença, propomos a utilização de três técnicas de seleção de características as quais buscaram reduzir a quantidade de características utilizadas no reconhecimento, sem todavia, prejudicar a acurácia do classificador. Essas técnicas nasceram da junção do OPF com técnicas de otimização já existentes como Otimização por Enxame de Partículas (PSO), Busca Harmônica (BH) e Algoritmo de Busca Gravitacional (GSA), resultando nos algoritmos híbridos PSO-OPF, BH-OPF e GSA-OPF, respectivamente. O banco de dados utilizado neste trabalho é proveniente de um Projeto da Faculdade de Medicina da UNESP de Botucatu que visou a criação de uma Unidade Móvel para atendimento oftalmológico à comunidades da região de Botucatu. Esse banco de dados possui 89 características de 7,654 pacientes dos quais 682 são acometidos por pterígio e os 6,972 restantes não possuem a doença. As técnicas foram aplicadas à esse Banco de dados em dois momentos distintos. Inicialmente, buscando a identificação da doença, aplicamos o OPF juntamente com outros nove classificadores buscando somente a identificação do pterígio, dividindo a base de dados em 50 % para treinamento dos classificadores e os 50 % restantes para a classificação dos dados, num ciclo repetido 10 vezes, com esses conjuntos treinamento e classificação gerados aleatoriamente a cada ciclo. Os resultados foram bastante satisfatórios para a técnica OPF, já que a mesma obteve acurácia semelhante à maioria dos classificadores, porém com um custo computacional reduzido, atingindo uma acurácia média de 99.4148 % de acertos, com desvio padrão de 0.06788 %, num tempo médio de 0.5185 segundos para treinamento e 0.08063 segundos para a classificação. Já num segundo momento, quando aplicamos as técnicas propostas para a caracterização da doença, utilizamos 30% do conjunto de dados para o treinamento, 20% para compor o conjunto de avaliação e os 50% restantes para o conjunto de teste. Nos resultados obtidos, tivemos uma redução no número de características de 17.28 % na técnica PSO-OPF, 34.56 % no caso da técnica GSA-OPF e 58.02 % na técnica BH-OPF, mantendo a taxa de acerto similar à obtida na etapa anterior. Todas as técnicas obtiveram resultados muito próximos em termos de eficácia, sendo que o PSO-OPF selecionou 67 características em 9.52 segundos, GSA-OPF e o BH-OPF selecionaram 53 e 34 características em 19.02 e 9.62 segundos, respectivamente, demonstrando que a técnica BH-OPF é a melhor escolha para os propósitos do trabalho.

**Palavras-chave:** Floresta de Caminhos Ótimos, Inteligência Artificial, Otimização por Enxame de Partículas, Pterígio, Oftalmologia.



# Sumário

<b>Lista de Figuras</b>	<b>xi</b>
<b>Lista de Tabelas</b>	<b>xii</b>
<b>Lista de Quadros</b>	<b>xiii</b>
<b>Lista de Gráficos</b>	<b>xiv</b>
<b>Lista de Algoritmos</b>	<b>xv</b>
<b>1 Introdução</b>	<b>1</b>
<b>2 Pterígio</b>	<b>4</b>
2.1 Queixas . . . . .	5
2.2 Etiologia e Patogenia . . . . .	7
2.3 Evolução . . . . .	9
2.4 Tratamento . . . . .	10
<b>3 Técnicas de Classificação de Padrões</b>	<b>11</b>
3.1 Redes Neurais Artificiais . . . . .	11
3.1.1 Mapas de Auto-Organização . . . . .	12
3.1.2 Redes Neurais Perceptron Multicamadas . . . . .	14
3.1.2.1 Perceptron . . . . .	14
3.1.2.2 Perceptron Multicamadas . . . . .	17
3.1.2.3 Treinamento por Retropropagação . . . . .	18
3.2 Máquinas de Vetores de Suporte . . . . .	20
3.2.1 Classificadores por Hiperplano . . . . .	21
3.2.2 Nuclearização e Hiperplanos de Margem Suave . . . . .	24
3.2.3 Considerações Adicionais . . . . .	27
3.3 $k$ -Vizinhos Mais Próximos . . . . .	28
3.4 Floresta de Caminhos Ótimos . . . . .	30
3.4.1 Classificador por OPF . . . . .	30
3.4.2 Fundamentação Teórica . . . . .	31
3.4.3 Treinamento . . . . .	33
3.4.4 Classificação . . . . .	34

---

<b>4</b>	<b>Algoritmos de Otimização</b>	<b>35</b>
4.1	Otimização por Enxame de Partículas . . . . .	35
4.2	Busca Harmônica . . . . .	36
4.2.1	O Problema da Otimização e os Parâmetros do Algoritmo . . . . .	37
4.2.2	Memória Harmônica . . . . .	37
4.2.3	Gerando uma Nova Harmonia . . . . .	38
4.2.4	Atualizando a Memória Harmônica . . . . .	39
4.2.5	Critério de Parada . . . . .	39
4.3	Algoritmo de Busca Gravitacional . . . . .	39
<b>5</b>	<b>Seleção de Características</b>	<b>42</b>
5.1	Abordagem PSO-OPF Proposta . . . . .	43
5.2	Abordagem BH-OPF Proposta . . . . .	44
5.3	Abordagem GSA-OPF Proposta . . . . .	47
<b>6</b>	<b>Procedimentos Metodológicos</b>	<b>50</b>
6.1	Base de dados . . . . .	50
6.2	Identificação de pterígio . . . . .	51
6.3	Caracterização de pterígio . . . . .	54
<b>7</b>	<b>Conclusão</b>	<b>60</b>
	<b>Referências Bibliográficas</b>	<b>63</b>

# Lista de Figuras

1	Exemplar de (a) olho livre de pterígio e (b) olho com pterígio. . . . .	4
2	Exemplar de olho com pterígio do tipo I (a), tipo II (b) e tipo III (c). . . . .	6
3	Média Anual de Irradiação Diária de UV(1990-2004) [44] . . . . .	8
4	Arquitetura bidimensional típica de uma rede de Kohonen adaptada de [72]. . . . .	15
5	Funções de Ativação Linear (a), Sigmoidal (b) e Identidade (c). . . . .	16
6	Modelo simplificado do Perceptron adaptado de [72]. . . . .	17
7	Representação básica de uma rede <i>feedforward</i> (multicamadas) adaptada de [72]. . .	19
8	Exemplo de hiperplano ótimo. . . . .	22
9	Mapeamento dos dados para um espaço de maior dimensionalidade. . . . .	25
10	Processo de classificação do $k$ -NN Adaptado de [81]. . . . .	29
11	Exemplo de Funcionamento do OPF . . . . .	32
12	Interpretação geométrica do PSO. . . . .	36

# Lista de Tabelas

1	Resultados para a identificação automática de pterígio. . . . .	54
2	Resultados para a caracterização de pterígio. . . . .	56
3	Tempo de execução em segundos para a etapa de seleção de características. . . . .	56

# Lista de Quadros

1	Características Seleccionadas por cada Algoritmo Parte 1/2. . . . .	57
2	Características Seleccionadas por cada Algoritmo Parte 2/2. . . . .	58

# Lista de Gráficos

1	Acurácia média. . . . .	53
2	Tempo médio de treinamento (escala logarítmica). . . . .	53
3	Tempo médio de classificação. . . . .	55
4	Gráfico da taxa de reconhecimento média com relação à variações no valor do limiar $\Phi$ . . . . .	55

# Lista de Algoritmos

1	C LASSIFICADOR k-NN .....	29
2	Classificador Supervisionado baseado em Floresta de Caminhos Ótimos .....	33
3	Algoritmo PSO-OPF .....	45
4	Algoritmo BH-OPF .....	46
5	Algoritmo GSA-OPF .....	48

# Capítulo 1

## Introdução

A visão é o principal sentido do ser humano, sendo responsável por 85% de toda a informação que recebemos e processamos do ambiente externo, contra 10% da audição e 5% dos demais sentidos [1]. No entanto, uma grande quantidade de pessoas são acometidas por doenças que podem causar desde pequenos transtornos, até danos irreparáveis para a visão. De acordo com a Organização Mundial de Saúde, existem atualmente 314 milhões de deficientes visuais, sendo que 45 milhões de pessoas são classificadas como cegas. De forma geral, cerca de 85% de toda a deficiência visual e 75% de cegueira poderiam ser evitadas ou curadas em todo o mundo, tanto por métodos de prevenção, quanto por tratamento específico [2]. Dentre essas doenças, podemos citar o pterígio, uma neoformação conjuntival triangular ou trapezoidal benigna, com causa exata ainda não definida, que potencialmente pode acarretar cegueira, já que seu crescimento pode obstruir a pupila, impedindo a visão [3, 4]. Esse nível mais grave, que causa a cegueira, ocorre em uma média de 1,4% dos pacientes diagnosticados com pterígio [5].

Nesse ponto, os avanços médicos e tecnológicos proporcionam diagnósticos cada vez mais precoces e tratamentos mais eficientes. Tais avanços tem causado várias mudanças, inclusive na forma de se observar as ciências médicas, evidenciando que, num futuro próximo, o fato de conhecer como buscar a melhor e mais rápida informação será mais importante do que tentar armazenar a própria informação. Neste contexto, uma nova ciência chamada Informática Médica, a qual integra as ciências da informação e da saúde, tem sido amplamente utilizada. Sistemas de suporte a decisão, os quais são usados para fins medicinais desde 1970, tem como principal ferramenta o poder de incorporar e associar uma grande quantidade de dados médicos. Assim, tais sistemas podem ajudar os profissionais da Saúde a tomar decisões sobre o diagnóstico do paciente. Além disso, teorias sobre o mecanismo do cérebro humano e outras técnicas baseadas em inteligência artificial tem atraído muita atenção, com alguns estudos orientados para o campo da pesquisa da oftalmologia [6, 7].

Lin *et al.* [6], por exemplo, aplicaram uma rede neural de três camadas para detectar a progres-



são do campo visual do glaucoma em 80 pacientes. A rede neural foi treinada com o algoritmo de retropropagação [8], a fim de realizar essa tarefa. Com o objetivo de identificar as fases do glaucoma, Galilea *et al.* [9] também aplicaram redes neurais artificiais em 106 olhos de 53 pacientes, obtendo 100% de exatidão. Huang *et al.* [7] propuseram um sistema neuro-nebuloso para diferenciar olhos normais e glaucomatosos. Como se pode notar, um grande esforço tem sido dedicado ao estudo de doenças oftalmológicas e sua detecção automática por meio de técnicas de aprendizado de máquina.

Apesar destes vários estudos utilizando redes neurais na identificação do glaucoma, não se tem informações sobre a aplicação de técnicas de inteligência artificial em outras doenças oculares, tais como o pterígio. Assim, tendo em vista que atualmente muitas doenças oftalmológicas não estão devidamente tratadas na literatura das ciências exatas, este trabalho tem como objetivo apresentar uma comparação detalhada entre algumas técnicas de reconhecimento de padrões para a identificação de pacientes acometidos com pterígio. Outro fator preponderante e não menos importante do que a sua identificação, corresponde à caracterização dessa enfermidade, ou seja, estabelecer quais as características mais relevantes para a identificação do pterígio é tão importante quanto o seu reconhecimento por meio de técnicas automáticas.

Recentemente, uma técnica de classificação de padrões denominada Floresta de Caminhos Ótimos (*Optimum-Path Forest* - OPF) foi proposta na literatura [10], a qual tem se mostrado eficiente, com excelentes resultados em comparação com técnicas tradicionais de reconhecimento de padrões, tais como redes neurais e máquinas de vetores de suporte (*Support Vector Machines* - SVM), mas com menor custo computacional para a etapa de treinamento dos dados.

Dessa forma, este trabalho teve como objetivo utilizar o classificador OPF para a identificação automática de pacientes acometidos pelo pterígio por meio de dados clínicos. Outra contribuição do presente trabalho é a proposta de um novo algoritmo de seleção de características baseado em Otimização por Enxame de Partículas e OPF para a caracterização do pterígio, ou seja, para o estabelecimento das características clínicas mais importantes e relevantes na tarefa de identificação dessa enfermidade.

Essa dissertação está dividida em capítulos dispostos da seguinte forma: No Capítulo 2 detalha-se o pterígio juntamente com as queixas, etiologia, patogenia, evolução e tratamento dessa importante doença com grande incidência na população brasileira e que desperta grandes controvérsias na literatura médica quando se diz respeito à sua caracterização. Já no Capítulo 3 é definido o que é Reconhecimento de padrões e é descrito as principais técnicas existentes, tais como Redes Neurais Artificiais, Máquinas de Vetores de Suporte, K-vizinhos mais Proximos e Floresta de Caminhos Ótimos(OPF). No Capítulo 4 é feito um delineamento dos algoritmos de otimização, dando enfoque para as técnicas de Otimização por Enxame de Partículas (PSO), Busca Harmônica (BH) e Algoritmo de Busca Gravitacional (GSA). No Capítulo 5 é frisado a importância da seleção de características e são

---

propostas três técnicas híbridas entre os algoritmos de otimização citados no Capítulo 3 e a técnica de reconhecimento de padrões OPF, dando origem aos algoritmos PSO-OPF, BH-OPF e GSA-OPF. No Capítulo 6 são frisados os Procedimentos Metodológicos, detalhando sobre as características da base de dados utilizada no trabalho, a aplicação das técnicas de reconhecimento de padrões para identificar o pterígio e a utilização das técnicas propostas para caracterização da doença, demonstrando e discutindo os resultados. Já no Capítulo 7, conclui-se o trabalho comentando a relevância do mesmo, propondo os trabalhos futuros e relacionando as publicações originadas dessa dissertação.

## Capítulo 2

### Pterígio

Neste capítulo será abordado o pterígio, uma neoformação comum na superfície ocular com possibilidade de causar cegueira e astigmatismo. Serão detalhadas as queixas mais comuns dos pacientes acometidos por essa enfermidade relatando a etiologia e a patogenia da doença as quais apresentam diversas inconsistências na literatura médica. Concluindo o capítulo será descrito a evolução da doença, seu tratamento e os riscos de reincidência.

O termo pterígio vem do grego e significa “pequena asa”, tendo em vista sua semelhança com uma asa de inseto [3]. Caracteriza-se como uma neoformação triangular ou trapezoidal que ocorre na superfície ocular; habitualmente se dispõe ao longo do eixo horizontal da fenda interpalpebral e, com maior frequência, na região do limbo medial, podendo ocorrer também no limbo temporal, formando uma cunha de tecido fibrovascular que avança superficialmente sobre a córnea [11, 12]. As Figuras 1.a e 1.b mostram um exemplo de olho sem pterígio e olho com pterígio, respectivamente.

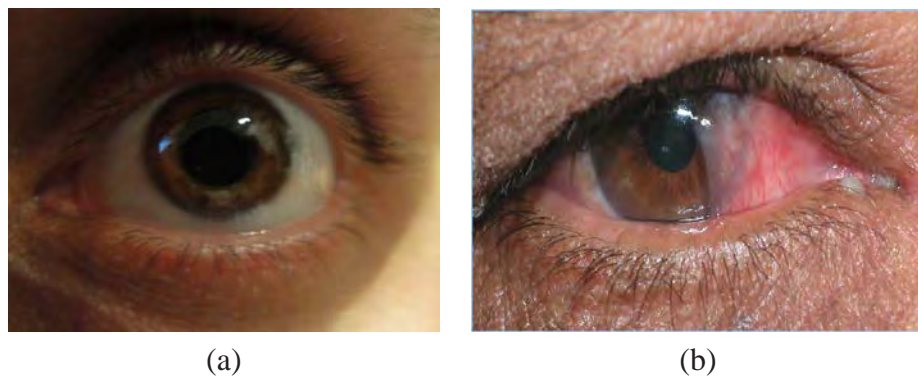


Figura 1: Exemplo de (a) olho livre de pterígio e (b) olho com pterígio.

O pterígio pode ser dividido em três partes: cabeça (parte corneana), corpo (parte conjuntival) e o pescoço que une as duas primeiras. Na cabeça distingue-se uma orla avascular, semitransparente e gelatinosa: a zona pelúcida e, proximamente a ela existe outra faixa, não transparente, esbranquiçada

e vascularizada, a zona opaca do pterígio. Distalmente à zona pelúcida, avascular, observam-se as ilhotas de Fuchs. Contornando-se a cabeça, nos pterígios estacionários, observa-se a linha de Stocker que corresponde ao depósito epitelial de pigmento hemossiderínico. O corpo corresponde à porção mais vascularizada, espessa e proximal do pterígio e que pode atingir a carúncula e prega semilunar. A vascularização normal da conjuntiva está substancialmente modificada no corpo do pterígio e salienta a impressão visual de tração da conjuntiva em direção à área pupilar da córnea [11, 12].

O pterígio pode ser classificado em três tipos [12, 13]:

- O pterígio tipo I apresenta o corpo bem definido e a cabeça avança sobre a córnea menos do que 2mm (Figura 2.a);
- O pterígio tipo II estende-se sobre a córnea por cerca de 2 a 4mm podendo induzir ao astigmatismo e redução de acuidade visual (Figura 2.b) e
- O pterígio tipo III avança sobre a córnea por mais de 4mm, entrando na zona óptica e causando redução de acuidade visual. Geralmente, está associado à extensa fibrose subconjuntival, algumas vezes levando a limitação de motilidade ocular extrínseca e diplopia, ou seja, a percepção de duas imagens a partir de um único objeto (Figura 2.c).

A lesão pode ter dois padrões de aparência: um aspecto involutivo quando os vasos não são tão exuberantes e um aspecto carnosos, quando não é possível observar a estrutura da íris sob a lesão por causa da proliferação densa e opaca do tecido conjuntival, com um número elevado de novos vasos.

A detecção do pterígio pode ser feita com facilidade, sendo possível o seu diagnóstico por meio de exame clínico, sem equipamentos sofisticados. Observa-se que o pterígio corresponde a proliferação de tecido conjuntival e novos vasos formando um tecido que cresce em direção da córnea, mais frequentemente localizado no lado nasal.

No entanto, apesar da fácil detecção clínica do pterígio, as queixas e outros fatores associados não são específicas, gerando uma grande dificuldade para construir um perfil dos indivíduos que teriam probabilidade de serem afetados por esta doença.

## 2.1 Queixas

Portadores de pterígio queixam-se mais frequentemente de olho vermelho, ardor, queimação, irritação, fotofobia e sensação de corpo estranho nos olhos. Estas queixas são consequência da quebra do filme lacrimal provocada pela irregularidade superficial da conjuntiva em decorrência da presença da lesão na superfície ocular, criando áreas “secas” [12, 14, 15].

Entretanto, muitas vezes as queixas que levam o paciente a procurar tratamento estão relacionadas com a estética, uma vez que o avanço do pterígio sobre a córnea e os vasos alargados podem dar ao

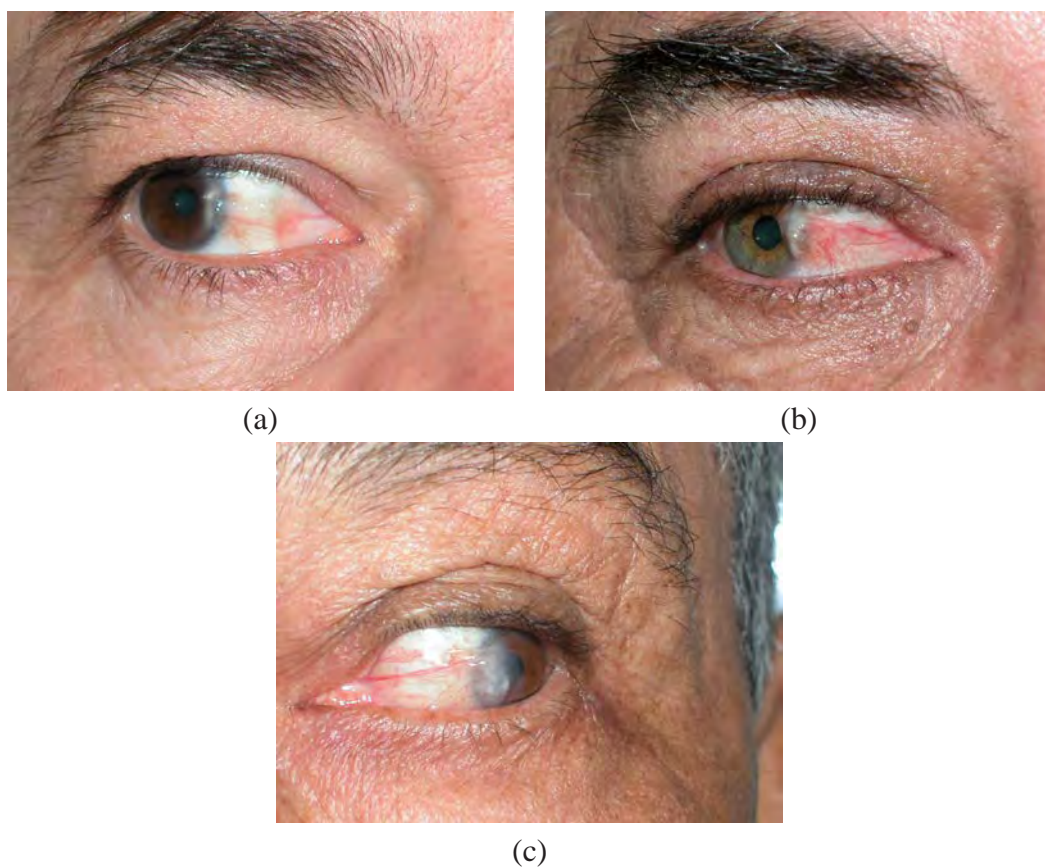


Figura 2: Exemplar de olho com pterígio do tipo I (a), tipo II (b) e tipo III (c).

paciente o aspecto de olho vermelho confundindo, muitas vezes, com o uso ou o abuso de álcool e drogas. Além disso, a visão pode ser afetada, principalmente quando a lesão se estende sobre a córnea e eixo visual, impedindo a transmissão dos estímulos visuais. A baixa visão pode resultar também do astigmatismo induzido pela lesão. Portanto, as reclamações não são específicas do pterígio. As queixas são mais importantes nos pterígios carnosos, devido à elevação da lesão e a maior quantidade de vasos dilatados [5, 16].

## 2.2 Etiologia e Patogenia

A ocorrência de pterígio é grande, e apesar de sua ocorrência comum, existem muitas controvérsias em relação a etiologia, patogênese e tratamento do pterígio [17, 18] e que muitos fatores ambientais, como a radiação ultravioleta - UV e a irritação crônica têm sido postulados como fatores causais. O pterígio afeta adultos e é muito raro em crianças, e sua prevalência aumenta linearmente com a idade [19, 20]. Afeta preferencialmente indivíduos adultos, do sexo masculino, acima dos 30 anos [19, 14]. Entretanto, pesquisas brasileiras mostraram acometimento de ambos os sexos, nas mesmas proporções [21], ou ainda predomínio no sexo feminino [5, 15]. Pesquisa feita na China [22], também reportou predomínio da lesão em mulheres. Este fato pode estar muito mais relacionado à procura do tratamento por estética, do que por uma tendência de ocorrência da lesão relacionada ao sexo. Esta observação está fortemente consubstanciada pelas queixas dos indivíduos, já que a estética foi a queixa predominante em algumas amostras estudadas [5].

Em outro levantamento, feito no Peru, que levou em conta uma amostra colhida em quatro hospitais daquele país, observou-se que não houve predomínio estatisticamente significativo quanto ao sexo ou ocupação [23]. Porém, por exercerem os indivíduos masculinos mais atividades laborativas com exposição solar, estariam eles mais predispostos ao desenvolvimento da lesão [24].

Estudos feitos em outros países, como Austrália e Estados Unidos, demonstraram forte correlação positiva entre a exposição à radiação ultravioleta e o desenvolvimento do pterígio [25, 26]. Numa população do Saara, foi verificado que a gravidade e a duração da exposição ao sol são importantes no desenvolvimento da lesão [27]. Em indivíduos que viveram 5 anos em latitudes menores que 30 graus e exercendo atividades predominantemente fora de casa, houve 40 vezes mais chance de desenvolver pterígio. Além disso, o uso de óculos com proteção UV e chapéu foram considerados medidas úteis para prevenção da lesão. Outros autores reconhecem a dificuldade de mensurar a exposição à radiação UV e comentam que ela é cumulativa e bastante nociva, principalmente quando ocorre em jovens e persiste por 2 ou 3 décadas [28, 29, 30].

Outro ponto a ser melhor elucidado é a existência de maior número de portadores entre habitantes de populações rurais, já que em pesquisa realizada em Melbourne (Austrália), foram diagnosticadas

ocorrências em 1,2% dos indivíduos da área urbana e 6,7% dos que moram na área rural desta cidade [31]. Este fato pode ter associação apenas com a maior exposição solar a que estão sujeitos os indivíduos que trabalham na lavoura. Dessa forma, podemos ver que há fortes evidências de que a luz ultravioleta representa importante fator de risco para o desenvolvimento da lesão [25].

Na Figura 3 pode-se observar o mapa da incidência média diária de irradiação UV em todo o mundo, o que comprova que nas regiões de maior índice de pterígio coincide-se uma maior incidência de raios UV, ou seja, prevalência de até 22,5 % nas regiões equatoriais, enquanto nas regiões fora do paralelo de 40 graus norte e sul a prevalência diminui para 2% [32]. Foi observada incidência de 4% no Qatar [33], 5,2% na Índia [34], 6,2% na Austrália [35], 8,2% na Nigéria [36], 11,2% no Texas [37], 24,9 % em Manaus e 14,4 % em Salvador [21]. Esta distribuição fortalece a idéia da radiação solar ser importante na patogenia da lesão. Porém, esse fator é questionável, já que ainda não foi possível induzir experimentalmente o pterígio em animais expostos à radiação [38]. Outros fatores etiológicos possíveis são a exposição ao vento, poeira, altas temperaturas e baixa umidade do ar, além de populações que habitam ilhas e altas latitudes [24, 39], de modo que, a agressão crônica aos olhos, na área da fenda palpebral que é a mais exposta, provoca lesões superficiais na córnea e causa ulcerações limbares tróficas indolores, fazendo com que a membrana de Bowman alterada, possibilite a migração dos ceratoblastos como ilhas progressivas subepiteliais, com progressão contínua da cabeça do pterígio em direção ao centro da córnea [40, 41, 42, 43].

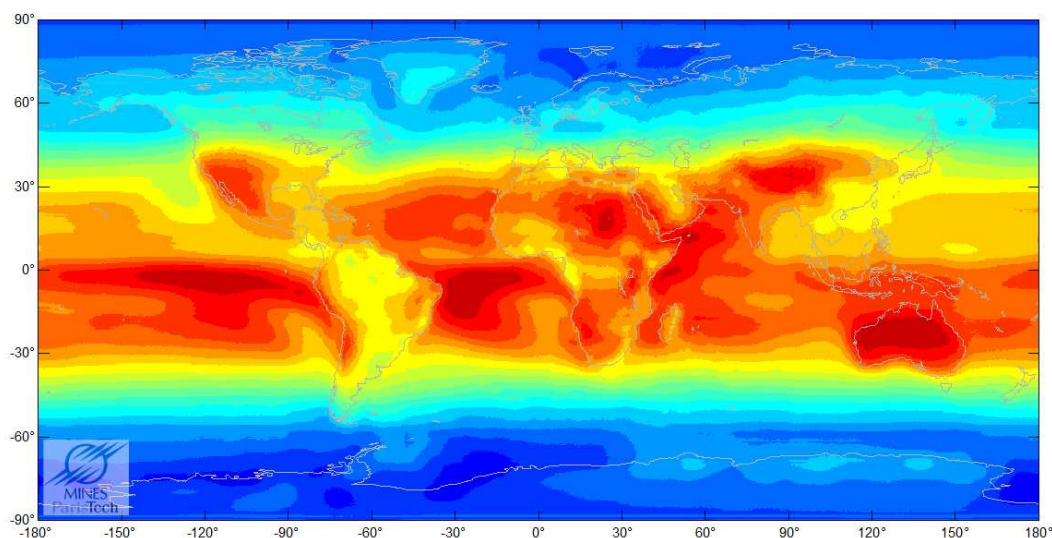


Figura 3: Média Anual de Irradiação Diária de UV(1990-2004) [44]

Já a teoria conhecida como teoria imunológica, sugere que há um desequilíbrio na imunidade mediada por células na conjuntiva bulbar em portadores de pterígio, e foram observados linfócitos CD3, mastócitos, monócitos, macrófagos e células plasmáticas aumentadas nos pterígios [13]. Além disso, já foi observada a alteração do curso normal da apoptose em pterígio, devido à inativação de

supressores do gene p53 [45].

Além de fatores ambientais, aparentemente portadores de pele e cabelos mais pigmentados estão mais propensos a desenvolver o pterígio [19]. Avaliação feita em Barbados mostrou incidência superior de pterígio em negros (23,4% em negros e 10,2% em brancos) [46].

Do ponto de vista histológico, o epitélio que faz parte da lesão é normal, estando a alteração localizada na porção subepitelial, ou seja, na cápsula de Tenon, onde ocorre crescimento fibrovascular com características de degeneração elastótica [47]. O estroma possui também IgE, IgG, plasmócitos e linfócitos, apontando para um possível fator alérgico [48].

Outra suposição para a patogenia é a teoria viral, já que Piras *et al.* [49] e Varinli *et al.* [50] correlacionaram o pterígio com a positividade para HPV. Já Detorakis *et al.* [51] encontraram Vírus herpes simplex e HPV no pterígio e relacionaram esse fato com o risco de recidiva. Gallagher *et al.* [52] confirmaram HPV (tipos 6, 11 e 16) em 50% dos pacientes com pterígio. Já Dusku *et al.* [53] e Chen *et al.* [54] contradisseram essa teoria negando a existência de vírus no pterígio. A positividade ao HPV poderia ser apenas regional. Estudo feito em Botucatu também não observou a positividade para HPV [55]. Uma predisposição hereditária ao pterígio também parece existir. Alguns estudos sugerem um modelo de herança autossômico dominante [38, 56].

## 2.3 Evolução

Por razões desconhecidas, o crescimento do pterígio pode parar em qualquer fase de sua evolução. A afecção pode permanecer quiescente ou voltar a crescer novamente a qualquer momento. O pterígio em sua fase atrófica apresenta uma linha arqueada de depósito epitelial de ferro na frente da cabeça (linha de Stocker) [12]. O pterígio em fase de crescimento ativo pode alcançar a metade correspondente da pupila, porém sem ultrapassá-la. Ambos olhos são comumente envolvidos, mas frequentemente de forma assimétrica [57].

O crescimento do pterígio sobre a córnea, além de causar sintomas como sensação de corpo estranho, ardor, irritação e lacrimejamento, pode desencadear astigmatismo e conseqüente alteração visual [58, 59, 60, 61, 62]. Schellini *et al.* [5] observaram que 1,4% dos portadores de pterígio possuíam a visão impedida pela presença de proliferação fibrosa sobre a área pupilar, em estudo realizado na região de Botucatu (São Paulo). Portanto, é possível encontrar em nosso meio indivíduos portadores de cegueira evitável por pterígio.



## 2.4 Tratamento

Para o pterígio do tipo I, o desconforto pode ser controlado com o uso tópico de lubrificantes, vasoconstrictores, compressas frias e evitando ambientes poluídos com fumaça e partículas em suspensão no ar, e usando lentes com filtro protetor contra raios ultravioleta, ou seja, tratamento clínico [12]. Já portadores de pterígio do tipo 2 e 3 necessitam de tratamentos cirúrgicos [12].

Desde os tempos egípcios, várias técnicas cirúrgicas e muitas variedades de tratamento tem sido relatados com o objetivo de retirar essa hipertrofia, trazendo mais conforto aos pacientes [13]. No entanto, ainda não foi encontrada uma solução definitiva para pterígio, e a chance de recidiva ou recorrência da lesão é bastante variável, 10% [63] a 40% [64, 65], e muitos são os fatores que podem estar a ela relacionados, inclusive a manutenção das condições climáticas e de exposição aos agressores que persistem no ambiente depois do tratamento [12].

Os principais procedimentos cirúrgicos constantes na literatura são [12]:

- excisão simples (esclera exposta, fechamento conjuntival simples e por rotação de retalho conjuntival);
- transplante da cabeça do pterígio;
- transplante de conjuntiva autólogo livre (com ou sem tecido límbico);
- transplante de membrana amniótica e
- transplante lamelar e penetrante de córnea.

Para diminuir a taxa de recorrência, os cirurgiões têm combinado técnicas excisionais com várias modalidades de terapia adjuvante: betaterapia, tiotepa, 5-fluorouracil (5-FU) e mitomicina C (MMC) [12].

Tendo em vista a etiologia incerta e as dificuldades na resolução, os estudos em desenvolvimento do pterígio são muito importantes, já que a alta prevalência e a possibilidade de causar baixa acuidade visual por astigmatismo e cegueira por envolver o eixo visual ou provocar opacidades corneanas é eminente [66]. Devido a etiologia incompreensível, o elevado número de pacientes afetados e as dificuldades no tratamento [16], é muito importante possuímos meios de detecção precoce aliados à melhoria do tratamento. Dessa forma, esse trabalho objetivou reconhecer e caracterizar alguns fatores relacionados ao diagnóstico dessa doença utilizando técnicas de reconhecimento de padrões e técnicas de otimização.

## Capítulo 3

# Técnicas de Classificação de Padrões

Neste capítulo são exploradas as técnicas de Inteligência Artificial (IA) para classificação de padrões. Será feita uma revisão das principais técnicas e será detalhado o funcionamento de cada uma. Inicializará com Redes Neurais Artificiais, uma técnica que apresenta um modelo matemático baseado na estrutura neural de organismos inteligentes. Em seguida se discutirá as Máquinas de Vetores de Suporte, uma técnica que trata a separabilidade das classes com um modelo estatístico. Logo após, será apresentada a técnica denominada K-Vizinhos mais próximos. Finalmente, será destacada a técnica de Floresta de Caminhos Ótimos, a qual é baseada na teoria de grafos, e apresenta ótimos resultados quando comparada com outras técnicas.

### 3.1 Redes Neurais Artificiais

Redes Neurais Artificiais (*Artificial Neural Networks* - ANN) são técnicas computacionais que apresentam um modelo matemático inspirado na estrutura neural de organismos inteligentes e que adquirem conhecimento através da experiência [8]. Uma ANN é caracterizada pelo padrão de conexão (modelo conexionista) entre os neurônios (topologia), pelo método de determinação dos pesos e conexões (treinamento ou aprendizagem) e pela função de ativação responsável pela saída da rede. Esses processos comportam-se de maneira similar aos grupos de neurônios do cérebro humano que recebem e transmitem informações através dos dendritos e axônios, respectivamente [67]. Quando é apresentado à rede um conjunto de entradas e suas respectivas saídas, as quais estão representando o comportamento de um processo específico, a mesma, através do treinamento, é capaz de se auto-ajustar com o objetivo de mapear o relacionamento funcional entre as entradas e saídas. Por conseguinte, após a execução desse algoritmo de treinamento, a rede deve ser capaz de generalizar o comportamento do processo quando outras entradas, diferentes daquelas utilizadas no treinamento, são apresentadas. Tal característica é particularmente útil quando o relacionamento entre as entradas

e saídas do processo analisado é não-linear, ou então, quando o relacionamento não é claramente definido de forma que a sua modelagem, por técnicas de identificação convencionais, torna-se muito difícil.

As redes neurais representam o desenvolvimento de sistemas computacionais capazes de reconhecer e classificar padrões, resolver problemas complexos, realizar processos indutivos e dedutivos, entre outros.

Em geral, a utilização de redes neurais, comparando-se com outras técnicas, possui as seguintes vantagens [68]:

- os relacionamentos funcionais entre os padrões de entrada e saída podem ser capturados por uma rede neural, sendo que tais relacionamentos não precisam ser conhecidos ou descritos explicitamente;
- nenhuma hipótese precisa ser feita sobre as distribuições estatísticas dos padrões de entrada;
- as redes neurais são tolerantes à falhas no sentido que, mesmo quando alguns elementos ou conexões do processamento apresentam-se degradados, a performance da rede pode ser pouco afetada.

As arquiteturas de redes neurais artificiais utilizadas para modelar os sistemas neurais biológicos podem ser divididas em três categorias distintas. A primeira categoria de redes neurais, "feedforward" transforma conjuntos de sinais de entrada em conjuntos de sinais de saída, sendo que os parâmetros dessa transformação são ajustados de forma supervisionada, de acordo com os resultados desejados. Na segunda categoria de redes neurais, "feedback", as informações de entrada definem o estado inicial de atividade do sistema de realimentação. Depois de algumas transições de estado, atinge-se um estado que é considerado o resultado final da computação. Na terceira categoria, as células vizinhas da rede neural interagem mutuamente, a fim de transformarem-se adaptativamente em detectores especializados de diferentes padrões. Nessa última categoria, o aprendizado da rede neural é realizado de forma não supervisionada, através de mapas de auto-organização (*Self-Organizing Maps - SOM*) [69].

### 3.1.1 Mapas de Auto-Organização

No modelo de redes neurais baseado em mapas de auto-organização SOM proposto por Kohonen [70, 69, 71], a segregação espacial de diferentes respostas e suas organizações em sub-conjuntos topologicamente relacionados resultam em um alto grau de eficiência nas operações típicas de uma rede neural.

O modelo SOM implementa uma projeção não-linear de um espaço multidimensional  $X$  em um espaço bidimensional  $M$ , denominado mapa de auto-organização. Essa projeção, de modo análogo

à maioria das projeções encontradas no cérebro, implementam mapeamentos topológicos, ou seja, elementos vizinhos em  $X$  são geralmente mapeados em elementos vizinhos em  $M$ .

O mapeamento topológico constitui uma importante característica do modelo de redes neurais proposta por Kohonen, pois permite encontrar características e outras abstrações do espaço multidimensional através de identificação de agrupamentos ("clusters") no mapa bidimensional.

O mapeamento do espaço multidimensional no bidimensional é realizado por uma função  $f : X \rightarrow M$ , onde  $X \subset R^n$  e  $M \subset R^2$ , que associa cada elemento  $x \in X$  a um par  $(i, j) \in M$ . Os elementos  $m_{ij}$  do mapa  $M$ , assim como os dados de entrada, são vetores de tamanho  $n$ , que mantém os pesos das ligações sinápticas da rede neural.

Apresentado um estímulo específico  $x \in X$  à rede neural, o "nó vencedor" do mapa  $M$  para esse estímulo é determinado verificando-se qual dos elementos do mapa possui a menor distância ao estímulo, ou seja, o nó vencedor tem as coordenadas  $(i, j) \in M$ , tal que:

$$\|m_{ij} - x\| = \min\{\|m_{kl} - x\|, \forall i, j, k, l = 0 \dots D\} \quad (3.1)$$

onde  $D$  é o tamanho de  $M$ .

Portanto, um estímulo  $x \in X$  apresentado à camada de entrada da rede neural é mapeado no elemento  $(i, j)$  da camada de saída da rede neural, ou seja do mapa de auto-organização  $M$ , para o qual a distância entre esse estímulo e o vetor dos pesos das ligações sinápticas  $m_{ij}$  da rede neural seja mínima, comparando-se com os demais vetores de pesos das ligações sinápticas  $m_{kl}$ .

Nesse modelo de rede neural, os neurônios da camada de saída interagem lateralmente com seus vizinhos. Essas interações se dão na fase de aprendizado, quando os pesos da célula "vencedora" e dos seus vizinhos são ajustados, de tal modo que a célula vencedora tenha pesos idênticos aos valores do estímulo sendo apresentado à rede num determinado instante do treinamento, e as células vizinhas tenham pesos semelhantes aos da célula vencedora, sendo que o grau de semelhança é ponderado de acordo com a distância entre a célula vizinha e a célula vencedora.

A interação da célula vencedora  $(i, j)$  com uma célula vizinha  $(k, l)$ , num determinado instante  $t$  do treinamento, é definida por uma função  $h_{kl}(t)$  da distância entre elas. Normalmente, a função  $h_{kl}(t)$  é uma Gaussiana.

O aprendizado nesse modelo é dito não supervisionado porque o mapeamento do espaço multidimensional para o bidimensional é feito a despeito dos resultados esperados na saída do processamento, ao contrário dos outros modelos, onde os valores esperados e os valores obtidos pela rede, para um dado estímulo de entrada, influenciam no reajuste dos pesos das ligações sinápticas.

No modelo de redes neurais baseadas em mapas de auto-organização, os pesos das ligações si-

nápticas são ajustadas da seguinte forma:

$$m_{kl}(t+1) = m_{kl}(t) + (x - m_{kl}(t))h_{kl}(t) \quad (3.2)$$

onde  $h_{kl}(t) = \alpha(t) \exp \left[ -\frac{\|r_{kl} - r_{ij}\|^2}{\delta(t)^2} \right]$ ,  $(i, j)$  são as coordenadas do nó vencedor para o estímulo  $x$ ,  $\|r_{kl} - r_{ij}\|$  é a distância espacial entre o nó vencedor e o nó  $(k, l)$ , e  $\delta(t)$  é uma função do tempo de treinamento  $t$ , que determina a variância da Gaussiana  $h_{kl}$ .

O tamanho da vizinhança  $N_{ij}(t)$  decresce de acordo com o número de iterações  $T$  da fase de treinamento. No início do treinamento, isto é, quando  $t = 0$ ,  $N_{ij}(t)$  inclui toda a rede, e no final, quando  $t = T$ ,  $N_{ij}(t)$  contém somente o nó vencedor  $(i, j)$ . A seguinte função linear pode ser adotada para determinar os ajustes nas extensões da vizinhança:  $N_{ij}(t) = D \left[ 1 - \frac{t}{T} \right]$ , para  $t = 0, \dots, T$ .

De forma análoga, as funções  $\delta(t)$  e  $\alpha(t)$ , podem ser reajustadas em cada iteração através das funções lineares  $\delta(t) = 2 - \frac{t}{T}$  e  $\alpha(t) = 1 - \frac{t}{T}$ , para  $t = 1, \dots, T$ , respectivamente.

Após a fase de treinamento, os nós da rede são rotulados de acordo com a classe de estímulos para os quais eles apresentam respostas mais intensas. Se essa fase for supervisionada, então deve-se apresentar para a rede conjuntos de estímulos para os quais se conhece a classificação. Para cada classe de estímulos conhecidos verifica-se quais são os nós vencedores no mapa de auto-organização. Esses nós são rotulados com o rótulo associado àquela classe de estímulos. Se essa fase não for supervisionada, é necessário fazer uma análise de agrupamentos no mapa de auto-organização.

Um modelo da arquitetura bidimensional típica da rede de Kohonen é apresentado na Figura 4.

## 3.1.2 Redes Neurais Perceptron Multicamadas

### 3.1.2.1 Perceptron

Durante as décadas de 50 e 60, uma grande revolução ocorreu no campo de pesquisa relacionado à teoria de reconhecimento de padrões com o aparecimento das chamadas máquinas que aprendem, os perceptrons [8].

Em sua metodologia mais básica, o perceptron aprende uma função de decisão linear que dicotomiza dois conjuntos de treinamento, os quais são linearmente separáveis, em um número finito de passos iterativos. A resposta desse dispositivo básico baseia-se na soma ponderada de sua entrada, ou seja,

$$d(x) = \phi(\chi); \quad (3.3)$$

$$\chi = \sum_{i=1}^n w_i x_i + w_{n+1} \quad (3.4)$$

que é uma função de decisão linear em relação aos elementos do vetor de características. Os coefici-

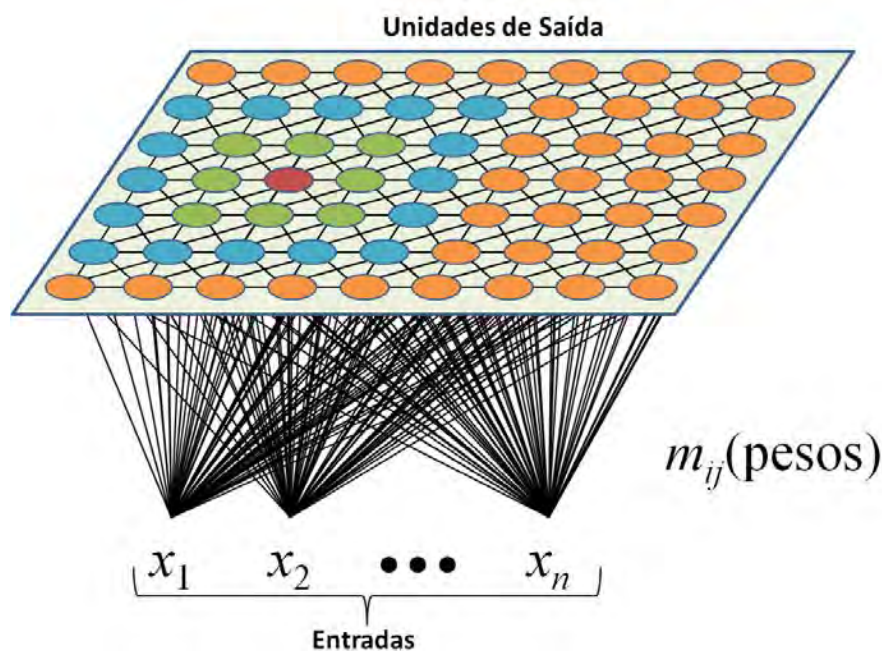


Figura 4: Arquitetura bidimensional típica de uma rede de Kohonen adaptada de [72].

entes  $w_j$ ,  $j = 1 \dots n+1$ , são os chamados pesos das conexões, sendo análogos às sinapses no sistema neural humano. Tem-se ainda que  $\phi$  é a chamada função de ativação e  $\chi$  é o somatório da rede [8].

Dentre as várias escolhas para  $\phi$ , tem-se:

- Função limiar (threshold):

$$\phi(\chi) = \begin{cases} 1 & \text{se } \chi \geq 1, \\ 0 & \text{caso contrário} \end{cases} \quad (3.5)$$

- Função sigmoideal:

$$\phi(\chi) = \frac{1}{1 + \exp(-\chi)} \quad (3.6)$$

- Função identidade:

$$\phi(\chi) = \chi \quad (3.7)$$

A Figura 5 ilustra essas funções.

O algoritmo de treinamento do perceptron utiliza uma função de ativação por limiarização muito semelhante à Equação (3.5), descrita por

$$\phi(\chi) = \begin{cases} 1 & \text{se } \chi \geq 1, \\ -1 & \text{caso contrário.} \end{cases} \quad (3.8)$$

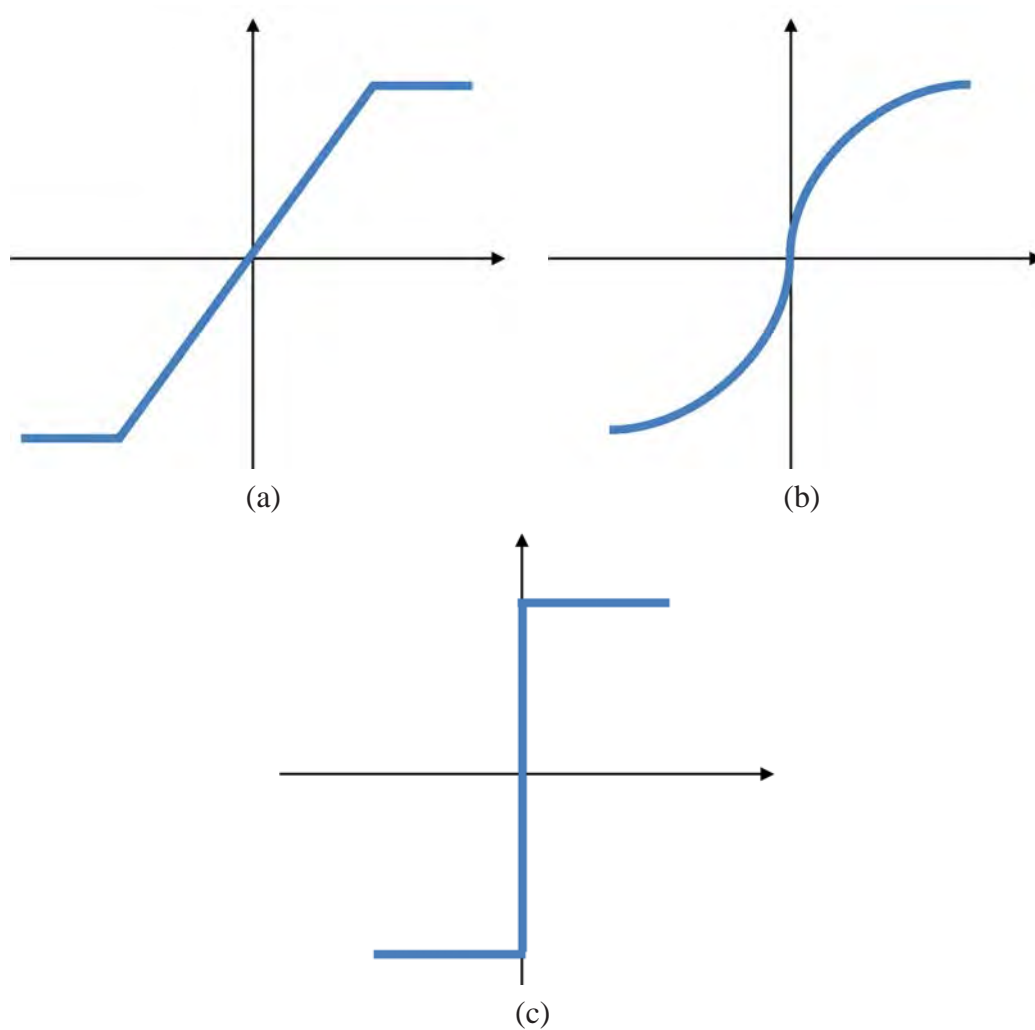


Figura 5: Funções de Ativação Linear (a), Sigmoidal (b) e Identidade (c).

Geometricamente, a equação

$$\chi = \sum_{i=1}^n w_i x_i + w_{n+1} = 0 \quad (3.9)$$

ou

$$\chi = w_1 x_1 + w_2 x_2 + \dots + w_n x_n + w_{n+1} = 0 \quad (3.10)$$

define um hiperplano no espaço  $n$ -dimensional de padrões. Os  $n$  primeiros coeficientes estabelecem a orientação do hiperplano, enquanto o último coeficiente,  $w_{n+1}$ , também chamado de *bias*, é proporcional à distância perpendicular à origem do hiperplano.

Caso as duas classes apresentadas ao algoritmo de perceptron sejam linearmente separáveis em  $\mathfrak{R}^n$ , o mesmo converge em um número finito de passos. Entretanto, caso ambas classes não obedeçam a tal critério, o algoritmo executará indefinidamente, não convergindo para a solução ótima.

O modelo simplificado do perceptron é apresentado na Figura 6.

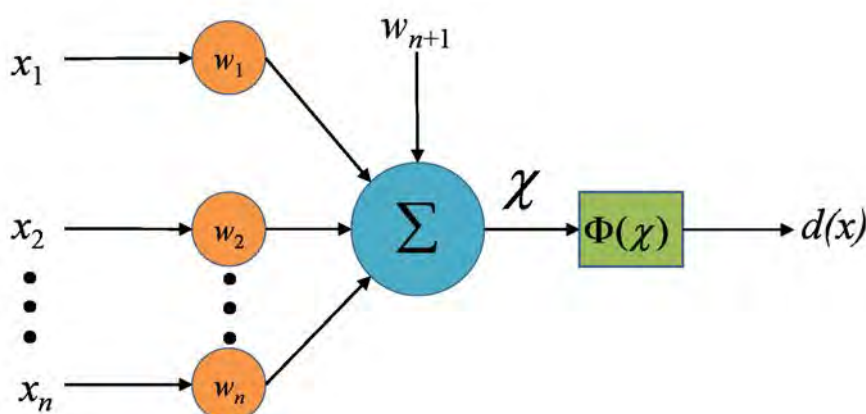


Figura 6: Modelo simplificado do Perceptron adaptado de [72].

### 3.1.2.2 Perceptron Multicamadas

Para o tratamento de problemas que possuem várias classes, independentemente de as mesmas serem separáveis ou não, o algoritmo de Perceptron visto anteriormente é ineficiente, visto que o mesmo funciona adequadamente quando a tarefa de classificação consiste em duas classes linearmente separáveis. Conectando vários perceptrons, pode-se projetar uma estrutura chamada Perceptron Multicamadas, também conhecida como MLP [8], a qual consiste de várias camadas de elementos computacionais idênticos (neurônios) dispostos de tal maneira que a saída de cada um deles alimente a entrada de cada neurônio da camada seguinte. O número de neurônios da primeira camada, denominada  $A$ , corresponde a  $N_A$ ; sendo, frequentemente denotado  $N_A = n$ , ou seja, o número de elementos



da primeira camada é igual à dimensionalidade do vetor de características. O número de neurônios da camada de saída, chamada  $Q$ , corresponde a  $N_Q = M$ , sendo  $M$  igual ao número de classes do problema de classificação. A rede neural reconhece um vetor de padrões  $\mathbf{x}$  como pertencente à classe  $\omega_m$  caso a  $m$ -ésima saída da rede possuir o maior valor dentre as  $m - 1$  saídas restantes.

No caso da arquitetura citada anteriormente, a entrada de cada elemento em qualquer camada corresponde à soma ponderada das saídas da camada anterior. Denotando  $K$  a camada anterior a  $J$ , tem-se que a entrada de cada neurônio na camada  $J$ , denotada por  $I_j$  é dada por

$$I_j = \sum_{k=1}^{N_K} w_{jk} O_k \quad (3.11)$$

sendo que

$$O_k = \phi(I_k), \quad (3.12)$$

onde  $j = 1, 2, \dots, N_J$ , sendo  $N_J$  e  $N_K$  o número de elementos da camada  $J$  e  $K$ , respectivamente, e  $w_{jk}$  são os pesos que modificam as saídas  $O_k$  dos elementos da camada  $K$ .

Algumas convenções geralmente adotadas para tais estruturas de redes neurais são:

- a função de ativação utilizada na camada de entrada é a função identidade, dada pela Equação (3.7);
- camadas não adjacentes não são conectadas diretamente e, por conseguinte,
- todos os neurônios das camadas escondidas possuem a mesma função de ativação  $\phi$ .

Sendo assim, este modelo de rede neural é chamado de *feedforward*, devido ao fato de tanto a camada de entrada quanto as intermediárias, ou escondidas, serem submetidas somente à camada mais alta, ou seja, a camada  $K$  é subalterna à camada  $K - 1$ , assim por diante.

Uma representação de um modelo básico do modelo *feedforward* pode se visto na Figura 7.

### 3.1.2.3 Treinamento por Retropropagação

O principal objetivo deste algoritmo é desenvolver uma regra de treinamento com o intuito de minimizar o erro quadrático total entre as saídas desejadas  $r_q$  e as saídas reais  $\phi_q$  dos nós em uma camada de saída  $Q$ , ou seja, minimizar a equação abaixo:

$$E_Q = \frac{1}{2} \sum_{q=1}^{N_Q} (r_q - O_q)^2, \quad (3.13)$$

onde  $N_Q$  é o número de nós da camada de saída  $Q$ .

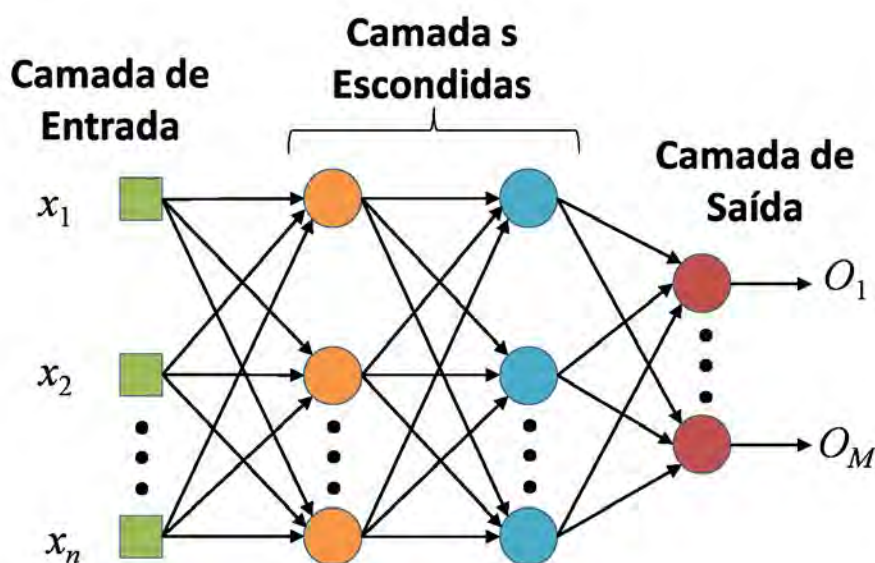


Figura 7: Representação básica de uma rede *feedforward* (multicamadas) adaptada de [72].

Durante o treinamento com o algoritmo de retropropagação, primeiramente, um padrão é apresentado à camada de entrada da rede. A saída obtida pela última camada é comparada com a saída desejada. Caso a mesma não estiver correta, o erro é calculado através da Equação (3.13) e propagado a partir da camada de saída até a camada de entrada, sendo que os pesos das conexões são modificados através da regra delta generalizada conforme o erro é retropropagado.

Desta forma, a regra delta generalizada  $\Delta w_{qp}$  permite o ajuste dos pesos em cada uma das camadas de maneira a tentar minimizar a função de erro mostrada acima, ou seja, calcular o gradiente negativo de  $E_Q$ , dado pela equação

$$\Delta w_{qp} = -\alpha \frac{\partial E_Q}{\partial w_{qp}}, \quad (3.14)$$

em que  $P$  precede a camada  $Q$  e  $\alpha$  é um incremento positivo de correção.

Pode-se resumir e generalizar o procedimento de treinamento da seguinte maneira: para quaisquer duas camadas  $K$  e  $J$ , em que  $K$  precede imediatamente  $J$ , a Equação (3.14), através de algumas manipulações algébricas, pode ser re-escrita como

$$\Delta w_{jk} = \alpha \delta_j O_k. \quad (3.15)$$

Caso  $J$  seja uma camada de saída,  $\delta_j$  é dado por:

$$\delta_j = (r_j - O_j) O'_j, \quad (3.16)$$

onde  $O'_j$  denota a derivada de  $O_j$ .

Se  $J$  for uma camada interna e  $P$  for a próxima camada, então  $\delta_j$  é dado por

$$\delta_j = O'_j \sum_{p=1}^{N_P} \delta_p w_{jp}, \quad (3.17)$$

onde  $j = 1, 2, \dots, N_J$ .

Desta forma, todo o processo de treinamento começa com um conjunto arbitrário de pesos da rede. Em seguida, a aplicação da regra delta generalizada em qualquer passo iterativo envolve duas etapas. Na primeira fase, um vetor de treinamento é apresentado à rede e propagado através das camadas da rede para o cálculo de  $O_j$  para cada nó. As saídas  $O_q$  dos nós da camada de saída são então comparadas com as respostas desejadas  $r_q$  para que os termos de erro  $\delta_q$  sejam gerados. A segunda fase envolve uma passagem para trás na rede, retropropagação, durante a qual o sinal de erro apropriado é passado por cada nó e as mudanças correspondentes nos pesos são realizadas. Uma vez que o sistema tenha sido treinado, o mesmo passa a classificar os padrões utilizando os parâmetros estabelecidos durante a fase de treinamento descrita acima.

## 3.2 Máquinas de Vetores de Suporte

As Máquinas de Vetores de Suporte, do inglês "Support Vector Machines" (SVM), são normalmente consideradas a primeira aplicação prática da teoria do aprendizado estatístico [73, 74]. Trata-se de uma área de pesquisa que oferece muitas opções para se trabalhar, sendo grande parte delas mais conceituais que meramente técnicas. Nos últimos anos seu escopo tem crescido significativamente tanto em termos de novos algoritmos quanto de um entendimento teórico mais aprofundado. Parte destes novos algoritmos se deve aos chamados métodos de nuclearização, uma proposta para solução de problemas de *aprendizado de máquina* cuja arquitetura tem se demonstrado capaz de lidar com questões relativas às bases desta teoria. Além disso, aplicações bem sucedidas de SVM demonstram que esta técnica não só possui uma fundamentação mais sólida do que as Redes Neurais Artificiais como também são capazes de substituí-las com desempenho melhor ou semelhante [74].

As SVM propõem resolver o problema de classificação de padrões assumindo ser possível separar as classes em um espaço de mais alta dimensão. Suponha uma situação na qual os dados não são linearmente separáveis. Tais amostras podem ser separadas em grupos usando curvas ou círculos como superfícies de decisão, porém encontrar tais limiares é uma tarefa custosa. A principal ideia de uma SVM é pré-processar os dados de tal forma que o problema de encontrar uma função discriminante não-linear seja transformado em um problema de encontrar um hiperplano, ou seja, mapear os dados que estão em uma dimensão qualquer para outra maior, tornando os mesmos linearmente separáveis. Isto é feito definindo um mapeamento que transforma o vetor de entrada em outro (usualmente

maior) vetor. Espera-se que, escolhendo um mapeamento adequado, o novo conjunto de treinamento seja linearmente separável.

### 3.2.1 Classificadores por Hiperplano

De acordo com as questões destacadas para o controle da efetividade dos algoritmos de aprendizado, se faz necessário que a capacidade da classe de funções possa ser calculada. Nos primórdios de seu estudo, Vapnik [73] considerou uma classe de hiperplanos em um espaço  $\mathcal{H}$  com produto interno,

$$\langle \mathbf{w}, \mathbf{x} \rangle - b = 0 \quad (3.18)$$

onde  $\mathbf{w} \in \mathcal{H}, b \in \mathbb{R}$ , correspondendo com funções de decisão do tipo

$$f(x) = \text{sgn}(\langle \mathbf{w}, \mathbf{x} \rangle - b) \quad (3.19)$$

Baseado em dois argumentos, ele então propôs o algoritmo de aprendizado denominado Retrato Generalizado (do inglês "*Generalized Portrait*") para problemas separáveis por hiperplanos.

1. Dentre todos os hiperplanos que separam os dados, existe apenas um hiperplano ótimo distinguido pela margem de máxima separação entre qualquer ponto de treinamento e este hiperplano. Esta é a solução de

$$\underset{\mathbf{w} \in \mathcal{H}, b \in \mathbb{R}}{\text{maximize}} \quad \min\{\|\mathbf{x} - \mathbf{x}_i\| \mid \mathbf{x} \in \mathcal{H}, \langle \mathbf{w}, \mathbf{x} \rangle - b = 0\} \quad (3.20)$$

onde  $i = 1, \dots, m$ .

2. A capacidade da classe de hiperplanos de separação decresce com o crescimento da margem.

Para construir tal hiperplano ótimo, é necessário resolver

$$\underset{\mathbf{w} \in \mathcal{H}, b \in \mathbb{R}}{\text{minimize}} \quad \tau(\mathbf{w}) = \frac{1}{2} \|\mathbf{w}\|^2 \quad (3.21)$$

sujeito à

$$(\langle \mathbf{w}, \mathbf{x}_i \rangle - b) \geq 1 \quad \text{para todo } i = 1, \dots, m \quad (3.22)$$

ou

$$(\langle \mathbf{w}, \mathbf{x}_i \rangle - b) \leq -1 \quad \text{para todo } i = 1, \dots, m. \quad (3.23)$$

Temos que as Equações 3.22 e 3.23 pode ser escritas como:

$$y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle - b) \geq 1 \quad \text{para todo } i = 1, \dots, m, \quad (3.24)$$

onde  $y_i \in \{+1, -1\}$  denota o rótulo da amostra  $x_i$ .

A razão por detrás da minimização de  $w$  (Equação (3.21)) pode ser interpretada da seguinte maneira: se  $\|w\|$  fosse 1, então o termo da esquerda na Equação (??) seria igual à distância de  $x_i$  ao hiperplano. Em geral, é preciso dividir  $y_i(\langle w, x_i \rangle - b)$  por  $\|w\|$  para transformá-lo nesta distância. Sendo assim, se a Equação (??) for satisfeita para todos  $i = 1, \dots, m$  com um  $w$  de tamanho mínimo, a margem será maximizada como um todo. Um resumo destes argumentos é dado na Figura 8.

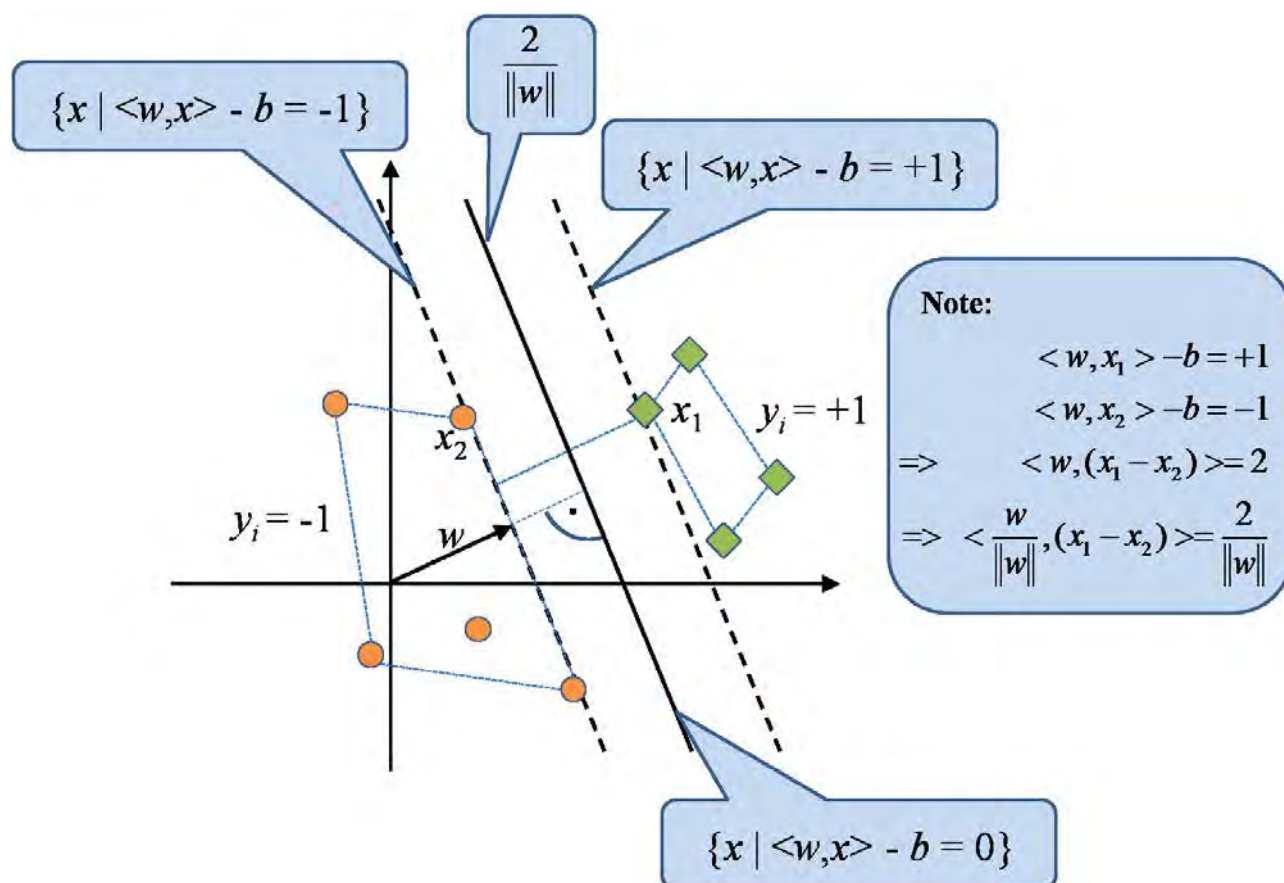


Figura 8: Um problema de classificação binária extraído de [74] cujo propósito é separar as bolas dos diamantes. O hiperplano ótimo (Equação (3.21)) é mostrado com uma linha sólida. No problema sendo separado há apenas um vetor ponderado  $w$  e um limiar  $b$  tal que  $y_i(\langle w, x_i \rangle + b) > 0$  ( $i = 1, \dots, m$ ). Ao reescalar  $w$  e  $b$  com o objetivo de que o(s) ponto(s) mais próximos ao hiperplano satisfaçam  $|\langle w, x_i \rangle + b| = 1$  se obtém a forma *canônica*  $(w, b)$  do hiperplano.

A função  $\tau$  na Equação (3.21) é chamada de *função objetivo*, enquanto a Equação (??) representa as *restrições de desigualdade*. Juntas elas dão origem ao que se conhece como *problema de otimização restrita*, que por sua vez é resolvido através da introdução do *Lagrangiano*

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^m \alpha_i (y_i (\langle \mathbf{x}_i, \mathbf{w} \rangle + b) - 1) \quad (3.25)$$

onde  $\alpha_i \geq 0$  são os *multiplicadores de Lagrange* e  $L$  tem que ser minimizada com relação às *variáveis primais*  $\mathbf{w}$  e  $b$  e maximizada com relação às *variáveis duais*  $\alpha_i$ . Em outras palavras, um ponto de sela deve ser encontrado. Como se pode notar, as restrições do problema original foram incorporadas no segundo termo no Lagrangiano.

Olhando melhor para o que o problema de otimização restrita nos mostra, é possível observar que se a restrição da Equação (??) no segundo termo do Lagrangiano for violada com  $\sum_{i=1}^m \alpha_i (y_i (\langle \mathbf{x}_i, \mathbf{w} \rangle + b) - 1) < 0$ ,  $L$  pode ser incrementado através do decréscimo do  $\alpha_i$  correspondente. Ao mesmo tempo,  $\mathbf{w}$  e  $b$  terão que mudar (reduzindo a margem) de tal forma que  $L$  decresça. Dado que o problema é separável, num dado momento as restrições serão finalmente satisfeitas. De forma análoga, podemos entender que para todas as restrições não atendidas precisamente como igualdades, seu  $\alpha_i$  correspondente deve ser 0: este é o valor de  $\alpha_i$  que maximiza  $L$ . Desta última observação extrai-se as chamadas condições de complementaridade de Karush-Kuhn-Tucker (KKT) presente na teoria da otimização.

Considerando que num ponto de sela as derivadas de  $L$  com relação às variáveis primais deve desaparecer,

$$\frac{\partial}{\partial b} L(\mathbf{w}, b, \alpha) = 0 \quad \text{e} \quad \frac{\partial}{\partial \mathbf{w}} L(\mathbf{w}, b, \alpha) = 0 \quad (3.26)$$

nos leva à

$$\sum_{i=1}^m \alpha_i y_i = 0 \quad (3.27)$$

e

$$\mathbf{w} = \sum_{i=1}^m \alpha_i y_i \mathbf{x}_i \quad (3.28)$$

Conseqüentemente, o vetor-solução nada mais é que uma expansão (Equação (3.28)) de um subconjunto dos padrões de treinamento, especificamente aqueles com  $\alpha_i$  diferente de zero, que são os chamados vetores de suporte (*Support Vectors* - SVs).

Ao substituir as Equações (3.27) e (3.28) no Lagrangiano (Equação (3.25)), elimina-se as variáveis primais  $\mathbf{w}$  e  $b$  resultando no chamado *problema de otimização dual*, que é o tipo de problema que as SVM normalmente resolvem na prática:

$$\underset{\alpha \in \mathcal{R}^m}{\text{maximize}} \quad W(\alpha) = \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=1}^m \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \quad (3.29)$$

sujeito à

$$\alpha_i \geq 0 \quad \text{para todo } i = 1, \dots, m$$

e

$$\sum_{i=1}^m \alpha_i y_i = 0$$

Ainda baseado na Equação (3.28), a função-hiperplano de decisão (Equação (3.19)) pode ser escrita como

$$f(x) = \text{sgn} \left( \sum_{i=1}^m y_i \alpha_i \langle \mathbf{x}, \mathbf{x}_i \rangle - b \right) \quad (3.30)$$

Para fechar esta parte, nos falta estabelecer como  $b$  pode ser calculado. De acordo com as condições KKT, todos os pontos com

$$\alpha_i [y_i (\langle \mathbf{x}_i, \mathbf{w} \rangle - b) - 1] = 0 \quad \text{para todo } i = 1, \dots, m \quad (3.31)$$

são SVs situados na margem e, sendo assim,  $\alpha_i > 0$ . Em tais casos, é possível demonstrar que

$$\langle \mathbf{x}_i, \mathbf{w} \rangle - b = y_i \quad (3.32)$$

Dessa forma, o limiar pode ser obtido, por exemplo, por

$$b = y_i - \langle \mathbf{x}_i, \mathbf{w} \rangle \quad (3.33)$$

Alternativamente,  $b$  também pode ser calculado a partir dos valores das variáveis duais  $\alpha_i$  e  $\alpha_j$  correspondentes.

### 3.2.2 Nuclearização e Hiperplanos de Margem Suave

Esta seção está relacionada ao mapeamento das entradas para um espaço de maior dimensionalidade e ao acréscimo de artifícios de tolerância para que as SVM possam lidar com problemas não-separáveis.

Para expressar as fórmulas a partir dos padrões de entrada  $\mathcal{X}$ , emprega-se o produto interno dos vetores  $\mathbf{x}, \mathbf{x}'$  em termos do núcleo  $k$  estimado pelos elementos de entrada  $x, x'$

$$k(x, x') = \langle \mathbf{x}, \mathbf{x}' \rangle \quad (3.34)$$

Esta substituição, também conhecida como truque do núcleo (do inglês "kernel trick"), é usada

para estender o conceito de classificação por hiperplanos para máquinas de vetores de suporte não-lineares. O truque do núcleo pode ser aplicado desde que todos os vetores de característica ocorram apenas em produto internos, o que pode ser observado nas Equações (3.29) e (3.30). O vetor ponderado (Equação (3.28)) torna-se então uma expansão no espaço de características e, sendo assim, não mais corresponderá ao seu respectivo vetor no espaço de entradas. A partir da Equação (3.30), a função de decisão se tornará

$$\begin{aligned} f(x) &= \operatorname{sgn} \left( \sum_{i=1}^m y_i \alpha_i \langle \Phi(x), \Phi(x_i) \rangle - b \right) \\ &= \operatorname{sgn} \left( \sum_{i=1}^m y_i \alpha_i k(x, x_i) - b \right) \end{aligned} \quad (3.35)$$

e o problema de otimização (Equação (3.29)) assumirá a forma

$$\underset{\alpha \in \mathcal{R}^m}{\text{maximize}} \quad W(\alpha) = \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=1}^m \alpha_i \alpha_j y_i y_j k(x_i, x_j) \quad (3.36)$$

sujeito à  $\alpha_i \geq 0$  para todo  $i = 1, \dots, m$ , e  $\sum_{i=1}^m \alpha_i y_i = 0$ . A Figura 9 captura a ideia de mapear os dados do espaço de entradas para o espaço de características.

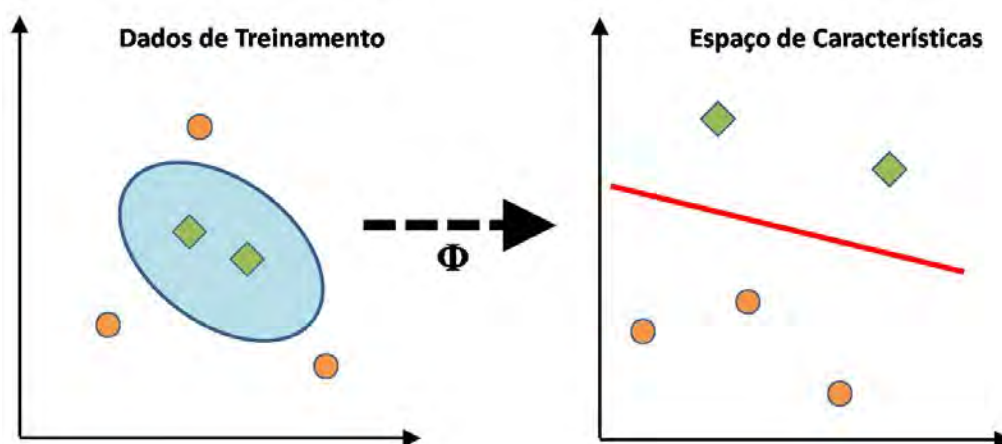


Figura 9: A ideia de se aplicar o truque do núcleo nas SVM: mapear os dados de treinamento (*input space*) em um espaço de características (*feature space*) de maior dimensão através de  $\Phi$  e construir lá um hiperplano de máxima margem de separação. Apesar da fronteira de decisão ser linear no espaço de características, no espaço de entradas original ela ganha formas não-lineares. Adaptado de [72].

Mesmo com as vantagens da nuclearização do problema, na prática, o hiperplano de separação ainda assim pode não existir. Isto pode acontecer, por exemplo, em um conjunto de dados com muito



ruído onde haja sobreposição das classes. Para permitir que os exemplos que violam a Equação (??) possam ser considerados, as variáveis de afrouxamento  $\xi_i$  são introduzidas [75, 74], onde

$$\xi_i \geq 0 \quad \text{para todo } i = 1, \dots, m \quad (3.37)$$

fazendo com que as restrições assumam a forma

$$y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle - b) \geq 1 - \xi_i \quad \text{para todo } i = 1, \dots, m \quad (3.38)$$

Desta maneira, um classificador de boa generalização pode ser obtido pelo controle tanto da capacidade (através de  $\|\mathbf{w}\|$ ) quando da soma das variáveis de afrouxamento  $\sum_i \xi_i$ . É possível demonstrar que este somatório provê um limite superior no número de erros de treinamento [74].

Neste contexto, uma possível elaboração de tal classificador de margem suave é obtida pela minimização da função-objetivo

$$\tau(\mathbf{w}, \xi) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^m \xi_i \quad (3.39)$$

sujeita às restrições da Equações (3.37) e (3.38), onde a constante  $C > 0$  determina o equilíbrio entre a maximização da margem e a minimização dos erros de treinamento. Quanto maior for  $C$ , menor será a margem, menor o número de erros de treinamento e menor também a capacidade de generalização da máquina de aprendizado.

Incorporando o núcleo e reescrevendo o problema em termos dos multiplicadores de Lagrange novamente nos leva a maximização da Equação (3.36), sujeita às restrições  $0 \leq \alpha_i \leq C$  para todo  $i = 1, \dots, m$  e  $\sum_{i=1}^m \alpha_i y_i = 0$ . A única diferença deste caso para o caso separável está no limite superior  $C$  dos multiplicadores de Lagrange  $\alpha_i$ . Sendo assim, a influência dos padrões individuais (que podem ser valores discrepantes) se torna limitada. A solução na Equação (3.35) não muda e o limiar  $b$  pode ser calculado pela exploração do fato de que para todos os SVs  $x_i$  com  $\alpha_i < C$ , a variável de afrouxamento  $\xi$  é zero e, assim sendo

$$\sum_{j=1}^m \alpha_j y_j k(x_i, x_j) - b = y_i \quad (3.40)$$

Analisando geometricamente, escolher  $b$  significa deslocar o hiperplano, ao passo que a Equação (3.40) determina ser necessário situá-lo de tal forma que todo SVs com variável de afrouxamento iguais à zero estejam nas linhas  $\pm 1$  (conforme Figura 8).

A função de núcleo apresentada como produto interno na Equação (3.34) é normalmente chamada de função de Núcleo Linear e, dadas algumas restrições [74], pode ser substituída por outras funções.

Outra função de mapeamento bastante conhecida é a gaussiana chamada Função de Base Radial (do inglês "*Radial Basis Function*" - RBF),

$$k(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right) \quad (3.41)$$

onde  $\sigma > 0$  é um parâmetro da função de núcleo. Em função da variável de ajuste  $C$ , este tipo de SVM é comumente chamado de C-SVC e representa a classificação por SVM em sua forma original [76].

### 3.2.3 Considerações Adicionais

Como foi possível observar ao longo desta seção, as SVM são fundamentalmente máquinas de classificação binária. No entanto, a maioria dos problemas reais a serem otimizados requerem a classificação de mais de duas classes. Uma questão que emerge desta constatação é sobre como lidar com a propriedade multiclases inerente a muitos problemas em um classificador binário por definição. Três estratégias são comumente adotadas para emprego de SVM em problemas de  $k$  classes:

**Uma Contra o Resto:** Na abordagem uma-contra-o-resto,  $k$  SVM são treinadas, onde cada uma delas separa uma determinada classe de todas as demais.

**Classificação aos Pares:** Na abordagem de classificação aos pares,  $k(k - 1)/2$  máquinas são treinadas. Cada uma destas SVM separa um par de classes. Os classificadores são então arranjadas como nós de uma árvore, onde cada nó representa uma SVM. Tanto a busca de cima para baixo quanto de baixo para cima podem ser adotadas, sendo esta última análoga ao processo de eliminação de equipes em um campeonato esportivo.

**Espaço de diferenças:** Para transformar problemas multiclases em binários, um novo espaço de representações chamado espaço de diferenças foi proposto em [77]. Através da modelagem de dissimilaridades, um problema com  $k$  classes é transformado em um problema com os conjuntos diferenças intraclasse e diferenças interclasses. Quanto mais parecida do conjunto intraclasse for a diferença entre um exemplo conhecido e um desconhecido, mais provável os dois serem da mesma classe. Por outro lado, quanto mais parecida do conjunto interclasse for a diferença entre um exemplo conhecido e um desconhecido, mais provável deles serem de classes distintas.

Uma biblioteca de código aberto bastante conhecida que implementa SVM nas mais variadas formas é a chamada LibSVM [78]. Trata-se de um repositório de código aberto cujo objetivo é ajudar as pessoas a usarem facilmente SVM. Além de prover 4 diferentes tipos de função de núcleo,

ela oferece SVM não só para classificação, mas também para "clusterização" e regressão. Através de uma interface por linhas de comando, é possível manipular os conjuntos de treinamento e teste, treinar SVM de diferentes maneiras, buscar por parâmetros ótimos das funções de núcleo e executar as classificações.

### 3.3 *k*-Vizinhos Mais Próximos

O classificador *k*-Vizinhos Mais Próximos (*k*-NN, do inglês *k-Nearest Neighbor*), que é um método bastante utilizado em aplicações que envolvem tarefas de classificação, pois é de fácil entendimento e implementação. O *k*-NN é um classificador onde o aprendizado é baseado na analogia [79].

A regra de classificação do *k*-NN [80] é um método de classificação supervisionado e não-paramétrico, onde um padrão é dito pertencer a uma classe de acordo com a quantidade de vizinhos que pertençam a essa classe, conforme um critério de distância, normalmente usa-se a distância Euclidiana, porém existem outras distâncias como a Manhattan e a Minkowski.

Seja  $p = (p_1, p_2, \dots, p_n)$  e  $q = (q_1, q_2, \dots, q_n)$  dois pontos de  $\mathbb{R}^n$ :

- A distância Euclidiana entre  $p$  e  $q$  é dada por:

$$d(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \dots + (p_n - q_n)^2}, \quad (3.42)$$

- A distância Manhattan entre  $p$  e  $q$  é dada por:

$$d(p, q) = |p_1 - q_1| + |p_2 - q_2| + \dots + |p_n - q_n|, \quad (3.43)$$

- A distância Minkowski entre  $p$  e  $q$  é dada por:

$$d(p, q) = (|p_1 - q_1|^j + |p_2 - q_2|^j + \dots + |p_n - q_n|^j)^{1/j}, \quad (3.44)$$

onde  $j \in \mathbb{N}$ .

A distância Minkowski é a generalização das duas distâncias anteriores. Quando  $j = 1$ , esta distância representa a distância de Manhattan e quando  $j = 2$ , a distância Euclidiana. Os classificadores representam diferentes paradigmas, sendo que o *k*-NN faz parte do chamado *lazy learning*, pois não há um treinamento explícito, considerando que cada padrão de treinamento é adicionado nos vetores de treinamento, diferente do SVM e da ANN-MLP com *backpropagation* que estão, respectivamente, no contexto de aprendizado estatístico e conexionista. Logo, uma de suas principais vantagens do *k*-NN é o pequeno tempo de treinamento, pois se resume apenas a adicionar padrões nos vetores.

O conjunto de treinamento é formado por vetores  $n$ -dimensionais e cada elemento deste conjunto representa um ponto no espaço  $n$ -dimensional.

O método  $k$ -NN é um método de classificação que não possui processamento na fase de treinamento. Para cada padrão de teste é medida a distância entre ele e todos os padrões de treinamento. Verifica-se a quais classes pertencem os  $k$  padrões mais próximos e a classificação é feita, associando-se o padrão de teste à classe que for predominante [79]. O Algoritmo 1 implementa esta ideia.

#### Algoritmo 1 – CLASSIFICADOR $k$ -NN

ESTRUTURAS AUXILIARES: Variáveis  $p$ , e  $k$ .

1. **Para** cada padrão de teste  $p$ , **faça**
2.     |     Selecionar os  $k$  padrões de treinamento mais próximos de  $p$ .
3.     |     Determinar a classe de  $p$  como sendo a classe de maior ocorrência entre  $k$  padrões.
4.     |     Em case de empate, selecione uma classe aleatoriamente.

A classificação pode ser para o caso em que o parâmetro  $k$  é igual a 1 (NN), e se o  $k$  for maior que um, por exemplo,  $k = 3$ , na qual são considerados três vizinhos do novo padrão, ou seja, são analisadas as três menores distâncias do novo padrão para os padrões de treinamento. A classe que tiver o maior número de padrões entre essas distâncias é que irá determinar a classe do novo padrão.

A Figura 10 ilustra esse processo de classificação. Para classificar um ponto, primeiro tomam-se os  $k$ -vizinhos mais próximos dele e, dentro desse conjunto, encontra-se a classe mais representativa. Na Figura 10, o ponto Desconhecido 1 será classificado como classe B e o ponto Desconhecido 2 será classificado como classe A.

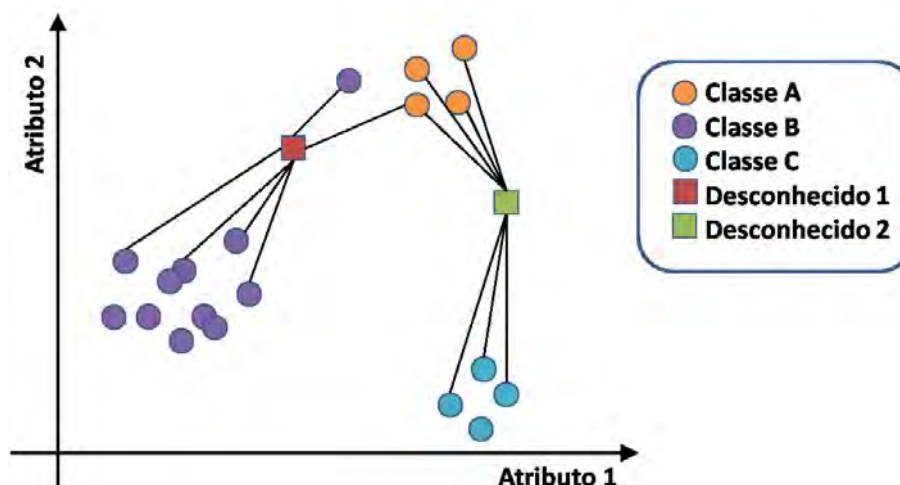


Figura 10: Processo de classificação do  $k$ -NN Adaptado de [81].

Deve-se ter cuidado na escolha do parâmetro  $k$  para ser um número ímpar, pois um número par poderia causar conflito, quando um ponto tivesse o mesmo número de vizinhos de cada classe, por exemplo. A classificação  $k$ -NN é bem simples, porém tem a desvantagem de ter que armazenar todos os dados na memória, ou seja, gerar classificadores complexos, por causa do armazenamento, além da necessidade de se realizar muitos cálculos de distância.

O  $k$ -NN é um classificador que possui apenas um parâmetro livre (o número de  $k$ -vizinhos) que é controlado pelo usuário com o objetivo de obter uma melhor classificação. Este processo de classificação pode ser computacionalmente exaustivo se considerado um conjunto com muitos dados. Para determinadas aplicações, no entanto, o processo é bem aceitável.

## 3.4 Floresta de Caminhos Ótimos

É apresentada nesta seção uma técnica de classificação supervisionada de padrões baseadas em grafos, chamada Floresta de Caminhos Ótimos (*Optimum-Path Forest - OPF*) [10]. Abordagens baseadas em OPF tratam as amostras como sendo os nós de um grafo, sendo os arcos definidos por uma relação de adjacência e ponderados por alguma métrica de distância aplicada em seus vetores de atributos, e diferem dos métodos tradicionais por não utilizar a ideia de geometria do espaço de atributos conseguindo, assim, melhores resultados em bases com *outliers* (amostras de uma determinada classe que estão presentes em uma região de outra classe no espaço de atributos) e sobreposição de classes. Duas abordagens supervisionadas já foram propostas atualmente, as quais diferem tanto na relação de adjacência e na função de valor de caminho utilizadas, quanto na maneira de encontrar os protótipos. A primeira delas utiliza como relação de adjacência o grafo completo e busca como protótipos as amostras que pertencem à intersecção entre as classes no conjunto de treinamento [28]. A outra metodologia desenvolvida utiliza um grafo  $k$ -NN (*k-Nearest Neighbor*, ou seja,  $k$ -vizinhos mais próximos) e encontra os protótipos como sendo os máximos regionais ou amostras de cada classe na junção entre as densidades [82, 83]. Será abordada aqui a primeira técnica desenvolvida, ou seja, aquela que faz uso do grafo completo.

### 3.4.1 Classificador por OPF

A técnica de classificação supervisionada baseada em florestas de caminhos ótimos apresentada neste capítulo modela as amostras como sendo os nós de um grafo completo. Os elementos mais representativos de cada classe do conjunto de treinamento, isto é, os protótipos, são escolhidos como sendo os elementos pertencentes às regiões de fronteira entre as classes. Os protótipos participam de um processo de competição disputando as outras amostras oferecendo-lhes caminhos de menor custo

e seus respectivos rótulos. Ao final deste processo, obtemos um conjunto de treinamento particionado em árvores de caminhos ótimos, sendo que a união das mesmas nos remete a uma floresta de caminhos ótimos. Esta abordagem apresenta vários benefícios com relação a outros métodos de classificação de padrões supervisionados: (i) é livre de parâmetros, (ii) possui tratamento nativo de problemas multiclases e (iii) não faz alusão sobre forma e/ou separabilidade das classes.

### 3.4.2 Fundamentação Teórica

Seja  $Z$  uma base de dados  $\lambda$ -rotulada e  $Z_1$  e  $Z_2$  os conjuntos de treinamento e teste, respectivamente, com  $|Z_1|$  e  $|Z_2|$  amostras, tal que  $Z = Z_1 \cup Z_2$ . Seja  $\lambda(s)$  uma função que associa o rótulo correto  $i$ ,  $i = 1, 2, \dots, c$  da classe  $i$  a qualquer amostra  $s \in Z_1 \cup Z_2$ . Seja  $S \in Z_1$  um conjunto de protótipos de todas as classes (isto é, amostras que melhor representam as classes). Seja  $\vec{o}$  um algoritmo que extrai  $n$  atributos de qualquer amostra  $s \in Z_1 \cup Z_2$ , e retorna um vetor de atributos  $\vec{o}(s) \in \mathfrak{R}^m$ . A distância  $d(s, t)$  entre duas amostras,  $s$  e  $t$ , é dada pela distância entre seus vetores de atributos  $\vec{o}(s)$  e  $\vec{o}(t)$ . Nosso problema consiste em usar  $S$ ,  $(\vec{o}, d)$  e  $Z_1$  para projetar um classificador ótimo, o qual pode prever o rótulo correto  $\lambda(s)$  de qualquer amostra  $s \in Z_2$ . Assim sendo, propomos um classificador que cria uma partição discreta ótima, a qual é uma floresta de caminhos ótimos computada em  $\mathfrak{R}^m$  pelo algoritmo da transformada imagem floresta [84].

Seja  $(Z_1, A)$  um grafo completo cujos nós são as amostras em  $Z_1$ , onde qualquer par de amostras define um arco em  $A$  (isto é,  $A = Z_1 \times Z_1$ ) (Figura 11a). Note que os arcos não precisam ser armazenados e o grafo não precisa ser explicitamente representado.

Temos, também, que um caminho em um grafo é uma seqüência de amostras  $\pi_{s_k} = \langle s_1, s_2, \dots, s_k \rangle$ , onde  $(s_i, s_{i+1}) \in A$  para  $1 \leq i \leq k - 1$ . Um caminho é dito ser trivial se  $\pi_s = \langle s \rangle$ . Nós associamos a cada caminho  $\pi_s$  um valor dado por uma função de valor de caminho  $f$ , denotada  $f(\pi_s)$ . Dizemos que um caminho  $\pi_s$  é ótimo se  $f(\pi_s) \leq f(\tau_s)$  para qualquer caminho  $\tau_s$ , onde  $\pi_s$  e  $\tau_s$  terminam na mesma amostra  $s$ , independente de sua origem. Também denotamos  $\pi_s \cdot \langle s, t \rangle$  a concatenação do caminho  $\pi_s$  com término em  $s$  e o arco  $(s, t)$ . O algoritmo da OPF pode ser utilizado com qualquer função de valor de caminho suave [84]. Uma função de valor de caminho  $f$  é suave quando, para qualquer amostra  $t$ , existe um caminho ótimo  $\pi_t$  o qual é trivial ou possui a forma  $\pi_s \cdot \langle s, t \rangle$ , onde

- $f(\pi_s) \leq f(\pi_t)$ ;
- $\pi_s$  é ótimo, e
- para qualquer caminho ótimo  $\tau_s$ ,  $f(\tau_s \cdot \langle s, t \rangle) = f(\pi_t)$ .

Na versão OPF com grafo completo a função de custo abordada foi a  $f_{max}$ , a qual é definida como:

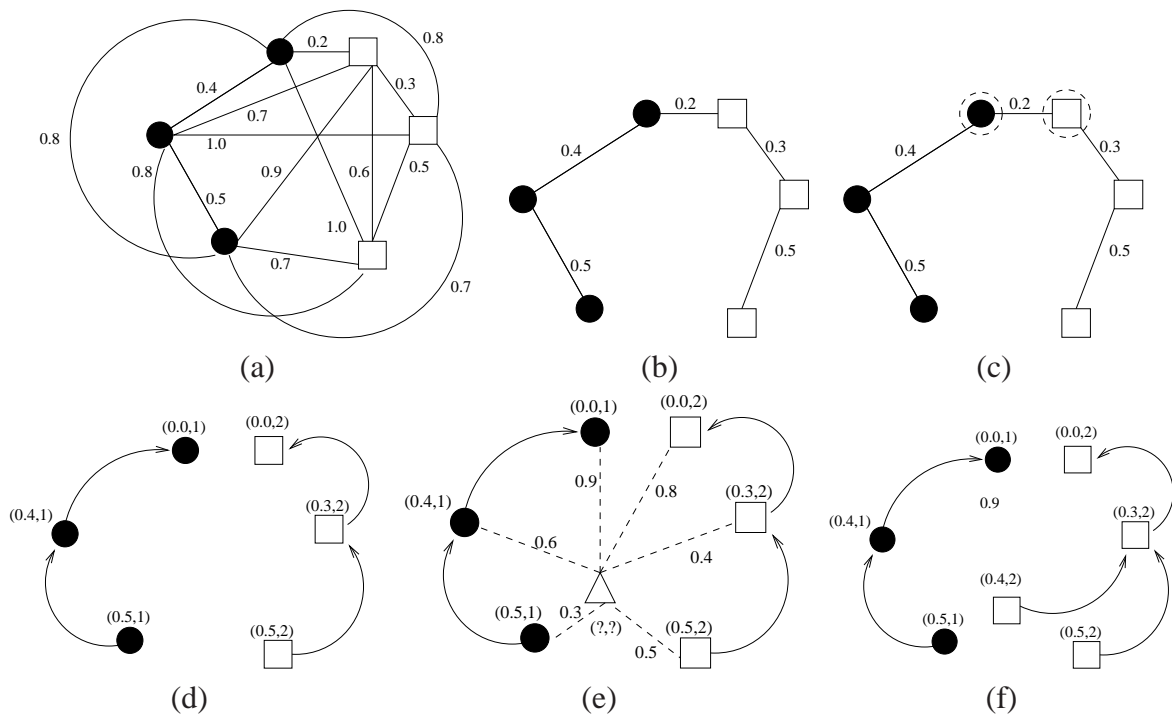


Figura 11: (a) Grafo completo ponderado nas arestas para um determinado conjunto de treinamento. (b) MST do grafo completo. (c) Protótipos escolhidos como sendo os elementos adjacentes de classes diferentes na MST (nós circulados). (d) Floresta de caminhos ótimos resultante para a função de valor de caminho  $f_{max}$  e dois protótipos. Os identificadores  $(x, y)$  acima dos nós são, respectivamente, o custo e o rótulo dos mesmos. A seta indica o nó predecessor no caminho ótimo. (e) Uma amostra de teste (triângulo) da classe 2 e suas conexões (linhas pontilhadas) com os nós do conjunto de treinamento. (f) O caminho ótimo do protótipo mais fortemente conexo, seu rótulo 2 e o custo de classificação 0.4 são associados a amostra de teste. Note que, mesmo a mostra de teste estando mais próxima de um nó da classe 1, ela foi classificada como sendo da classe 2. Extraído de [10].

$$\begin{aligned}
 f_{max}(\langle s \rangle) &= \begin{cases} 0 & \text{se } s \in S, \\ +\infty & \text{caso contrário} \end{cases} \\
 f_{max}(\pi_s \cdot \langle s, t \rangle) &= \max\{f_{max}(\pi_s), d(s, t)\},
 \end{aligned} \tag{3.45}$$

onde  $f_{max}(\pi_s)$  computa a distância máxima entre amostras adjacentes em  $\pi_s$ , quando  $\pi_s$  não é um caminho trivial.

O algoritmo baseado em OPF associa um caminho ótimo  $P^*(s)$  de  $S$  a toda amostra  $s \in Z_1$ , formando uma floresta de caminhos ótimos  $P$  (uma função sem ciclos, a qual associa a todo  $s \in Z_1$  seu predecessor  $P(s)$  em  $P^*(s)$ , ou uma marca *nil* quando  $s \in S$ , como mostrado na Figura 11d). Seja  $R(s) \in S$  a raiz de  $P^*(s)$  a qual pode ser alcançada por  $P(s)$ . O algoritmo computa, para cada

$s \in Z_1$ , o custo  $V(s)$  de  $P^*(s)$ , o rótulo  $L(s) = \lambda(R(s))$  e o seu predecessor  $P(s)$ , como segue.

**Algoritmo 2** – CLASSIFICADOR SUPERVISIONADO BASEADO EM FLORESTA DE CAMINHOS ÓTIMOS USANDO GRAFO COMPLETO

ENTRADA: Um conjunto de treinamento  $Z_1$   $\lambda$ -rotulado, protótipos  $S \subset Z_1$  e o par  $(v, d)$  para vetor de atributos e cálculo das distâncias.  
 SAÍDA: Floresta de caminhos ótimos  $P$ , mapa de valores de custo de caminhos  $V$  e mapa de rótulos  $L$   
 ESTRUTURAS AUXILIARES: Fila de prioridades  $Q$ , e variável  $cst$ .

1. **Para todo**  $s \in Z_1$ , **Faça**  $P(s) \leftarrow nil$  e  $V(s) \leftarrow +\infty$ .
2. **Para todo**  $s \in S$ , **Faça**
3.      $V(s) \leftarrow 0$ ,  $P(s) \leftarrow nil$ ,  $L(s) = \lambda(s)$
4.     Insira  $s$  em  $Q$ .
5. **Enquanto**  $Q$  é não vazia, **Faça**
6.     Remova de  $Q$  uma amostra  $s$  tal que  $V(s)$  é mínimo.
7.     **Para cada**  $t \in Z_1$  tal que  $s \neq t$  e  $V(t) > V(s)$ , **Faça**
8.         Calcule  $cst \leftarrow \max\{V(s), d(s, t)\}$ .
9.         **Se**  $cst < V(t)$ , **Então**
10.             **Se**  $V(t) \neq +\infty$ , **Então** remova  $t$  de  $Q$ .
11.              $P(t) \leftarrow s$ ,  $L(t) \leftarrow L(s)$  e  $V(t) \leftarrow cst$ .
12.             Insira  $t$  em  $Q$ .

As Linhas 1–4 inicializam os mapas e inserem protótipos em  $Q$ . O laço principal calcula um caminho ótimo de  $S$  para cada amostra  $s \in Z_1$  em uma ordem não decrescente de custos (Linhas 5–12). A cada iteração um caminho de custo ótimo  $V(s)$  é obtido em  $P$  (Linha 6). Empates são resolvidos em  $Q$  utilizando a política FIFO (first-in-first-out), ou seja, quando dois caminhos atingem uma determinada amostra  $s$  com o mesmo custo mínimo,  $s$  é associado ao primeiro caminho que o atingiu. O restante das linhas avalia se o caminho que atinge uma amostra adjacente  $t$  através de  $s$  é mais barato que o caminho que termina em  $t$ . Caso positivo, atualiza  $Q$ ,  $P(t)$ ,  $L(t)$  e  $V(t)$ . No final do algoritmo,  $V$  armazena o valor do custo do caminho ótimo de  $S$  a cada amostra  $s \in Z_1$  de acordo com  $f_{max}$ .

### 3.4.3 Treinamento

A fase de treinamento do classificador baseado em floresta de caminhos ótimos usando o grafo completo consiste, basicamente, em encontrar o conjunto  $S$  de protótipos, ou seja, os elementos mais representativos de cada classe e iniciar o processo de competição entre eles no conjunto de treinamento. Várias heurísticas poderiam ser adotadas como, por exemplo, uma escolha aleatória de protótipos. Entretanto, tal escolha pode prejudicar o desempenho do classificador, tornando-o



instável e com um alto grau de sensibilidade com relação aos protótipos escolhidos. Desejamos, assim, estimar protótipos nas regiões de sobreposição de amostras e nas fronteiras entre as classes, visto que são regiões muito susceptíveis a erros de classificação.

Computando uma Árvore de Espalhamento Mínima (*Minimum Spanning Tree - MST*) no grafo completo  $(Z_1, A)$ , obtemos um grafo conexo acíclico cujos nós são todas as amostras em  $Z_1$ , e os arcos são não direcionados e ponderados (Figura 11b). Seus pesos são dados pela distância  $d$  entre os vetores de atributos de amostras adjacentes. Esta árvore de espalhamento é ótima no sentido em que a soma dos pesos de seus arcos é mínima se comparada a outras árvores de espalhamento no grafo completo. Os protótipos a serem escolhidos são os elementos conectados na MST com diferentes rótulos em  $Z_1$ , isto é, elementos mais próximos de classes diferentes (Figura 11c). Removendo-se os arcos entre classes diferentes, tais amostras adjacentes tornam-se protótipos em  $S$  e o Algoritmo 2 pode computar uma floresta de caminhos ótimos em  $Z_1$  (Figura 11d). Note que uma dada classe pode ser representada por múltiplos protótipos (isto é, árvores de caminhos ótimos) e deve existir pelo menos um protótipo por classe.

#### 3.4.4 Classificação

Para qualquer amostra  $t \in Z_2$ , consideramos todos os arcos conectando  $t$  com amostras  $s \in Z_1$ , tornando  $t$  como se fosse parte do grafo original (ver Figura 11e, onde a amostra  $t$  é representada pelo triângulo no grafo). Considerando todos os possíveis caminhos entre  $S$  e  $t$ , desejamos encontrar o caminho ótimo  $P^*(t)$  de  $S$  até  $t$  com a classe  $\lambda(R(t))$  de seu protótipo  $R(t) \in S$  mais fortemente conexo. Este caminho pode ser identificado incrementalmente, avaliando o valor do custo ótimo  $V(t)$  como segue:

$$V(t) = \min\{\max\{V(s), d(s, t)\}\}. \forall s \in Z_1. \quad (3.46)$$

Seja  $s^* \in Z_1$  o nó que satisfaz a equação acima (isto é, o predecessor  $P(t)$  no caminho ótimo  $P^*(t)$ ). Dado que  $L(s^*) = \lambda(R(t))$ , a classificação simplesmente associa  $L(s^*)$  como a classe de  $t$  (Figura 11f). Um erro ocorre quando  $L(s^*) \neq \lambda(t)$ .

# Capítulo 4

## Algoritmos de Otimização

Neste capítulo serão abordados 3 Algoritmos de Otimização. Inicialmente será detalhado a Otimização por Enxame de Partículas (PSO-Particle Swarm Optimization) a qual procura encontrar uma solução ótima baseado na dinâmica do comportamento. Em seguida será visto o Algoritmo de Busca Harmônica (Harmony Search - HS) o qual tenta encontrar a solução ótima baseado no processo de improvisação de músicos. Por último, será relatado o Algoritmo de Busca Gravitacional (Gravitational Search Algorithm - GSA) o qual baseado na lei da gravitação universal de Newton, procura encontrar uma solução ótima.

### 4.1 Otimização por Enxame de Partículas

Recentemente, várias aplicações tem utilizado técnicas evolucionárias como métodos heurísticos para encontrar soluções ótimas ou pseudo-ótimas. Uma atenção particular tem sido dada à técnica denominada Otimização por Enxame de Partículas (*Particle Swarm Optimization - PSO*), devido à sua simplicidade e eficácia. Basicamente, PSO é um algoritmo modelado em inteligência coletiva que encontra uma solução em um espaço de busca com base na dinâmica do comportamento social [85]. Cada possível solução do problema é modelada como uma partícula do enxame que imita seu vizinho baseando-se em uma função objetivo. Outras definições consideram PSO como um algoritmo de pesquisa baseado em processos estocásticos e populacionais, onde a aprendizagem do comportamento social permite a cada solução possível (partícula) “voar” dentro desse espaço (enxame) a procura de outras partículas que possuam melhores características e, assim, maximizando a função objetivo.

Cada partícula tem uma memória para armazenar sua melhor solução local (máximos locais) e a melhor solução global (máximos globais). Levando-se em conta essas informações, cada partícula tem a capacidade de imitar as outras que proporcionam a ela melhores posições no enxame. Este processo simula a interação social entre um bando de aves a procura de comida, por exemplo. Esse

mecanismo sócio-recognitivo pode ser resumido em três grandes princípios [85]: (i) avaliação, (ii) comparação e (iii) imitação. Cada partícula pode avaliar outras dentro de sua vizinhança através de alguma função objetivo, pode compará-la com seu próprio valor e, finalmente, pode decidir se é uma boa escolha imitá-la. O enxame é modelado como sendo um espaço multidimensional  $\mathfrak{R}^m$ , em que cada partícula  $p_i = (x_i, v_i) \in \mathfrak{R}^m$  tem duas características principais: (i) sua posição  $x_i$  e (ii) velocidade  $v_i$ . A melhor solução (posição no enxame) local  $\hat{x}_i$  e global  $\hat{s}$  são também conhecidas (Figura 12).

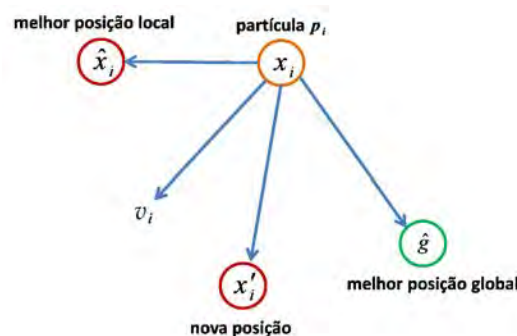


Figura 12: Interpretação geométrica do PSO.

Após a definição do tamanho do enxame, ou seja, o número de partículas, cada uma delas é inicializada com valores aleatórios de velocidade e posição. Cada indivíduo é então avaliado com relação a alguma função objetivo e seu máximo local é atualizado. No final, o valor máximo global é atualizado com a partícula que alcançou a melhor posição no enxame. Este processo é repetido até que algum critério de convergência seja atingido. A posição atualizada e equações da velocidade da partícula  $p_i$  da forma mais simples que governam o PSO são, respectivamente, dadas por

$$v_i = wv_i + c_1r_1(\hat{x}_i - x_i) + c_2r_2(\hat{s} - x_i) \quad (4.1)$$

e

$$x_i = x_i + v_i, \quad (4.2)$$

onde  $w$  é a força de inércia que controla o poder de interação entre as partículas, e  $r_1, r_2 \in [0, 1]$  são variáveis aleatórias que dão a idéia de estocasticidade ao método PSO. As constantes  $c_1$  e  $c_2$  são também utilizadas para guiar as partículas e definidas como parâmetros de entrada para o algoritmo.

## 4.2 Busca Harmônica

A Busca Harmônica (*Harmony Search - HS*) é um algoritmo evolutivo inspirado na música, considerando o processo de improvisação de artistas relacionados a área musical [86]. O HS é um conceito

simples, com poucos parâmetros e rápido em sua execução. A idéia principal é utilizar o mesmo processo adotado pelos músicos para criar novas canções com o intuito de obter uma solução ótima em algum processo de otimização. Basicamente, qualquer solução possível é modelada como uma harmonia e cada parâmetro a ser otimizado pode ser visto como uma nota musical. A melhor harmonia (solução) é escolhida como aquela que maximiza alguns critérios de otimização. O algoritmo é composto por algumas etapas, como descrito abaixo:

- Passo 1: Inicializa o problema de otimização e parâmetros do algoritmo;
- Passo 2: Inicializa a Memória Harmônica (MH);
- Passo 3: Improvisa uma nova harmonia para MH;
- Passo 4: Atualiza a MH se a nova harmonia é melhor do que a pior harmonia na MH, inclui a nova harmonia na MH, e remove a pior da MH, e
- Passo 5: Se o critério de parada não for satisfatório, retorna para o Passo 3.

A seguir, discutiremos cada uma das etapas acima mencionadas.

### 4.2.1 O Problema da Otimização e os Parâmetros do Algoritmo

Para descrever como funciona a Busca Harmônica, um problema de otimização é especificado na Etapa 1, conforme segue:

$$\text{Minimizar } f(x^i) \text{ sujeito a } x^i \in X_i, \forall i = 1, 2, \dots, M, \quad (4.3)$$

onde  $f(x^i)$  é a função objetivo,  $x^i$  e  $X_i$ , significam, respectivamente, a variável de projeto  $i$  e seu conjunto de valores possíveis, e  $M$  é o número de variáveis do projeto.

Os parâmetros do algoritmo HS necessários para resolver o problema de otimização (Equação 4.3) também são especificados nessa etapa: o tamanho da memória harmônica (*Harmony Memory Size - HMS*), a taxa de memória harmônica considerada (*Harmony Memory Considering Rate - HMCR*), a taxa de passo de ajuste (*Pitch Adjusting Rate - PAR*), e o critério de parada. Os parâmetros HMCR e PAR são usados para melhorar o vetor solução, ou seja, eles podem ajudar o algoritmo a encontrar soluções a nível global e local. Em nosso texto, temos que  $N = HMS$ .

### 4.2.2 Memória Harmônica

No passo 2, a matriz MH é inicializada com vetores solução gerados aleatoriamente com seus respectivos valores da função objetivo:

$$MH = \left[ \begin{array}{cccc|c} x_1^1 & x_2^1 & \dots & x_M^1 & f(x^1) \\ x_1^2 & x_2^2 & \dots & x_M^2 & f(x^2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_1^{HMS} & x_2^{HMS} & \dots & x_M^{HMS} & f(x^{HMS}) \end{array} \right], \quad (4.4)$$

onde  $x_i^j$  denota a variável de decisão  $i$  da harmonia  $j$ .

### 4.2.3 Gerando uma Nova Harmonia

Na Etapa 3, um novo vetor de harmonia  $x' = (x'_1, x'_2, \dots, x'_M)$  é gerado com base nas considerações da memória e randomização (improvisação musical), como descrito a seguir:

$$x' \leftarrow \begin{cases} x'_i \in \{x_i^1, x_i^2, \dots, x_i^{HMS}\} & \text{com probabilidade HCMR,} \\ x'_i \in X & \text{com probabilidade (1-HCMR).} \end{cases} \quad (4.5)$$

Assim, temos que HMCER é a probabilidade de escolher um valor a partir de valores armazenados na MH, e (1-HMCER) é a probabilidade de escolher aleatoriamente um valor viável não se limitando aos armazenados na MH.

Além disso, cada componente  $i$  do novo vetor de harmonia  $x'$  é examinado para determinar se ele deve ser ajustado:

$$x'_i \leftarrow \begin{cases} \mathbf{Sim} & \text{com probabilidade PAR,} \\ \mathbf{Não} & \text{com probabilidade (1-PAR).} \end{cases} \quad (4.6)$$

O ajuste de cada variável de decisão é frequentemente utilizado com o intuito de melhorar as soluções e evitar ótimos locais. Trata-se de um mecanismo que muda os valores vizinhos de algumas variáveis de decisão na harmonia. Assim, se a decisão de ajuste para a variável de decisão  $x'_i$  for "Sim",  $x'_i$  é substituído da seguinte forma:

$$x'_i \leftarrow x'_i + rb, \quad (4.7)$$

onde  $b$  é uma distância arbitrária, e  $r$  é uma distribuição uniforme em  $[0, 11$ .

#### 4.2.4 Atualizando a Memória Harmônica

Na Etapa 4, se a nova harmonia for melhor do que a pior harmonia na MH, este último é substituído por esta nova harmonia. Note que a melhor harmonia é definida como aquela que maximiza a função objetivo.

#### 4.2.5 Critério de Parada

Na etapa 5, o algoritmo de Busca Harmônica finaliza sua execução quando ele satisfaz o critério de parada. Caso contrário, as Etapas 3 e 4 são repetidas a fim de improvisar novamente uma nova harmonia.

### 4.3 Algoritmo de Busca Gravitacional

A gravidade é uma das quatro interações fundamentais da natureza, juntamente com a força forte, o eletromagnetismo e a força nuclear fraca. A idéia que diz respeito às regras de gravidade é o fato de que um objeto com massa atrai outro. Uma das teorias mais aceitas é a lei da gravitação universal de Newton, que diz que “todas as partículas no universo atraem outras com uma força que é diretamente proporcional ao produto das suas massas e inversamente proporcional ao quadrado da distância entre elas” [87], ou seja:

$$F = G \frac{M_1 M_2}{R^2}, \quad (4.8)$$

onde  $M_1$  e  $M_2$  correspondem às massas de duas partículas,  $R$  é a distância entre elas,  $G$  é uma constante gravitacional e  $F$  é a magnitude da força gravitacional. A “constante gravitacional”  $G$  é dependente do tempo, e diminui com a idade do universo:

$$G(t) = G(t_0) \frac{t_0^\beta}{t}, \quad \beta < 1, \quad (4.9)$$

em que  $G(t)$  é o valor da constante gravitacional no tempo  $t$  e  $G(t_0)$  é o valor da constante gravitacional no momento da “criação do universo” que está sendo considerado.

A segunda lei de Newton diz que quando uma força  $F$  é aplicada a uma massa, a sua aceleração  $a$  depende somente dessa força e da sua massa  $M$ :

$$a = \frac{F}{M}. \quad (4.10)$$

Com base nas afirmações acima, Rashedi *et al.* [88] propuseram o Algoritmo de Busca Gra-

vitacional (*Gravitational Search Algorithm - GSA*), que pode ser definido como segue. Seja  $X = \{x_1, x_2, \dots, x_n\}$  um universo com  $n$  massas, de modo que  $x_i \in \mathfrak{R}^m$ . Pode-se definir, em um momento específico  $t$ , a força que age sobre a massa  $i$  a partir da massa  $j$  na dimensão  $d$  da seguinte forma:

$$F_{ij}^d(t) = G(t) \frac{M_i(t)M_j(t)}{R_{ij}(t) + \epsilon} (x_j^d(t) - x_i^d(t)), \quad (4.11)$$

onde  $R_{ij}(t)$  é a distância euclidiana entre as massas  $i$  e  $j$ , e  $\epsilon$  é uma pequena constante.

A fim de dar um comportamento estocástico para o GSA, Rashedi *et al.* [88] assumem a força total que age sobre a massa  $i$  em uma dimensão  $d$  como uma soma aleatória ponderada das forças exercidas de outros agentes:

$$F_i^d(t) = \sum_{j=1, j \neq i}^m \gamma_j F_{ij}^d(t), \quad (4.12)$$

onde  $\gamma_j$  denota um número gerado aleatoriamente entre 0 e 1. A aceleração da massa  $i$  em tempo  $t$  e dimensão  $d$  é dada por:

$$a_i^d(t) = \frac{F_i^d(t)}{M_i(t)}, \quad (4.13)$$

em que a massa  $M_i$  é calculada do seguinte modo:

$$M_i(t) = \frac{q_i(t)}{\sum_{j=1}^m q_j(t)}, \quad (4.14)$$

com

$$q_i(t) = \frac{C_i(t) - w(t)}{b(t) - w(t)}. \quad (4.15)$$

Os termos  $w(t)$  e  $b(t)$  significam, respectivamente, as massas com melhor e pior valor de função objetivo. O termo  $C_i(t)$  denota o vetor de características da massa  $i$ , ou seja, o conjunto de variáveis a serem otimizadas, as quais estão representadas pela massa  $i$ .

Finalmente, para evitar soluções ótimas locais, somente as  $k$  melhores massas, ou seja, aquelas com maiores valores de função objetivo, vão atrair as outras. Seja  $\mathcal{K}$  um conjunto dessas massas. O valor de  $k$  é definido como  $k_0$  no início do algoritmo e diminui com o tempo. Assim, a Equação 4.12 é reescrita como

$$F_i^d(t) = \sum_{j \in \mathcal{K}, j \neq i} \gamma_j F_{ij}^d(t). \quad (4.16)$$

A velocidade e a posição de atualização das massas são dadas por:

$$v_i^d(t+1) = \gamma_j v_i^d(t) + a_i^d(t) \quad (4.17)$$

e

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1). \quad (4.18)$$



## Capítulo 5

# Seleção de Características

Neste capítulo será detalhada a importância da seleção de características e apresentadas três propostas de algoritmos híbridos que utilizam a técnica OPF em conjunto com uma técnica de otimização com o intuito de alcançar a melhor acurácia com o menor conjunto de características possível.

Muitas aplicações exigem um grande número de características a serem extraídas devido, principalmente, à natureza do problema. É comum encontrarmos na literatura abordagens que empregam centenas de informações com o intuito de descrever melhor as amostras do conjunto de dados, sendo que muitas características são extremamente custosas de serem extraídas. Ademais, não existe uma garantia que, aumentando a dimensão do espaço de características, estaremos melhorando a sua separabilidade.

Nesse contexto, métodos para seleção de características estão sendo empregados com o intuito de identificarmos as informações mais relevantes e descartar as demais. Tais abordagens permitem, em muitos casos, aumentar a sensibilidade do sistema, bem como acelerar os processos de extração de informação e classificação e treinamento dos dados. Assim sendo, podemos tratar a questão de seleção de características como sendo um problema de otimização, sendo que as variáveis de decisão, ou seja, as variáveis a serem otimizadas correspondem às características, e a função objetivo a ser maximizada seria a taxa de acerto do classificador em um conjunto de dados.

Desta forma, fazemos uso de um terceiro conjunto de dados, denominado conjunto de avaliação, o qual irá guiar o algoritmo de otimização a encontrar as características mais relevantes para o problema em questão. Diferentemente do descrito na Seção 3.4.2, a base de dados  $Z$  agora será descrita como  $Z = Z_1 \cup Z_2 \cup Z_3$ , sendo que  $Z_1$  e  $Z_2$  denotam os conjuntos de treinamento e avaliação, respectivamente, e  $Z_3$  corresponde ao conjunto de teste. Nesse caso, o subconjunto de características mais relevantes é projetado utilizando os conjuntos  $Z_1$  e  $Z_2$ , pois o classificador será treinado em  $Z_1$  e a sua taxa de acerto em  $Z_2$  será utilizada como valor de função objetivo para o algoritmo de otimização. Finalmente, a taxa de acerto real será calculada no conjunto de teste  $Z_3$ , o qual não participou

da etapa de aprendizado de seleção das características. Dentre as possíveis técnicas de otimização, o presente trabalho aborda algumas técnicas baseadas em comportamentos e dinâmicas sociais, bem como outras baseadas em fenômenos naturais.

## 5.1 Abordagem PSO-OPF Proposta

Nesse trabalho, propomos um algoritmo híbrido para a seleção de características baseado em PSO e OPF, o qual chamamos PSO-OPF. Nesse contexto, cada possível solução (partícula) no enxame será composta por um conjunto de bits denotando que a característica  $i$ ,  $\forall i = 1, 2, \dots, m$ , deverá fazer parte ou não do conjunto final. Para cada partícula, o classificador de padrões deverá ser treinado e posteriormente avaliado no conjunto de avaliação. Dado que o classificador Floresta de Caminhos Ótimos tem demonstrado ser preciso e muito rápido na etapa de treinamento dos dados [10], a combinação de PSO e OPF pode fornecer uma solução rápida e robusta para a seleção de características, especialmente em grandes conjuntos de dados, os quais são muito comuns no contexto abordado neste trabalho.

A abordagem PSO descrita na Seção 4.1 descreve um algoritmo de otimização para o caso de variáveis contínuas, sendo que o problema em questão é tratado em um reticulado booleano, ou seja, as características a serem otimizadas são valores binários. Nesse contexto, Kennedy e Eberhart [89] propuseram uma versão discreta do PSO, chamada de *Binary PSO*, a qual foi posteriormente empregada para fins de seleção de características por Hiram e Goodman [90]. A ideia consiste, basicamente, em modificar as equações de atualização de posição das partículas, sendo que a formulação para o cálculo da velocidade (Equação 4.1) se mantém. Assim sendo, a Equação 4.2 pode ser descrita como:

$$x_i \leftarrow \begin{cases} 1 & \text{se } \gamma < S(v_i) \\ 0 & \text{caso contrário,} \end{cases} \quad (5.1)$$

onde  $\gamma$  corresponde a um valor aleatório entre 0 e 1. Temos que  $S(v_i)$  é dado por:

$$S(v_i) = \frac{1}{1 + e^{-v_i}}. \quad (5.2)$$

Este procedimento é apresentado pelo Algoritmo 3. O primeiro laço nas Linhas 1 – 6 inicializa o enxame com valores binários para a posição de cada partícula (Linhas 2 – 3), que corresponde se uma característica será selecionada ou não para compor o novo conjunto de dados (a função  $R\{0, 1\}$  na linha 3 gera um valor aleatório entre 0 e 1). As Linhas 8 – 15 compõe a nova formação e apresenta a avaliação com as características selecionadas e avalia cada partícula, a fim de atualizar sua função

objetivo (Linhas 12 – 15). As Linhas 16 – 20 encontra a melhor solução do enxame e, finalmente, as Linhas 21 – 30 atualiza a posição de cada partícula de acordo com (4.1) (Linha 26). Como estamos trabalhando com um espaço de busca binária, a posição da nova partícula só pode assumir valores binários. Assim, a estrutura condicional na Linha 28 foi idêntico ao proposto por Kennedy e Eberhart [89], que apresentaram uma versão binária do PSO, ou seja, o BPSO (*Binary PSO*).

## 5.2 Abordagem BH-OPF Proposta

O algoritmo de BH é uma abordagem interessante para diversas aplicações, principalmente devido à sua simplicidade e baixo custo computacional. Como o número de parâmetros a ser otimizado em um subconjunto ótimo de recursos do algoritmo de busca é geralmente alto, muitas vezes é desejável a utilização de técnicas rápidas. Em nossa abordagem, propomos a aplicação de OPF para calcular o valor da função objetivo de cada harmonia na MH.

Dado que o classificador Floresta de Caminhos Ótimos tem demonstrado ser preciso e muito rápido para os padrões de treinamento [10], acreditamos que a combinação de BH e OPF pode fornecer uma solução rápida e robusta para a seleção de recursos, especialmente em grandes conjuntos de dados, que são muito comuns no contexto abordado neste trabalho.

O algoritmo híbrido proposto neste trabalho utiliza a precisão de OPF para guiar o algoritmo de BH para encontrar uma solução adequada. Basicamente, para cada harmonia, o classificador OPF é treinado com o grupo correspondente de recursos em um conjunto de treinamento, e sua precisão é avaliada sobre um conjunto de avaliação. Este procedimento é apresentado pelo Algoritmo 4.

O loop inicial nas Linhas de 1 – 6 é responsável pela inicialização da memória harmônica. Na Linha 3, cada posição da memória harmônica  $HM_{i,j}$  recebe um valor binário que indica se esse recurso será selecionado ou não. Linhas de 4 – 5 criam um novo conjunto de treinamento e de avaliação apenas com as características previamente selecionadas, ou seja,  $HM_{i,j} = 1$ ,  $i = 1, 2, \dots, HMS$  e  $j = 1, 2, \dots, F$ . Na Linha 6, o classificador OPF é treinado e avaliado com esses novos conjuntos e sua acurácia sobre o conjunto de avaliação é armazenado em  $HM_{i,F+1}$ . A Linha 8 calcula o número de recursos que serão selecionados a partir da memória harmônica para compor a nova harmonia  $H$ . O laço nas Linhas 9 – 10 seleciona os valores da memória harmônica para construir o vetor da nova harmonia  $H$ . A função  $n(HM_{i,j})$  retorna o valor mais recorrente para o recurso  $j$ . Além disso, o próximo ciclo nas Linhas 11 – 12 termina a etapa de consideração da memória, atribuindo valores aleatórios  $x \in \{0, 1\}$  para as posições remanescentes de  $H$ . Lembre-se que as Linhas 8 – 12 referem-se a Equação 4.5.

A Linha 13 localiza o índice da harmonia  $i$  em  $HM_{i,F+1}$ , que obtiveram classificação inferior a acurácia sobre  $Z'_2$ , e substitui toda a harmonia  $HM_i$  por um novo  $H$  obtido se  $HM_{i,F+1}$  for menor

**Algoritmo 3** – ALGORITMO PSO-OPF

ENTRADA: Conjuntos de treinamento  $Z_1$  e de avaliação  $Z_2$   $\lambda$ -rotulados, valor do peso de inércia  $w$ , tamanho do exame  $n$ , número de características  $m$ , número de interações  $T$ , valores  $c_1$  e  $c_2$ .

SAÍDA: Melhor posição global  $\hat{g}$ .

ESTRUTURAS AUXILIARES: Vetor  $l$  com tamanho  $n$  e variáveis  $r_1, r_2, k, acc, maxfit, globalfit, maxindex$  e  $\gamma$ .

1. Para cada partícula  $p_i$  ( $\forall i = 1, \dots, n$ ), faça.
2.     Para cada dimensão  $j$  ( $\forall j = 1, \dots, m$ ), faça.
3.          $x_{i,j} \leftarrow R\{0, 1\}$ .
4.      $v_{i,j} \leftarrow 0$
5.      $l_i \leftarrow -\infty$
6.  $globalfit \leftarrow -\infty$
7. Para cada interação  $t$  ( $t = 1, \dots, T$ ), faça.
8.     Para cada partícula  $p_i$  ( $\forall i = 1, \dots, n$ ), faça.
9.         Criar  $Z'_1$  e  $Z'_2$  de  $Z_1$  e  $Z_2$ , respectivamente, de tal forma que ambos tenham somente funções  $p_i$
10.         tal que  $x_{i,j} \neq 0, \forall j = 1, \dots, m$ .
11.         Treinar OPF em  $Z'_1$ , avaliar  $Z'_2$  e armazenar a acurácia em  $acc$ .
12.         Se ( $acc > l_i$ ), então.
13.              $l_i \leftarrow acc$
14.             Para cada dimensão  $j$  ( $\forall j = 1, \dots, m$ ), faça.
15.                  $\hat{x}_{i,j} \leftarrow x_{i,j}$
16.      $[maxfit, maxindex] \leftarrow \max(f)$
17.     Se ( $maxfit > globalfit$ ), então.
18.          $globalfit \leftarrow maxfit$
19.         Para cada dimensão  $j$  ( $\forall j = 1, \dots, m$ ), faça.
20.              $\hat{g}_j \leftarrow x_{maxindex,j}$
21.     Para cada partícula  $p_i$  ( $\forall i = 1, \dots, n$ ), faça.
22.         Para cada dimensão  $j$  ( $\forall j = 1, \dots, m$ ), faça.
23.              $r_1 \leftarrow R\{0, 1\}$ .
24.              $r_2 \leftarrow R\{0, 1\}$ .
25.              $k \leftarrow c_2 * r_2 * (\hat{g}_j - x_{i,j})$
26.              $v_{i,j} \leftarrow w * v_{i,j} + c_1 * r_1 * (\hat{x}_{i,j} - x_{i,j}) + k$
27.              $\gamma \leftarrow$  valor aleatório uniforme
28.             Se ( $\gamma < \frac{1}{1+e^{-v_{i,j}}}$ ), então
29.                  $x_{i,j} \leftarrow 1$
30.             senão  $x_{i,j} \leftarrow 0$

**Algoritmo 4** – ALGORITMO BH-OPF

ENTRADA: Conjuntos de treinamento  $Z_1$  e de avaliação  $Z_2$   $\lambda$ -rotulados, Tamanho da Memória Harmônica  $HMS$ , Número de Recursos  $F$ , Memória Harmônica considerando a taxa de  $hmcr$  e número de interações  $T$  por convergência.

SAÍDA: Vetor harmônico  $H^*$  de tamanho  $F$  que dá a maior precisão de classificação para  $Z_2$ .

ESTRUTURAS AUXILIARES: Memória Harmônica  $HM$  de tamanho  $HMS \times F + 1$ , novo vetor harmônico  $H$  de tamanho  $F + 1$  e variável  $tmp$ .

1. Para cada  $i \leftarrow 1$  até  $HMS$ , faça
2.     Para cada  $j \leftarrow 1$  até  $F$ , faça
3.      $\perp$   $HM_{i,j} \leftarrow \text{Randomiza}\{0,1\}$ .
4.     Criar  $Z'_1$  e  $Z'_2$  de  $Z_1$  e  $Z_2$ , respectivamente, tais que ambos contém
5.     características  $HM_{i,j} = 1, j = 1, 2, \dots, F$ .
6.     Treinar OPF em  $Z'_1$ , avaliar em  $Z'_2$  e armazena a acurácia em  $HM_{i,F+1}$ .
7. Para cada  $k \leftarrow 1$  até  $T$ , faça
8.      $tmp \leftarrow \lfloor hmcr * F \rfloor$ .
9.     Para cada  $j \leftarrow 1$  até  $tmp$ , faça
10.      $\perp$   $H_j \leftarrow n(HM_{ij}), i = 1, 2, \dots, HMS$ .
11.     Para cada  $j \leftarrow tmp + 1$  até  $F$ , faça
12.      $\perp$   $H_j \leftarrow \text{Randomiza}\{0,1\}$ .
13.     Encontrar  $i$  tal que  $HM_{i,F+1}$  é mínimo.
14.     Se  $(H_{F+1} > HM_{i,F+1})$ , então
15.      $\perp$  Para cada  $j \leftarrow 1$  até  $F + 1$ , faça
16.      $\perp$   $\perp$   $HM_{i,j} \leftarrow H_j$ .
17.     Criar  $Z'_1$  e  $Z'_2$  de  $Z_1$  e  $Z_2$ , respectivamente, tais que ambos contém
18.     características  $HM_{i,j} = 1, j = 1, 2, \dots, F$ .
19.     Treinar OPF com  $Z'_1$ , avalia  $Z'_2$  e armazenar a acurácia em  $HM_{i,F+1}$ .
20.     Para cada  $i \leftarrow 1$  até  $HMS$ , faça
21.      $\perp$  Encontrar  $i$  tal que  $HM_{i,F+1}$  é máximo.
22. Para cada  $j \leftarrow 1$  até  $F$ , faça
23.      $\perp$   $H_j^* \leftarrow HM_{i,j}$ .
24. Retornar  $[H^*]$

do que aquela dada por  $H_{F+1}$  (Linhas 14 – 16). As Linhas 17 – 19 são responsáveis por atualizar os valores de ajuste da memória harmônica. Finalmente, o laço nas Linhas 20 – 21 encontra a harmonia na memória harmônica que dá a maior acurácia de classificação do OPF sobre  $Z'_2$  e armazena em seu vetor de melhor harmonia  $H^*$  nas Linhas 22 – 23. Neste algoritmo, a etapa de ajuste do passo não foi aplicada porque o intervalo de valores das variáveis de decisão é composto apenas por valores binários.

### 5.3 Abordagem GSA-OPF Proposta

O algoritmo para a seleção de recurso proposto neste trabalho funciona de forma semelhante como as anteriores: a idéia principal é a utilização de um classificador de acurácia sobre um conjunto de avaliação como a função objetivo para orientar o GSA em busca das melhores soluções. Porém, a abordagem GSA descrita na Seção 4.3 não pode ser aplicada em problemas de otimização que requerem espaços binários (Equação 4.18). Portanto, Rashedi *et al.* [91] propuseram o GSA binário, que possui uma equação diferente para atualizar a posição de cada massa:

$$x_i^d(t+1) = \begin{cases} 1 - x_i^d(t) & \text{se } \gamma_i < S(v_i^d(t+1)) \\ x_i^d(t) & \text{caso contrário,} \end{cases} \quad (5.3)$$

tal que

$$S(v_i^d(t)) = |\tanh(v_i^d(t))|. \quad (5.4)$$

A fim de alcançar uma boa taxa de convergência, Rashedi *et al.* [91] propuseram limitar a velocidade para  $|v_i^d| < v_{max} = 6$ . Segundo eles, esse valor foi obtido através de algumas experiências. O algoritmo 5 implementa essa idéia.

O primeiro loop nas Linhas de 1 – 5 inicializa o universo das massas com valores binários para a posição de cada partícula (Linhas 2 – 3), que corresponde se um recurso será selecionado ou não para compor o novo conjunto de dados. Valores de velocidade e ajuste também são inicializados com valores 0 e  $-\infty$ , respectivamente. Linhas 9 – 12 compõem a nova formação e apresentam a avaliação com as características selecionadas e avaliando cada partícula de massa a fim de atualizar seu valor de ajuste (Linha 12). Linhas 13 e 14 encontram os índices das massas que dão os melhores e piores valores de ajuste, respectivamente. O circuito interno nas Linhas 16 – 18 computa  $q_i$  de acordo com a Equação 4.15. A massa de cada partícula é calculada nas Linhas 19 – 20, como descrito na Equação 4.14.

A Linha 21 classifica o universo das massas de acordo com a ordem decrescente de seus valores de ajuste e armazena na variável  $M'$ . A cópia é armazenada em  $x'$  (Linha 22). O laço nas Linhas 23 – 29 calcula a aceleração de cada massa de acordo com a Equação 4.13 (Linha 29). O loop interno de

**Algoritmo 5** – ALGORITMO GSA-OPF

ENTRADA: Conjuntos de treinamento  $Z_1$  e de avaliação  $Z_2$ , constante gravitacional inicial  $G_0$ , número de massas  $m$ , dimensão  $n$ , número de interações  $T$  e  $k$  e valores  $\epsilon$ .

SAÍDA: Subconjunto de características  $H^*$  que dá o máximo de precisão sobre  $Z_2$ .

ESTRUTURAS AUXILIARES: Vetor de ajuste  $c$  com tamanho  $m$ , vetores auxiliares  $q$ ,  $\hat{F}$ ,  $M'$  e  $x'$  de tamanho  $m$ , vetor de massa  $M$  de tamanho  $m$ , matrizes  $F$  e  $a$  de tamanho  $m \times n$ , conjuntos  $Z'_1$  e  $Z'_2$  e variáveis  $G$ ,  $bindex$ ,  $windex$ ,  $i$ ,  $j$ ,  $z$ ,  $R$  e  $p$ .

1. Para cada partícula de massa  $p_i$  ( $\forall i = 1, \dots, n$ ), faça.
2.     Para cada dimensão  $j$  ( $\forall j = 1, \dots, m$ ), faça.
3.          $x_{i,j} \leftarrow R\{0, 1\}$ .
4.          $v_{i,j} \leftarrow 0$ .
5.          $f_i \leftarrow -\infty$ .
6.  $G \leftarrow G_0$ .
7. Para cada iteração  $t$  ( $t = 1, \dots, T$ ), faça.
8.      $G \leftarrow G * (1 - \frac{t}{T})$ .
9.     Para cada partícula de massa  $p_i$  ( $\forall i = 1, \dots, n$ ), faça.
10.         Criar  $Z'_1$  e  $Z'_2$  de  $Z_1$  e  $Z_2$ , respectivamente, de tal forma que ambos contêm somente a função  $p_i$ ,
11.         tal que  $x_{i,j} \neq 0, \forall j = 1, \dots, m$ .
12.         Treinar classificador OPF em  $Z'_1$ , avaliar  $Z'_2$  e armazena a acurácia em  $c_i$ .
13.      $bindex \leftarrow b(p)$ .
14.      $windex \leftarrow w(p)$ .
15.      $tmp \leftarrow 0$ .
16.     Para cada partícula de massa  $p_i$  ( $\forall i = 1, \dots, n$ ), faça.
17.          $q_i \leftarrow \frac{c_i - c_{windex}}{c_{bindex} - c_{windex}}$ .
18.          $tmp \leftarrow tmp + q_i$ .
19.     Para cada partícula de massa  $p_i$  ( $\forall i = 1, \dots, n$ ), faça.
20.          $M_i \leftarrow \frac{q_i}{tmp}$ .
21.     Classifica massas em ordem decrescente de ajuste e armazena em  $M'$ .
22.      $x' \leftarrow M'$ .
23.     Para cada partícula de massa  $p_i$  ( $\forall i = 1, \dots, n$ ), faça.
24.         Para cada dimensão  $j$  ( $\forall j = 1, \dots, m$ ), faça.
25.              $\hat{F}_i \leftarrow 0$ .
26.             Para cada  $z$  de 1 até  $k$ , faça.
27.                  $F_{ij} \leftarrow G * \frac{M_i * M_z}{R + \epsilon} * (x_{ij} - x'_{zj})$ .
28.                  $\hat{F}_i \leftarrow \gamma_i * F_{ij}$ .
29.              $a_{ij} \leftarrow \frac{\hat{F}_i}{M_i}$ .
30.      $k \leftarrow k - 1$ .
31.     Para cada partícula de massa  $p_i$  ( $\forall i = 1, \dots, n$ ), faça.
32.         Para cada dimensão  $j$  ( $\forall j = 1, \dots, m$ ), faça.
33.              $v_{ij} \leftarrow \gamma_j * v_{ij} + a_{ij}$ .
34.             Se  $(|v_{ij}| > 6)$ , então
35.                 Se  $(v_{ij} < 0)$ , então  $v_{ij} \leftarrow -6$ .
36.                 Senão  $v_{ij} \leftarrow 6$ .
37.             Se  $(\gamma_j^* < |\tanh(v_{ij})|)$ , então
38.                  $x_{ij} \leftarrow 1 - x_{ij}$ .
39.      $tmp \leftarrow -\infty$ .
40.     Para cada partícula de massa  $p_i$  ( $\forall i = 1, \dots, m$ ), faça.
41.         Se  $f_i > tmp$ , então
42.              $tmp \leftarrow f_i, z \leftarrow i$ .
43.     Para cada dimensão  $j$  ( $\forall j = 1, \dots, m$ ), faça.
44.          $H_j^* \leftarrow x_{zj}$ .
45. Retorna  $H^*$ .

Linhas 26 – 28 calcula as equações 4.11 e 4.16 para a melhor massa  $k$ , como descrito anteriormente. O valor de  $k$  decresce no tempo (idade do universo) na Linha 30. Observe que optou-se pela redução de  $k$  para 1 unidade, pois pode ser considerado uma opção “ingênua”.

O laço nas Linhas 31 – 38 atualiza a velocidade e a posição de cada massa. A Linha 33 atualiza a velocidade de acordo com a Equação 4.17 e as Linhas 37 – 38 atualizam a posição de cada partícula de massa, como descrito pelo algoritmo GSA binário na Equação 5.3. A condição nas Linhas 34 – 36 limita a velocidade da massa para  $|v_i^d| < v_{max} = 6$ . Finalmente, na Linha 39 encontra-se a partícula de massa, que proporciona o máximo de precisão e seu valor de índice são armazenados na variável  $z$  na Linha 42. O vetor de saída com as características selecionadas é composto nas Linhas 43 – 44 e retornado pelo algoritmo na Linha 45.



# Capítulo 6

## Procedimentos Metodológicos

Neste capítulo são apresentados os dados relativos à base de dados, a aplicação das técnicas de reconhecimento de padrões para detectar o pterígio, a aplicação dos três algoritmos de seleção de características propostos e os resultados experimentais alcançados.

### 6.1 Base de dados

A base de dados utilizada pelo trabalho é proveniente de um Projeto de Políticas Públicas que visou a criação de uma Unidade Móvel para atendimento oftalmológico, com a intenção de realizar pesquisas e assistência a pacientes com queixas oculares, como uma extensão dos serviços prestados pela Faculdade de Medicina (FMB) da UNESP de Botucatu dentro de Comunidades que se encontram fora dos “muros” da mesma.

Essa proposta decorreu do fato de que na Regional de Saúde onde a FMB está inserida há muitas cidades pequenas e que não possuem consultório oftalmológico. Desta forma, existe uma demanda grande de atendimento primário e secundário. Este fator impulsionou a criação do projeto voltado para esta população que, em geral, é de baixa renda e não possui outros meios de ter assistência especializada. Outro objetivo e o principal propósito foi o de avaliar as causas de cegueira e deficiência visual, usando amostra populacional. Estudos desta natureza não existem em nosso meio e são muito necessários para nortear ações na Comunidade, pois indicam onde se deve agir.

Esse projeto atendeu ao todo 11.121 pessoas nas cidades de Botucatu, São Manuel, Pirajú, Arandu, Areiópolis, Bofete, Conchas, Itaí, Manduri, Pereiras, Pratania e Taguaí, sendo que as 3 primeiras tiveram seus dados desconsiderados neste trabalho, tendo em vista que por elas terem sido estudadas no início do projeto, quando a unidade móvel ainda não estava completa, não foram analisadas todas as 89 características existentes nas demais cidades. Assim foram considerados 7,654 pacientes dos quais 6,972 são saudáveis e 682 são acometidos por pterígio. Essas características

foram extraídas por médicos especialistas em oftalmologia valendo-se de anamnese, avaliação da acuidade visual, exame externo, biomicroscopia e fundoscopia. Foi considerado somente o olho esquerdo de cada indivíduo, já que a utilização de ambos os olhos causaria uma repetição de fatores que poderiam descaracterizar a amostra, já que a doença é bilateral, apesar de assimétrica, e muito provavelmente teríamos o dobro de lesões que o relatado. Em termos de saúde pública o que interessa é o indivíduo, por isso a maioria dos estudos assim considera.

Cabe destacar que este estudo foi aprovado pelo Conselho de Ética em Pesquisa da Faculdade de Medicina de Botucatu e está em conformidade com os princípios da Declaração de Helsinque, uma declaração de princípios éticos criada pela Associação Médica Mundial com o intuito de fornecer orientações aos médicos e outros participantes em pesquisas clínicas envolvendo seres humanos, incluindo pesquisas com material humano identificável ou dados identificáveis. Consentimentos informados foram obtidos de todos os participantes. Cada paciente é descrito por 89 características, divididos em dados de identificação, acuidade visual, erro de refração, dados de anamnese, antecedentes oculares, antecedentes sistêmicos e antecedentes oculares familiares.

Foram executadas duas séries de experimentos, sendo que a primeira delas (Seção 6.2) objetiva o reconhecimento automático do pterígio, e a outra série (Seção 6.3) objetiva a caracterização do pterígio, ou seja, queremos encontrar quais características realmente influenciam no processo de aprendizado dos classificadores para a identificação dessa enfermidade.

## 6.2 Identificação de pterígio

Foram utilizados 50% da base para a etapa de treinamento dos classificadores e os 50% restantes para a classificação dos dados, sendo que este experimento foi repetido 10 vezes com conjuntos de treinamento e teste gerados aleatoriamente para o cálculo da média da acurácia e desvio padrão (técnica de validação cruzada).

Os classificadores empregados no presente trabalho foram:

- OPF: classificador Floresta de Caminhos Ótimos. Foi utilizado o pacote LibOPF [92], uma biblioteca em linguagem C de código aberto para o desenvolvimento de classificadores baseados em floresta de caminhos ótimos. Cabe destacar que a métrica utilizada é o logaritmo da distância euclideana;
- SVM-RBF: classificador SVM com função de núcleo de base radial (*Radial Basis Function* - RBF). Foi utilizada a biblioteca SVMtorch [93] com otimização de parâmetros;
- SOM: classificador por Mapas Auto-Organizáveis de Kohonen [94] (*Self Organizing Maps* - SOM). Foi utilizada implementação própria com uma grade neural de tamanho  $100 \times 100$  com

10 iterações para o aprendizado;

- SVM-LINEAR: classificador SVM com função de núcleo linear. Foi utilizada a biblioteca LIBSVM [78] com otimização de parâmetros;
- SVM-NOKERNEL: classificador SVM sem função de mapeamento. Foi utilizada a biblioteca LibLINEAR [95] com otimização de parâmetros;
- SVM-SIGMOID: classificador SVM com função de núcleo sigmoidal. Foi utilizada a biblioteca LIBSVM [78] com otimização de parâmetros;
- ANN-MLP: classificador Redes Neurais Artificiais com Perceptron Multicamadas (*Artificial Neural Networks with Multilayer Perceptron* - ANN-MLP) [8]. Foi utilizada a biblioteca FANN [96]. A arquitetura neural utilizada foi  $i:h_1:h_2:o$ , onde  $i = 89$  (número de características),  $h_1 = h_2 = 8$  e  $o = 2$  (número de classes) correspondem ao número de neurônios nas camadas de entrada, escondida e de saída, respectivamente. A rede neural foi treinada com o algoritmo de retropropagação, e sua arquitetura foi empiricamente escolhida;
- BAYES: classificador Bayesiano [97]. Foi utilizada implementação própria;
- $k$ -NN: classificador por  $k$ -vizinhos mais próximos ( $k$ -Nearest Neighbors -  $k$ -NN) [97]. Foi utilizada implementação própria. A fase de treinamento deste classificador foi implementada para encontrar o valor de  $k$  que maximizasse a taxa de acerto no conjunto de treinamento. Em seguida, esse valor de  $k$  era utilizado para a etapa de classificação e
- LA-SVM: classificador SVM com com função de núcleo linear implementado pela biblioteca LASVM [98].

A Tabela 1 mostra os resultados da taxa de acerto médio e respectivo desvio padrão de cada classificador, bem como seus tempos médios de treinamento e teste em segundos.

Podemos notar que as técnicas OPF, SVM-RBF, SOM, SVM-LINEAR, SVM-SIGMOIDE,  $k$ -NN e LA-SVM obtiveram resultados semelhantes. Entretanto, ao analisarmos o tempo de treinamento, podemos considerar as técnicas OPF, SOM e LA-SVM como sendo as mais rápidas dentre as que obtiveram os melhores resultados. Nesse contexto, OPF foi 2.37 vezes mais rápido que SOM e 1.10 vezes mais rápido que LA-SVM para a etapa de treinamento.

Já com relação ao tempo para classificação das amostras, se considerarmos as técnicas OPF, SOM e LA-SVM novamente, temos que OPF foi 3.98 vezes mais rápido que LA-SVM e a técnica SOM foi 1.67 vezes mais rápida que OPF. Entretanto, se considerarmos o tempo total de execução, ou seja, treinamento e teste, o classificador OPF foi 2.13 vezes mais rápido que SOM. Assim sendo, podemos

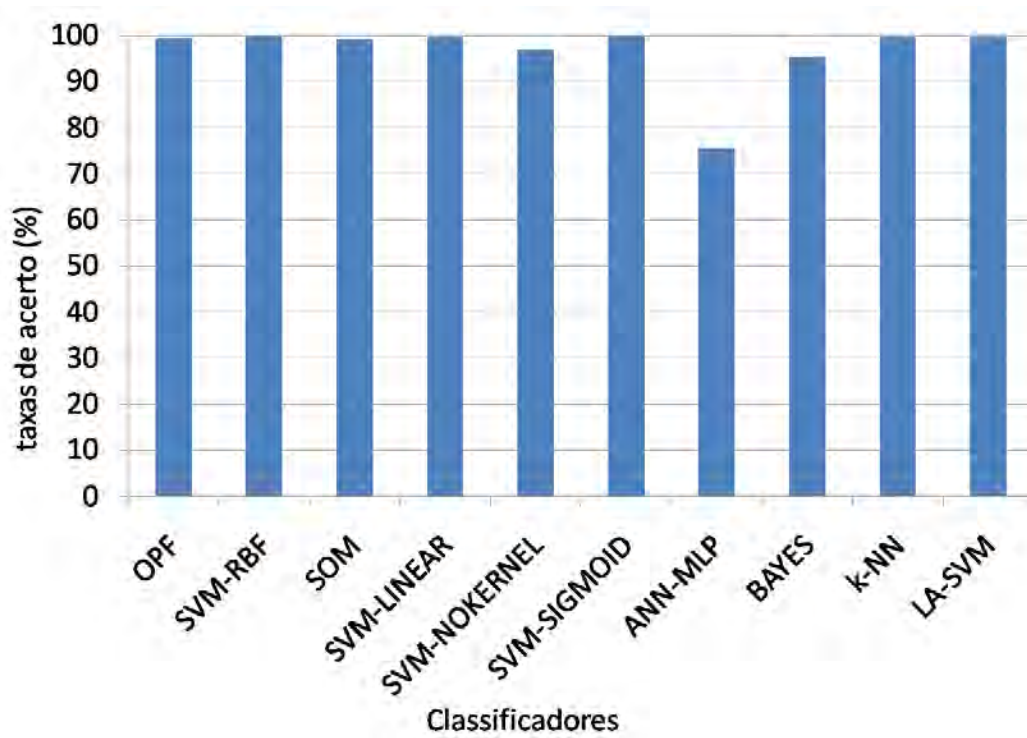


Gráfico 1: Acurácia média.

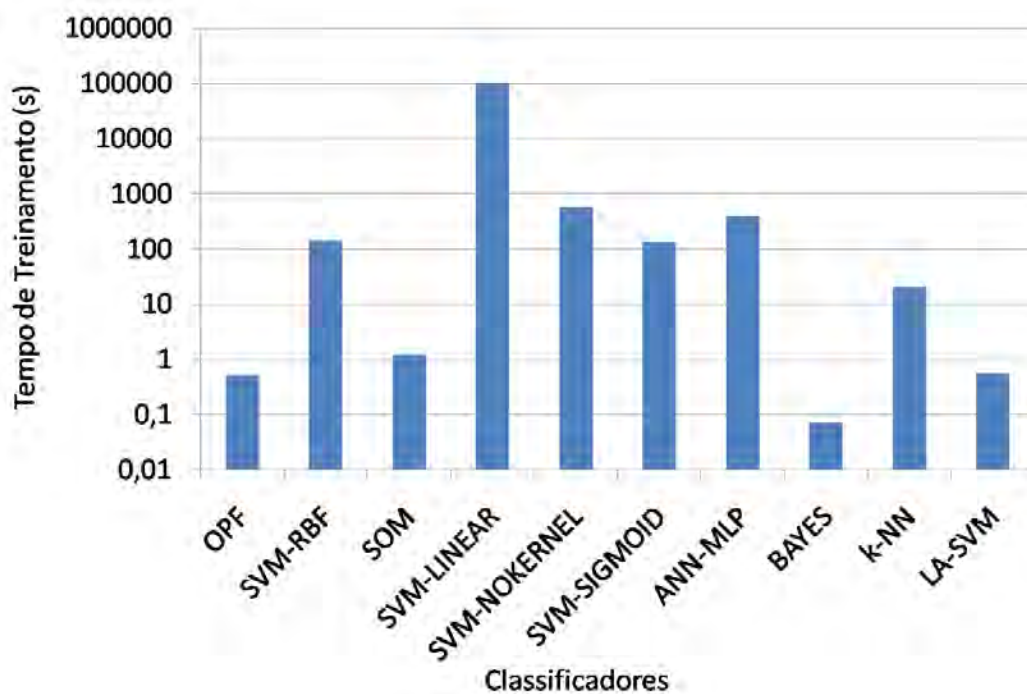


Gráfico 2: Tempo médio de treinamento (escala logarítmica).

Tabela 1: Resultados para a identificação automática de pterígio.

Classificador	Acurácia	Treinamento [s]	Classificação [s]
OPF	99.4148% $\pm$ 0.06788	0.5185	0.08063
SVM-RBF	99.7791% $\pm$ 0.05850	140.7479	0.44083
SOM	99.1992% $\pm$ 0.07256	1.2338	0.04808
SVM-LINEAR	99.5824% $\pm$ 0.41017	102382.0468	0.08111
SVM-NOKERNEL	96.8788% $\pm$ 1.60589	568.9259	0.05072
SVM-SIGMOID	99.7992% $\pm$ 0.07256	132.8938	0.17976
ANN-MLP	75.5107% $\pm$ 1.85572	399.8948	0.00244
BAYES	95.4006% $\pm$ 1.00308	0.0699	0.79738
$k$ -NN	99.6295% $\pm$ 0.38687	207.655	0.41204
LA-SVM	99.5710% $\pm$ 0.04477	0.5723	0.32083

concluir que o classificador OPF apresenta-se como sendo uma boa técnica para a identificação automática de pterígio devido, principalmente, a sua eficiência no treinamento e eficácia para a etapa de classificação.

Em um sistema funcional para diagnóstico de doenças, quando aplicado em campo, é interessante o retreinamento sempre que um novo indivíduo é inserido, e dessa forma, a velocidade de treinamento se torna um fator extremamente importante. Assim, mesmo as técnicas que possuem tempo de classificação menores, se não possuírem um treinamento rápido, inviabilizariam a funcionalidade do sistema.

### 6.3 Caracterização de pterígio

Esta seção descreve os experimentos realizados utilizando os três algoritmos de seleção de características descritos na Seção 4. Como dito anteriormente, a ideia consiste em utilizar a taxa de acerto de um classificador qualquer em um conjunto de avaliação como valor de função objetivo a ser maximizada pelos algoritmos de otimização. Para tal, utilizamos 30% do conjunto de dados para o treinamento, 20% para compor o conjunto de avaliação e os 50% restantes para o conjunto de teste. Entretanto, um grande problema reside na metodologia a ser aplicada quando esses experimentos são executados várias vezes com o intuito de proporcionarmos uma análise estatística mais robusta. Nesse sentido, propusemos escolher as características que foram selecionadas um número de vezes acima de um limiar  $\Phi$ , o qual variava para diferentes valores de porcentagem, conforme ilustra o gráfico na Figura 4. Por exemplo, para  $\Phi = 50\%$ , isso significa que a característica era escolhida caso fosse selecionada pelo classificador, no mínimo, em 50% dos casos.

A Tabela 2 mostra a taxa de reconhecimento média, bem como o limiar  $\Phi$  que obteve esse melhor desempenho e o número de características selecionadas para cada uma das técnicas empregadas.

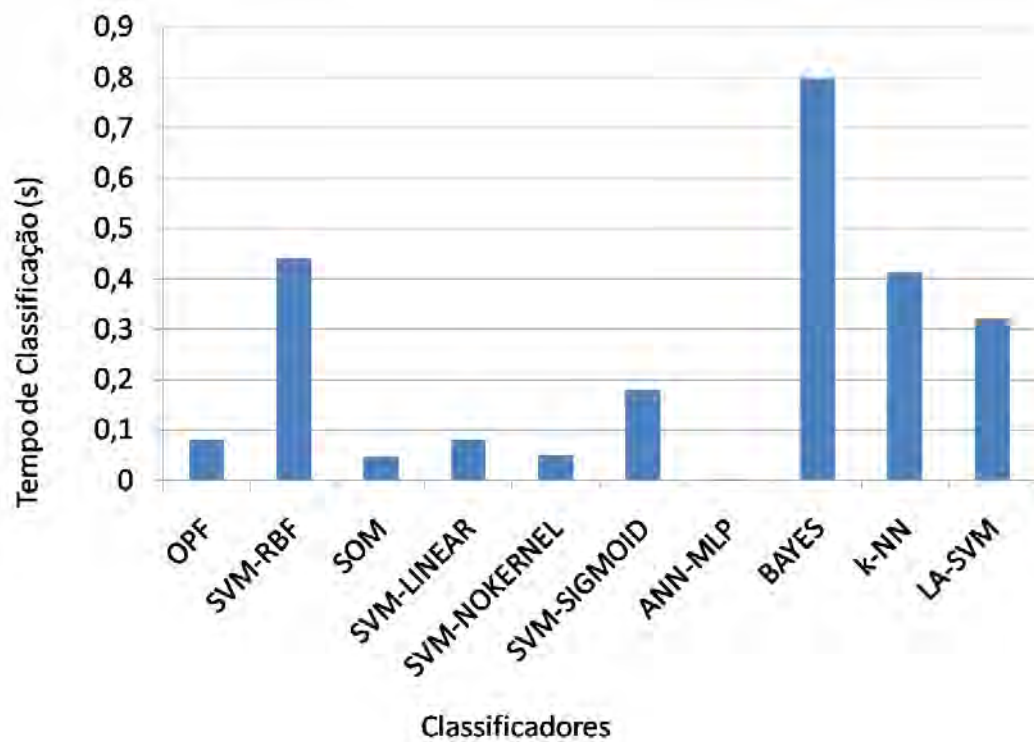


Gráfico 3: Tempo médio de classificação.

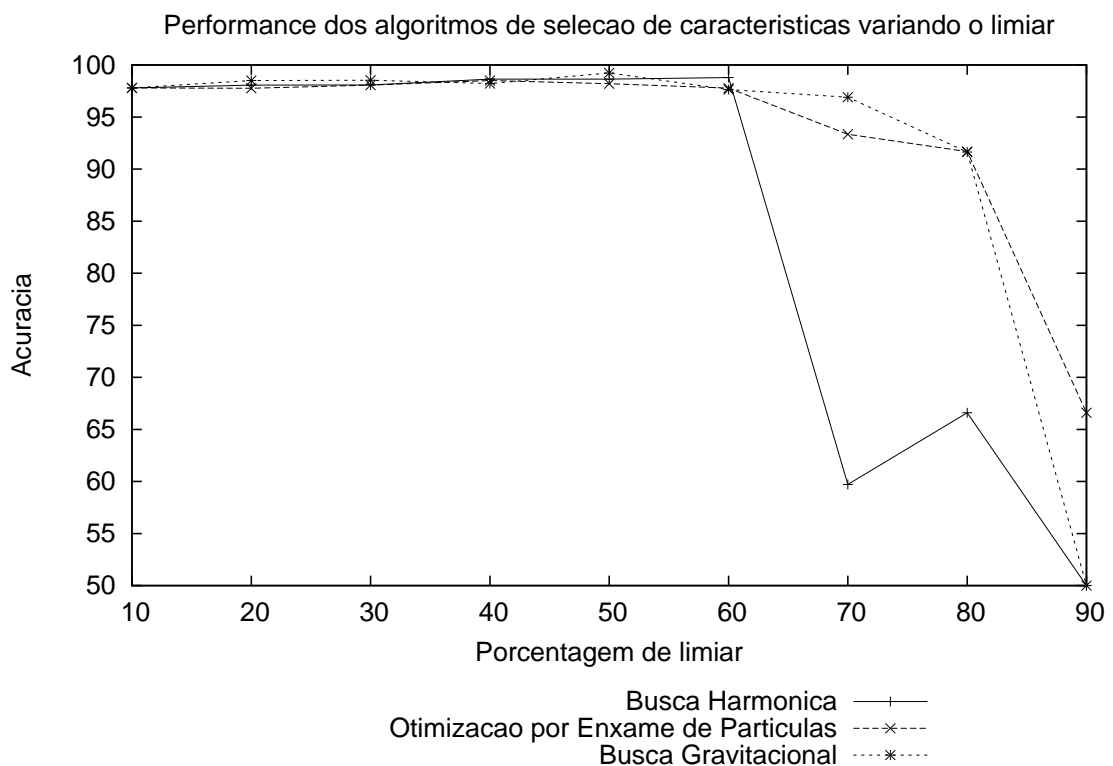
Gráfico 4: Gráfico da taxa de reconhecimento média com relação à variações no valor do limiar  $\Phi$ .

Tabela 2: Resultados para a caracterização de pterígio.

Técnica	Acurácia	$\Phi$	# características
PSO-OPF	98.49	40%	67
GSA-OPF	99.23	50%	53
HS-OPF	98.79	60%	34

Note que as técnicas atingiram acurácias muito próximas, sendo GSA-OPF a abordagem mais eficaz nesse quesito. Muito embora os resultados tenham sido similares à etapa anterior, o número de características foi diminuído em 17.28% na técnica PSO-OPF, 34.56% no caso da técnica GSA-OPF e 58.02 % na técnica HS-OPF. A Tabela 3 mostra o tempo de execução em segundos de cada uma das técnicas empregadas na etapa de seleção de características.

Tabela 3: Tempo de execução em segundos para a etapa de seleção de características.

Técnica	Tempo [s]
PSO-OPF	9.52
GSA-OPF	19.02
HS-OPF	9.62

Podemos notar que o método GSA-OPF foi o mais custoso computacionalmente. Isso deve-se ao fato de o mesmo possuir uma etapa que encontra as  $k$  massas que exercem a maior influência no posicionamento de uma massa em questão. Essa etapa é realizada utilizando um algoritmo de ordenação, geralmente o Quicksort [99]. Embora seja um algoritmo com complexidade polinomial, o custo computacional aumenta consideravelmente de acordo com o número de soluções iniciais (massas), tornando o algoritmo mais lento que os demais.

Nos Quadros 1 e 2 estão listadas todas as características da base de dados relacionadas com cada técnica aplicada. Ocorreram concordância entre os três métodos quando se considerou algumas variáveis como sexo, AV, Baixa de AV PP, Dor de Cabeça, Secreção, Ardência, DM + HAS, Asma, Antecedentes de Pterígio, Ordeolo, DCR, Conjuntivite, Cicatriz Toxoplasmose, Herpes e Toxoplasmose.

Destas variáveis, com relação ao sexo, já foi comentado que a doença pode estar relacionada mais com a questão estética do que propriamente com a característica. Assim, as mulheres podem apresentar as maiores queixas da doença. Alterações da Acuidade Visual (vista em duas características), é um fator sabidamente relacionado, já que a lesão causa astigmatismo, assim como a queixa de dor de cabeça, ardência e secreção. As queixas de olhos vermelhos e fotofobia, também muito frequentes, foram selecionadas somente pelo método PSO-OPF. A toxoplasmose ocular, em especial, a cicatriz, é muito frequente entre nós e a detecção pelas três técnicas pode estar a este fato, associada. O mesmo pode se dizer das doenças sistêmicas como Asma e DM + HAS. Pode-se notar que o antecedente

Característica	Tipo	PSO-OPF	HS-OPF	GSA-OPF
1.Idade	Dados Clínicos			
2.Sexo	Dados Clínicos	X	X	X
3.Cor	Dados Clínicos			X
4.AV/SC	Dados Clínicos	X		
5.AV/CC	Dados Clínicos		X	
6.PID	Dados Clínicos	X		X
7.Auto Esférico	Dados Clínicos	X		X
8.Auto Cilíndrico	Dados Clínicos	X	X	
9.Auto Eixo	Dados Clínicos	X		
10.Auto Eq. Esf.	Dados Clínicos			
11.R1	Dados Clínicos			X
12.R2	Dados Clínicos			X
13.AV	Dados Clínicos	X	X	X
14.Esférico	Dados Clínicos	X		
15.Cilíndrico	Dados Clínicos	X		X
16.Eixo	Dados Clínicos	X	X	
17.Eq. Esf.	Dados Clínicos	X		X
18.Dor	Queixas			X
19.Corpo Estranho	Queixas	X		X
20.Hiperemia	Queixas			
21.Baixa de AV PP	Queixas	X	X	X
22.Baixa de AV PL	Queixas	X	X	
23.Dor de Cabeça	Queixas	X	X	X
24.Olho Vermelho	Queixas	X		
25.Secreção	Queixas	X	X	X
26.Fotofobia	Queixas	X		
27.Prurido	Queixas			
28.Lacrimejamento	Queixas			X
29.Ardência	Queixas	X	X	X
30.Embaçamento	Queixas	X		X
31.Coceira	Queixas	X	X	
32.DM	Ant. Sistêmicos	X		
33.HAS	Ant. Sistêmicos		X	
34.Alcoolismo	Ant. Sistêmicos	X		
35.Fumante	Ant. Sistêmicos			X
36.Outros	Ant. Sistêmicos	X		
37.DM + HAS	Ant. Sistêmicos	X	X	X
38.Reumatismo	Ant. Sistêmicos		X	
39.Asma	Ant. Sistêmicos	X	X	X
40.Vitiligo	Ant. Sistêmicos		X	X
41.Tuberculose	Ant. Sistêmicos		X	X
42.Anemia	Ant. Sistêmicos	X		
43.Mal de Parkinson	Ant. Sistêmicos	X		X
44.Trauma	Ant. Oculares		X	X
45.Pterígio	Ant. Oculares	X	X	X

Quadro 1: Características Seleccionadas por cada Algoritmo Parte 1/2.



Característica	Tipo	PSO-OPF	HS-OPF	GSA-OPF
46.Calazio	Ant. Oculares	X		
47.Ordeolo	Ant. Oculares	X	X	X
48.DCR	Ant. Oculares	X	X	X
49.Ulcera	Ant. Oculares	X		X
50.Conjuntivite	Ant. Oculares	X	X	X
51.Cicatriz Toxoplasmose	Ant. Oculares	X	X	X
52.Catarata	Ant. Oculares	X		X
53.Descolamento Retina	Ant. Oculares	X	X	
54.Pseudofacico	Ant. Oculares	X		X
55.Glaucoma	Ant. Oculares	X		X
56.Estrabismo	Ant. Oculares		X	X
57.Triquíase (Laser)	Ant. Oculares	X		
58.Herpes	Ant. Oculares	X	X	X
59.Toxoplasmose	Ant. Oculares	X	X	X
60.CBC	Ant. Oculares		X	X
61.Nistagmo	Ant. Oculares	X		
62.Entropio	Ant. Oculares	X		X
63.C. Refrativa	Ant. Oculares	X		X
64.Ptose	Ant. Oculares		X	X
65.Trombose Veia Central	Ant. Oculares	X		X
66.Uveite	Ant. Oculares	X		
67.Trauma	Ant. O. Fam.	X		
68.Pterígio	Ant. O. Fam.	X		
69.Calazio	Ant. O. Fam.	X		
70.Ordeolo	Ant. O. Fam.	X		X
71.DCR	Ant. O. Fam.			X
72.Ulcera	Ant. O. Fam.			X
73.Conjuntivite	Ant. O. Fam.	X	X	
74.Cicatriz Toxoplasmose	Ant. O. Fam.	X		X
75.Catarata	Ant. O. Fam.	X	X	
76.Descolamento Retina	Ant. O. Fam.	X	X	
77.Pseudofacico	Ant. O. Fam.	X		X
78.Glaucoma	Ant. O. Fam.	X		X
79.Estrabismo	Ant. O. Fam.	X		
80.Triquíase (Laser)	Ant. O. Fam.	X		X
81.Herpes	Ant. O. Fam.	X	X	
82.Toxoplasmose	Ant. O. Fam.	X		X
83.CBC	Ant. O. Fam.	X		
84.Nistagmo	Ant. O. Fam.	X		
85.Entropio	Ant. O. Fam.	X	X	
86.C. Refrativa	Ant. O. Fam.	X		X
87.Ptose	Ant. O. Fam.	X		X
88.Trombose Veia Central	Ant. O. Fam.			X
89.Uveite.	Ant. O. Fam.	X		X

Quadro 2: Características Seleccionadas por cada Algoritmo Parte 2/2.

de pterígio também está presente nas três técnicas. Isso reforça o problema da recidiva da doença, o que é muito comum. Outro fator interessante apresentado é que nenhuma das características do grupo de antecedentes oculares familiares foi selecionada nas três técnicas concomitantemente, o que sugere que a teoria da hereditariedade não é válida. Essas análises são suposições e necessitam de um aprofundamento especialista maior por parte dos oftalmologistas.

# Capítulo 7

## Conclusão

Pterígio é uma enfermidade que tem acometido um grande número de pessoas atualmente, sendo que os fatores que ocasionam tal doença ainda não foram estabelecidos pela comunidade científica. Nesse contexto, o presente trabalho objetivou a aplicação de algoritmos supervisionados de reconhecimento de padrões para a identificação automática do pterígio e sua posterior caracterização utilizando algoritmos de otimização com o intuito de selecionar as características que realmente são utilizadas no processo de aprendizado dos classificadores. Cabe destacar que esta doença foi escolhida por ser de fácil reconhecimento clínico, ou seja, não há viés de seleção, já que a identificação clínica positiva tem alto grau de acerto. Desta forma, a identificação da doença pela inteligência artificial corrobora perfeitamente com a detecção clínica.

Uma primeira bateria de testes foi realizada comparando 10 classificadores supervisionados de padrões, sendo que a melhor relação eficiência no treinamento/eficácia de classificação foi atingida pelo classificador OPF, seguido pelo classificador SOM. Os experimentos foram realizados utilizando uma base de dados com 7,654 pacientes da região de Botucatu-SP, Brasil, nos quais 6,972 pacientes são livres de pterígio e 682 são acometidos pela doença. Para a descrição de cada paciente, foram utilizados 89 dados clínicos. Nessa primeira fase dos experimentos, utilizamos 50% dos dados para treinamento dos classificadores e os 50% restantes para a etapa de classificação. Nessa etapa, os resultados mostraram uma superioridade da técnica OPF, já que a mesma apresentou praticamente a mesma acurácia das técnicas mais bem classificadas, porém com um custo computacional menor, principalmente na etapa de treinamento. Esse fator é de extrema importância para programas de auxílio ao diagnóstico de doenças, já que é desejável um sistema que permita a adição de novos elementos para posterior retreinamento do mesmo.

Uma segunda rodada de experimentos foi realizada com o intuito de identificarmos quais as características que realmente foram importantes para o processo de aprendizado do classificador. Assim, essa etapa objetivou a caracterização do pterígio, ou seja, a ideia seria identificar os potenciais fatores

de risco que levam a essa doença. Essa etapa consistiu em utilizar a taxa de acerto do classificador OPF, o qual obteve os melhores resultados no experimento anterior, como sendo a função objetivo a ser maximizada por três algoritmos de otimização: PSO-OPF, HS-OPF e GSA-OPF. Para tal, utilizamos 30% dos dados para compor o conjunto de treinamento, 20% para o conjunto de avaliação e os 50% restantes para o conjunto de teste. Nos resultados obtidos, tivemos uma redução no número de características de 17.28 % na técnica PSO-OPF, 34.56 % no caso da técnica GSA-OPF e 58.02 % na técnica HS-OPF, mantendo a taxa de acerto similar à obtida na etapa anterior, ou seja, com todas as características disponíveis. Todas as técnicas obtiveram resultados muito próximos em termos de eficácia, sendo que o PSO-OPF selecionou 67 características em 9.52 segundos, GSA-OPF e o HS-OPF selecionaram 53 e 34 características em 19.02 e 9.62 segundos, respectivamente, demonstrando que a técnica HS-OPF é a melhor escolha para os propósitos do trabalho quando se diz respeito ao número de características e o custo computacional. Várias características defendidas na literatura como dados relacionados ao pterígio foram identificados pelas técnicas, tais como as queixas de dor de cabeça, ardência e secreção e o antecedente de pterígio. Porém, é prematuro afirmar qual seria a melhor técnica para a caracterização do pterígio, já que para isso seria necessária uma maior discussão e análise por parte dos oftalmologistas, já que a literatura médica é irrigada de controvérsias quando se diz respeito à caracterização dessa doença.

# Trabalhos Publicados Pelo Autor

1. PAGNIN, A. F.; PAPA, J. P.; SCHELLINI, S. A.; DAIBES NETO, S. N. Designing a Risk Profile for Pterygium Through an Optimum-Path Forest Analysis Pathway. In: *V Encontro de Pós-Graduação da FMB*, 2009, Botucatu. Anais do V Encontro de Pós-Graduação da FMB, 2009.
2. PAGNIN, A. F.; SCHELLINI, S. A.; PAPA, J. P. Supervised Pattern Classification Applied for Automatic Pterygium Identification. In: *XXI Congress of the International Society Geographical & Epidemiological Ophthalmology*, 2010, Berlim. Book of Abstracts, 2010. p. 35-35.
3. PAGNIN, A. F.; SCHELLINI, S. A.; PAPA, J. P. Preliminary Diagnosis of Ophtalmological Diseases Through Machine Learning Techniques. *Recent Patents on Signal Processing*, v. 1, p. 74-79, 2011.
4. PAPA, J. P. ; PAGNIN, A. F. ; SCHELLINI, S. A. ; SPADOTTO, A. A. ; GUIDO, R. C. ; PONTI JUNIOR, M. P. ; CHIACHIA, G. ; FALCAO, A. X. . Feature Selection Through Gravitational Search Algorithm. In: *36th International Conference on Acoustics, Speech and Signal Processing*, p. 38, 2011, Praga. (Qualis B2-CC)
5. SPADOTTO, A. A. ; GUIDO, R. C. ; CARNEVALI, F. ; PAGNIN, A. F. ; PAPA, J. P. .Improving Parkinson's Disease Identification Through Evolutionary-Based Feature Selection. In: *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2011, Boston, MA USA. *33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2010. p. 325. (Qualis B2-CC)

# Referências Bibliográficas

- [1] STEIN, H. A.; FREEMAN, M. J.; STENSON, S. M. *Guia CLAO para refração e óculos*. Soblec, 2009.
- [2] Visual impairment and blindness. Fact Sheet 282, World Health Organization, May 2009.
- [3] SERRACARBASSA, L. L.; DANTAS, M. C. N. *Manual do cbo: Doenças externas oculares e córnea*. 1999. v. 8, Cap. Desordens degenerativas da conjuntiva-córnea-esclera, p. 371–374.
- [4] LIANG, X.; LI, F.; QIU, W. An epidemiological survey of blindness and low vision in mexican county. *Zhonghua Yan Ke Za Zhi*, v. 37(1), p. 12–5, 2001.
- [5] SCHELLINI, S. A.; VELOSO, C. E. R.; LOPES, W.; PADOVANI, C. R.; PADOVANI, C. R. P. Características de portadores de pterígio na região de Botucatu. *Arquivos Brasileiros de Oftalmologia*, v. 68, n. 3, p. 291–294, 2005.
- [6] LIN, A.; HOFFMAN, D.; GAASTERLAND, D. E.; CAPRIOLI, J. Neural networks to identify glaucomatous visual field progression. *American Journal of Ophthalmology*, v. 135, n. 1, p. 49–54, 2003.
- [7] HUANG, M. L.; CHEN, H. Y.; HUANG, J. J. Glaucoma detection using adaptive neuro-fuzzy inference system. *Expert Systems with Applications*, v. 32, p. 458–468, 2007.
- [8] HAYKIN, S. *Neural networks: a comprehensive foundation*. Prentice Hall, 1994.
- [9] GALILEA, E. H.; S.GARCÍA, G.; SUÁREZ-BÁRCENA, I. F. Identification of glaucoma stages with artificial neural networks using retinal nerve fibre layer analysis and visual field parameters. *Advances in Soft Computing: Innovations in Hybrid Intelligent Systems*, v. 44, p. 418–424, 2007.
- [10] PAPA, J. P.; FALCÃO, A. X.; SUZUKI, C. T. N. Supervised pattern classification based on optimum-path forest. *International Journal of Imaging Systems and Technology*, New York, NY, USA, v. 19, n. 2, p. 120–131, 2009.

- [11] FARIA, F. P. *Manual de biomicroscopia da conjuntiva*. 1989. Cap. Pterígio e Conjuntivite lenhosa, p. 141–144.
- [12] ALVES, M. R.; VICTOR, G. O tratamento do pterígio. *Revista Brasileira de Oftalmologia*, v. 64, n. 5, p. 351–362, 2005.
- [13] BURATTO, L.; PHILLIPS, R. L.; CARITÓ, G. *Pterygium surgery*. 1. ed. Slack Incorporated, set 2000.
- [14] ALVES, M. R. *Conjuntiva cirúrgica*. 1999. Cap. Pterígio, p. 59–66.
- [15] CHRISTOV, R.; FORNO, E. A.; CAMPAGNA, C. M.; BECHARA, S. J.; JOSÉ, N. K. Análise histopatológica de pterígios primários e recidivados. *Revista Brasileira de Oftalmologia*, v. 50, n. 4, p. 59–62, 1991.
- [16] TOLANI, A.; MELKI, L. Pterygium current concepts in pathogenesis and treatment. *International Ophthalmology Clinics*, v. 1, p. 21–30, 2009.
- [17] ALVES, M. R.; SATO, S.; AZEVEDO, M. L. Pterígio: histopatologia e recidivas. *Arquivos Brasileiros de Oftalmologia*, v. 43, p. 242–245, 1980.
- [18] TAYLOR, H. R.; WEST, S. K.; NOZ, B. M.; ROSENTHAL, F. S.; BRESSLER, S. B.; BRESSLER, N. M. The long term effects of visible light on the eye. *Archives of Ophthalmology*, v. 110, n. 1, p. 99–104, 1992.
- [19] PANCHAPAKESAN, J.; HOURIHAN, F.; MITCHELL, P. Prevalence of pterygium and pinguecula: the blue mountains eye study. *Australian and New Zealand Journal of Ophthalmology*, v. 26, n. 1, p. S2–5, 1998.
- [20] WONG, T. Y.; FOSTER, P. J.; JOHNSON, G. J.; SEAH, S. K.; TAN, D. T. The prevalence and risk factors for pterygium in an adult chinese population in Singapore: the tanjong pagar survey. *American Journal of Ophthalmology*, v. 131, n. 2, p. 176–83, 2001.
- [21] NETO, T. G.; GARRIDO, C.; CARVALHO, R. C.; LIMA, H. C. Estudo da frequência de pterígio em hospitais de Salvador e Manaus. *Revista Brasileira de Oftalmologia*, v. 55, n. 9, p. 683–686, 1996.
- [22] LIU, H.; YANG, J.; ZHONG, L. Prevalence survey on pterygium in two countries of Hainan Province. *Zhonghua Yan Ke Za Zhi*, v. 37, n. 1, p. 21–23, 2001.

- [23] ROJAS, J. R.; MALAGA, H. Pterygium in Lima, Peru. *Annals of Ophthalmology*, v. 18, n. 4, p. 147–149, 1986.
- [24] SAW, S. M.; BANERJEE, K.; TAN, D. Risk factors for the development of pterygium in Singapore: a hospital-based case-control study. *Acta Ophthalmologica Scandinavica*, v. 78, n. 2, p. 216–220, 2000.
- [25] MORAN, D. J.; HOLLOWES, F. C. Pterygium and ultraviolet radiation: a positive correlation. *British Journal of Ophthalmology*, v. 68, n. 5, p. 343–346, 1984.
- [26] ROH, S.; WEITER, J. J. Light damage to the eye. *Journal of the Florida Medical Association*, v. 81, n. 4, p. 248–251, 1994.
- [27] CORNAND, G. Pterygium. clinical course and treatment. *Revue Internationale Trachome Pathologie Oculaire Tropicale Subtropicale Sante Publique*, v. 66, n. 3-4, p. 31–108, 1989.
- [28] MACKENZIE, F. D.; HIRST, L. W.; BATTISTUTTA, D.; GREEN, A. Risk analysis in the development of pterygia. *Ophthalmology*, v. 99, n. 7, p. 1056–1061, 1992.
- [29] TANG, F. C.; CHEN, S. C.; LEE, H. S.; LIN, W. F.; CHOU, M. C.; LEE, M. C. Relationship between pterygium/pinguecula and sunlight exposure among postmen in central Taiwan. *Zonghua Yi Xue Za Zhi (Taipei)*, v. 62, n. 8, p. 496–502, 1999.
- [30] SOLOMON, A. S. Pterygium. *British Journal of Ophthalmology*, v. 90, p. 665–666, 2006.
- [31] MCCARTY, C. A.; FU, C. L.; TAYLOR, H. R. Epidemiology of pterygium in Victoria, Australia. *British Journal of Ophthalmology*, v. 84, n. 3, p. 289–292, 2000.
- [32] CAMERON, M. E. Pterygium through the world. 1965.
- [33] ELLIOTT, R. The aetiology and pathology of pterygium. *Transactions of the Ophthalmological Society of Australia*, v. 25, p. 71–74, 1966.
- [34] HIRST, L. W. The treatment of pterygium. *Survey of Ophthalmology*, v. 48, n. 2, p. 145–180, 2003.
- [35] JENSEN, O. L. Pterygium, the dominant eye and the habit of closing one eye in sunlight. *Acta Ophthalmologica (Copenh)*, v. 60, n. 4, p. 568–574, 1982.
- [36] JOHNSON, G. J. Aetiology of spheroidal degeneration of the cornea in labrador. *British Journal of Ophthalmology*, v. 65, n. 4, p. 270–283, 1981.



- [37] KARAI, I.; HORIGUCHI, S. Pterygium in welders. *British Journal of Ophthalmology*, v. 68, n. 5, p. 347–349, 1984.
- [38] HILGERS, J. H. Pterygium: its incidence, heredity and etiology. *American Journal of Ophthalmology*, v. 50, p. 635–644, 1960.
- [39] MAEDA, L.; POTERIO, M. B.; SERPA, J. F.; ALVES, M. R.; JOSÉ, N. K. Análise da relação entre a ocorrência de pterígio e dominância ocular. *Arquivos Brasileiros de Oftalmologia*, v. 59, n. 5, p. 511–513, 1996.
- [40] FIALHO, T. A. Pterígio: considerações atualizadas; estudo sucinto. *Revista Brasileira de Oftalmologia*, v. 47, n. 3, p. 171–173, 1988.
- [41] DUSHKU, N.; SCHEELTZ, G. S.; REID, T. W. Pterygia pathogenesis: corneal invasion altered limbal basal stem cells and activation of fibroblasts. *Investigative Ophthalmology & Visual Science*, v. 15, p. S2388, 2000.
- [42] CORONEO, M. T.; GIROLAMO, N. D.; WAKEFIELD, D. The pathogenesis of pterygia. *Current Opinion in Ophthalmology*, v. 10, p. 282–288, 1999.
- [43] SCHELLINI, S. A.; HOYAMA, E.; OLIVEIRA, D. E. Matrix metalloproteinase-9 expression in pterygium. *Arquivos Brasileiros de Oftalmologia*, v. 69, p. 161–164, 2006.
- [44] LATOSINSKA, J. *Ultraviolet and environment*. Institute of Physics, Adam Mickiewicz University, 2011.
- [45] REISMAN, D.; MCFADDEN, J. W.; LU, G. Loss of heterozygosity and expression in pterygium. *Cancer Left*, v. 206, p. 77–83, 2004.
- [46] LUTHRA, R.; NEMESURE, B. B.; WU, S. Y.; XIE, S. H.; LESKE, M. C. Barbados eye studies group. Frequency and risk factors for pterygium in the Barbados eye study. *Archives of Ophthalmology*, v. 119, n. 12, p. 1827–1832, 2001.
- [47] WONG, W. W. A hypothesis on the pathogenesis of the pterygiums. *Annals of Ophthalmology*, v. 10, n. 3, p. 303–308, 1978.
- [48] PINKERTON, O. D.; HOKAMA, Y.; SHIGEMURA, L. A. Immunologic basis for the pathogenesis of pterygium. *American Journal of Ophthalmology*, v. 98, n. 2, p. 225–228, 1984.
- [49] PIRAS, F.; MOORE, P. S.; UGALDE, J.; PERRA, M. T.; SCARPA, A.; SIRIGU, P. Detection of human papillomavirus dna in pterygia from different geographical regions. *British Journal of Ophthalmology*, v. 87, n. 7, p. 864–866, 2003.

- [50] VARINLI, S.; VARINLI, I.; KÖKSAL-ERKISI, M.; DORAN, F. Human papillomavirus in pterygium. *Central African Journal of Medicine*, v. 40, n. 1, p. 24–26, 1994.
- [51] DETORAKIS, E. T.; SOURVINOS, G.; SPANDIDOS, D. A. Detection of herpes simplex virus and human papilloma virus in ophthalmic pterygium. *Cornea*, v. 20, n. 2, p. 164–167, 2001.
- [52] GALLAGHER, M. J.; GIANNOUDIS, A.; HERRINGTON, C. S.; HISCOTT, P. Human papillomavirus in pterygium. *British Journal of Ophthalmology*, v. 85, n. 7, p. 782–784, 2001.
- [53] DUSHKU, N.; HATCHER, S. L.; ALBERT, D. M.; REID, T. W. p53 expression and relation to human papillomavirus infection in pingueculae, pterygia, and limbal tumors. *Archives of Ophthalmology*, v. 117, n. 12, p. 1593–1599, 1999.
- [54] CHEN, K. H.; HSU, W. M.; CHENG, C. C.; LI, Y. S. Lack of human papillomavirus in pterygium of chinese patients from Taiwan. *British Journal of Ophthalmology*, v. 87, n. 8, p. 1046–1048, 2003.
- [55] SCHELLINI, S. A.; HOYAMA, E.; SHIRATORI, C. A.; SAKAMOTO, R. H.; CANDEIAS, J. M. G. Ausência de papilomavirus (HPV) em pterígios de uma amostra brasileira. *Arquivos Brasileiros de Oftalmologia*, v. 69, n. 4, p. 519–521, 2006.
- [56] JACKLIN, H. N. Familial predisposition to pterygium formation: report of a family. *American Journal of Ophthalmology*, v. 57, p. 481–482, 1964.
- [57] ALVES, M. R. *Conjuntiva cirúrgica*. 1999. Cap. Tratamento cirúrgico do pterígio, p. 67–81.
- [58] TRUJILLO, F. El pterigion. *Archivos Peruanos de Patologia Clinica*, v. 25, p. 97–105, 1971.
- [59] AMSLER, M. On the problem of astigmatism associated with pterygium. *Ophthalmologica*, v. 126, n. 1, p. 52–54, 1953.
- [60] BEDROSSIAN, R. H. The effects of pterygium surgery on refraction and corneal curvature. *Archives of Ophthalmology*, v. 64, p. 553–557, 1960.
- [61] HANSEN, A.; NORN, M. Astigmatism and surface phenomena in pterygium. *Acta Ophthalmologica (Copenh)*, v. 58, n. 2, p. 174–181, 1980.
- [62] NETO, A. T.; ALVES, M. R.; JOSÉ, N. K. Alterações topográficas corneanas desencadeadas pelo pterígio. *Arquivos Brasileiros de Oftalmologia*, v. 59, n. 5, p. 443–448, 1996.
- [63] HOSNI, F. A. Pterygium in Qatar. *Ophthalmologica*, v. 174, n. 2, p. 81–87, 1977.

- [64] ASHAYE, A. O. Pterygium in Ibadan. *West African Journal of Medicine*, v. 10, n. 3-4, p. 232–243, 1991.
- [65] SEBBAN, A.; HIRST, L. W. Pterygium recurrence rate at the Princess Alexandra Hospital. *Australian and New Zealand Journal of Ophthalmology*, v. 19, n. 3, p. 203–206, 1991.
- [66] FERRAZ, F. H. S.; SCHELLINI, S. A.; HOYAMA, E.; BERNARDES, S. R.; PADOVANI, C. R. Pterígio e alterações da curvatura corneana. *Arquivos Brasileiros de Oftalmologia*, v. 65, p. 533–6, 2002.
- [67] KOVACS, Z. L. *Redes neurais artificiais: Fundamentos e aplicações*. Escola Politécnica da Universidade de São Paulo, 1996.
- [68] RAGHU, P. P.; POONGODI, R.; YEGNANARAYANA, B. A combined neural network approach for texture classification. *Neural Networks*, v. 8, p. 975–987, 1995.
- [69] KOHONEN, T. Self-organizing map. *Proceedings of the IEEE*, v. 78, p. 1464–1480, 1990.
- [70] KOHONEN, T. *Self-organizing and associative memory*. Springer-Verlag, 1989.
- [71] KOHONEN, T. Things you haven't heard about self-organizing map. *IEEE International Conference on Neural Networks*, v. III, p. 1147–1156, 1993.
- [72] HAYKIN, S. *Neural networks: a comprehensive foundation*. Pearson, Prentice Hall, 1999.
- [73] VAPNIK, V. N. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, v. 10, p. 988–999, 1999.
- [74] SCHÖLKOPF, B.; SMOLA, A. J. *Learning with kernels*. MIT Press, 2002.
- [75] SCHÖLKOPF, B.; WILLIAMSON, R. C.; BARTLETT, P. L. New support vector algorithms. *Neural Computation*, p. 1207–1245, 2000.
- [76] CORTES, C.; VAPNIK, V. Support-vector networks. *Machine Learning*, p. 273–297, 1995.
- [77] PHILLIPS, P. J. Support vector machines applied to face recognition. *Advances in Neural Information Processing Systems*, v. 11, p. 803–809, 1999.
- [78] CHANG, C.-C.; LIN, C.-J. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, New York, NY, USA, v. 2, p. 27:1–27:27, 2011.
- [79] MITCHELL, T. M. *Machine learning*. WCB/McGraw-Hill, 1997.

- [80] PAN, J.; QIAO, Y.; SUN, S. A fast k nearest neighbors classification algorithm. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, v. E87-A, 2004.
- [81] MARTINS, E. R. S.; MARQUES, P. M.; OLIVEIRA, L. F.; PEREIRA-JR, R. R.; TRAD, C. S. Caracterização de lesões intersticiais de pulmão em radiograma de tórax utilizando análise local de textura. *Radiologia Brasileira*, v. 38, n. 6, p. 421–426, 2005.
- [82] PAPA, J. P.; FALCÃO, A. X. A new variant of the optimum-path forest classifier. *4th International Symposium on Advances in Visual Computing*, p. 935–944, 2008.
- [83] PAPA, J. P.; FALCÃO, A. X. A learning algorithm for the optimum-path forest classifier. *7th IAPR Workshop on Graph-based Representations in Pattern Recognition*, 2009.
- [84] FALCÃO, A.; STOLFI, J.; LOTUFO, R. The image foresting transform: Theory, algorithms, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 26, n. 1, p. 19–29, 2004.
- [85] KENNEDY, J.; EBERHART, R. *Swarm intelligence*. Morgan Kaufman, 2001.
- [86] GEEM, Z. W. Novel derivative of harmony search algorithm for discrete design variables. *Applied Mathematics and Computation*, v. 199, p. 223–230, 2008.
- [87] HALLIDAY, D.; RESNICK, R.; J. WALKER, D. HALLIDAY, R. R.; WALKER, J. *Fundamentals of physics*. 2000. v. 6th Edition, Cap. Extended.
- [88] RASHEDI, E.; NEZAMABADI-POUR, H.; SARYAZDI, S. Gsa: A gravitational search algorithm. *Information Sciences*, New York, NY, USA, v. 179, n. 13, p. 2232–2248, 2009.
- [89] KENNEDY, J.; EBERHART, R. C. A discrete binary version of the particle swarm algorithm. *IEEE International Conference on Systems, Man, and Cybernetics*, v. 5, p. 4104–4108, 1997.
- [90] HIRAM, A. F.; GOODMAN, E. Swarmed feature selection. *Applied Image Pattern Recognition Workshop*, Los Alamitos, CA, USA, v. 0, p. 112–118, 2004.
- [91] RASHEDI, E.; SARYAZDI, S.; NEZAMABADI-POUR, H. BGSA: binary gravitational search algorithm. *Natural Computing*, v. 9, p. 727–745, 2010.
- [92] PAPA, J.; SUZUKI, C.; FALCÃO, A. X. *LibOPF: A library for the design of optimum-path forest classifiers*, 2009. Software version 2.0 available at <http://www.ic.unicamp.br/~afalcao/LibOPF>.

- [93] COLLOBERT, R.; BENGIO, S. SVM Torch: Support vector machines for large-scale regression problems. *Journal of Machine Learning Research*, v. 1, p. 143–160, 2001.
- [94] KOHONEN, T. (Ed.). *Self-organizing maps*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1997.
- [95] FAN, R.-E.; CHANG, K.-W.; HSIEH, C.-J.; WANG, X.-R.; LIN, C.-J. LIBLINEAR: A library for large linear classification. *The Journal of Machine Learning Research*, v. 9, p. 1871–1874, 2008.
- [96] NISSEN, S. *Implementation of a fast artificial neural network library (fann)*, 2003. Department of Computer Science University of Copenhagen (DIKU). Software available at <http://leenissen.dk/fann/>.
- [97] DUDA, R. O.; HART, P. E.; STORK, D. G. *Pattern classification*. Wiley-Interscience Publication, 2000.
- [98] BORDES, A.; ERTEKIN, S.; WESTON, J.; BOTTOU, L. Fast kernel classifiers with online and active learning. *The Journal of Machine Learning Research*, v. 6, p. 1579–1619, 2005.
- [99] HOARE, C. A. R. Quicksort. *Computer Journal*, v. 5, n. 1, p. 10–15, 1962.