UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

LUIZ OTÁVIO VILAS BÔAS OLIVEIRA

CSCDR: Um Classificador Baseado em Seleção Clonal com Redução de Células de Memória

Dissertação apresentada como requisito parcial
para a obtenção do grau de
Mestre em Ciência da Computação

Prof. Dr. Dante Augusto Couto Barone
Orientador

Profa. Dra. Isabela Neves Drummond
Coorientadora

Porto Alegre, agosto de 2012

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Oliveira, Luiz Otávio Vilas Bôas

CSCDR: Um Classificador Baseado em Seleção Clonal com Redução de Células de Memória / Luiz Otávio Vilas Bôas Oliveira. – Porto Alegre: PPGC da UFRGS, 2012.

81 f.: il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2012. Orientador: Dante Augusto Couto Barone; Coorientadora: Isabela Neves Drummond.

1. Sistema imunológico artificial. 2. Seleção clonal. 3. Classificação de dados. I. Barone, Dante Augusto Couto. II. Drummond, Isabela Neves. III. Título.

“Quem pensa pouco erra muito.”
— LEONARDO DA VINCI

AGRADECIMENTOS

Primeiramente, agradeço a Deus. O Universo é grande e complexo demais para acreditarmos que estamos aqui por acaso. Agradeço pela oportunidade de ser o que sou e estar onde estou.

Aos meus pais, José Carlos e Ângela Maria, por tudo que já fizeram, fazem e farão por mim. Por confiarem, torcerem, apoiarem e principalmente por me ajudarem a chegar onde cheguei.

Aos meus familiares e amigos que ficaram em Minas. Agradeço pelo apoio, pela preocupação e pela amizade.

Aos meus amigos de Porto Alegre, Marcelo, Rê, Sérgio, Anderson, Tomio, Grasi, Alice e todos os outros. Pelos bons momentos, pelas risadas, pelas piadas, pela ajuda e pelo apoio. Que nossa amizade perdure através da distância e dos anos.

Ao meu orientador, Professor Dante Augusto Couto Barone. Obrigado pela oportunidade, pela orientação, pela motivação e pela amizade.

À minha coorientadora, Professora Isabela Neves Drummond. Obrigado por atender ao meu pedido de coorientação, pela disponibilidade, pela ajuda, pelos conselhos e principalmente pela amizade.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS 9

LISTA DE FIGURAS 11

LISTA DE ALGORITMOS 12

LISTA DE TABELAS 13

RESUMO 14

ABSTRACT 15

1 INTRODUÇÃO 16

1.1 Motivação 16

1.2 Trabalhos Relacionados 17

1.3 Hipótese e Objetivos 18

1.4 Organização da Dissertação 18

2 SISTEMA IMUNOLÓGICO BIOLÓGICO 20

2.1 Constituintes do Sistema Imunológico 20

2.2 Anatomia do Sistema Imunológico 21

2.3 Sistema Imune Adaptativo 23

2.4 Reconhecimento Antigênico 24

2.4.1 Especificidade Antigênica 24

2.4.2 Espaço de Formas 25

2.5 Seleção Clonal 25

2.6 Memória Imunológica . . . 26
2.7 Maturação de Afinidade . . . 28
2.8 Não Reatividade a Antígenos Próprios . . . 28
2.9 Rede Imunológica . . . 29

3 SISTEMAS IMUNOLÓGICOS ARTIFICIAIS . . . 30
3.1 Considerações Iniciais . . . 31
3.2 CLONALG . . . 32
3.2.1 Inspiração Biológica . . . 32
3.2.2 Algoritmo . . . 33
3.3 CSCA . . . 34
3.3.1 Inicialização e Particionamento do Conjunto de Entrada . . . 34
3.3.2 Fase Iterativa . . . 35
3.3.2.1 Poda e Seleção . . . 36
3.3.2.2 Clonagem . . . 36
3.3.2.3 Mutação . . . 36
3.3.3 Fase Final . . . 36
3.4 AIRS . . . 37
3.4.1 Inicialização . . . 37
3.4.2 Geração de ARBs . . . 37
3.4.3 Competição por Recursos . . . 39
3.4.4 Introdução da Célula de Memória . . . 39
3.4.5 Outras Versões . . . 40
3.5 Outros Algoritmos . . . 40

4 APRENDIZADO BASEADO EM INSTÂNCIA . . . 41
4.1 Aprendizagem de Máquina . . . 41
4.2 Aprendizado Supervisionado . . . 42
4.2.1 k-Vizinhos Mais Próximos . . . 43
4.2.2 1NN e o Diagrama de Voronoi . . . 44
4.2.3 Seleção e Construção de Protótipos . . . 44
4.3 Avaliação de Modelos . . . 45
4.3.1 Validação Cruzada . . . 47

4.3.2 Métricas Avaliativas 47
4.3.3 Comparação de Modelos 47

5 UM NOVO CLASSIFICADOR BASEADO EM SELEÇÃO CLONAL . . 50
5.1 A Base de Funcionamento do CSCDR 50
5.1.1 Metáfora Biológica 50
5.1.2 Inspiração e Melhorias 51
5.2 Parâmetros de Entrada 51
5.3 O Algoritmo 52
5.3.1 Inicialização 52
5.3.2 Valor de Aptidão 53
5.3.3 Poda 54
5.3.4 Adição de Novos Anticorpos 54
5.3.5 Clonagem e Mutação 55
5.3.6 Última Poda 55
5.3.7 Classificação 56

6 ESTUDO EXPERIMENTAL 57
6.1 Comportamento do Algoritmo 57
6.1.1 Conjuntos de Dados 57
6.1.2 Organização dos Experimentos 58
6.1.3 Resultados 59
6.2 Estudo Experimental com Bases de Dados de *Benchmark* 60
6.2.1 Bases de Dados 62
6.2.1.1 Conjunto de Dados Íris 62
6.2.1.2 Conjunto de Dados WDBC 62
6.2.1.3 Conjunto de Dados Ionosfera 62
6.2.1.4 Conjunto de Dados *E. coli* 63
6.2.1.5 Conjunto de Dados Vidros 63
6.2.1.6 Conjunto de Dados Diabetes 63
6.2.2 Métricas Comparativas 64
6.2.3 Metodologia 64
6.2.4 Seleção de Parâmetros 64

6.2.5 Resultados 65
6.2.6 Discussão 67

7 CONCLUSÕES E PROPOSTAS DE TRABALHOS FUTUROS 70
7.1 Contribuições 70
7.2 Conclusões 70
7.3 Proposta para Trabalhos Futuros 71

REFERÊNCIAS 72

GLOSSÁRIO 78

GLOSSÁRIO 78

LISTA DE ABREVIATURAS E SIGLAS

CSCDR	*Clonal selection classifier with data reduction*
EB	*Exabyte*
SIA	Sistema imunológico artificial
SI	Sistema imunológico
APC	*Antigen presenting cell*
NK	*Natural killer*
MHC	*Major histocompatibility complex*
BCR	*B-cell receptor*
TCR	*T-cell receptor*
NN	*Nearest neighbor*
TP	*True positive*
FP	*False positive*
TN	*True negative*
FN	*False negative*
CSA	*Clonal selection algorithm*
CLONALG	*Clonal selection algorithm*
CSCA	*clonal selection classifier algorithm*
BMU	*Best match unit*
AIRS	*Artificial immune recognition system*
ARB	*Artificial recognition balls*
RWTSAIRS	*Real world tournament selection AIRS*
WAIRS	*Weighted AIRS*
CLONCLAS	*Clonal classification*
SVM	*Support vector machine*
UMC	Unidade com maior correspondência
WEKA	*Waikato environment for knowledge analysis*

MLP	*Multi layer perceptron*
WDBC	*Wisconsin diagnostic breast cancer*
FNA	*Fine needle aspiration*

LISTA DE FIGURAS

Figura 2.1: Constituintes do sistema imunológico 20
Figura 2.2: Principais órgãos linfoides 22
Figura 2.3: Molécula de imunoglobulina 24
Figura 2.4: Representação de um espaço de formas 26
Figura 2.5: Princípio de funcionamento da seleção clonal 27
Figura 2.6: Respostas imunológicas 28
Figura 2.7: Resposta negativa e positiva 29

Figura 3.1: Framework em camadas para sistemas imunológicos artificiais 31

Figura 4.1: Interpretação gráfica de regressão. 42
Figura 4.2: Interpretação gráfica de classificação. 43
Figura 4.3: Classificação utilizando kNN com diferentes valores de k 44
Figura 4.4: Exemplo de um diagrama de Voronoi 45
Figura 4.5: Diagramas de Voronoi de diferentes métodos de seleção de instância . 46
Figura 4.6: Nomenclatura das instâncias, em relação à classe c_1. 48

Figura 5.1: Relação entre distância e afinidade 54

Figura 6.1: Base de dados sintética A 58
Figura 6.2: Base de dados sintética B 59
Figura 6.3: Células de memória geradas para o conjunto de dados A 60
Figura 6.4: Células de memória geradas para o conjunto de dados B 61
Figura 6.5: Diagramas de Voronoi mostrando a representatividade dos anticorpos 68

LISTA DE ALGORITMOS

3.1 Princípio de funcionamento do CLONALG 33
3.2 Princípio de funcionamento do CSCA 35
3.3 Versão simplificada do funcionamento do AIRS 38

5.1 Princípio de funcionamento do CSCDR 53

LISTA DE TABELAS

Tabela 4.1: Métricas avaliativas utilizadas na literatura 48

Tabela 5.1: Número de parâmetros de entrada de SIAs classificadores. 52

Tabela 6.1: Acurácias obtidas para as bases sintéticas. 60

Tabela 6.2: Distribuição das classes do conjunto de dados *E. coli*. 63

Tabela 6.3: Variação de parâmetros de entradas dos algoritmos testados. 65

Tabela 6.4: Variação do parâmetro N para cada base de dados 65

Tabela 6.5: Combinações de parâmetros utilizadas para cada algoritmo/base de dados. 66

Tabela 6.6: Comparação das acurácias médias obtidas para os algoritmos CSCDR, MLP, C4.5, *k*NN, CSCA e AIRS2 66

Tabela 6.7: Comparação das quantidades médias de protótipos utilizados nos testes pelos algoritmos baseados em instâncias 66

Tabela 6.8: Aptidão dos anticorpos da figura 6.2.6 69

RESUMO

O sistema imunológico dos vertebrados é extremamente complexo, sendo responsável por proteger o organismo contra agentes causadores de doenças. Para funcionar apropriadamente, é necessário que seus componentes reconheçam de forma eficaz os elementos patógenos, a fim de neutralizá-los, e também os elementos do próprio organismo, de forma a não reagirem a estes.

Estas e outras características são similares àquelas exigidas em soluções para problemas de engenharia e computação. Desta forma, os sistemas imunológicos artificiais utilizam a contraparte biológica como metáfora para o desenvolvimento de diversas ferramentas computacionais utilizadas nas mais diversas tarefas.

Esta dissertação utiliza os conceitos apresentados pelos sistemas imunológicos artificiais para o desenvolvimento de um novo algoritmo de aprendizado supervisionado, baseado principalmente no mecanismo de seleção clonal.

O método proposto neste trabalho, denominado *clonal selection classifier with data reduction* (CSCDR), utiliza uma função de aptidão com base no número de classificações corretas e incorretas apresentadas por cada anticorpo. O algoritmo tenta maximizar este valor através do processo de seleção clonal, envolvendo mutação, maturação de afinidade e seleção dos melhores indivíduos, transformando a fase de treinamento em um problema de otimização. Isto leva a anticorpos com maior representatividade e, portanto, diminui a quantidade de protótipos gerados ao final do algoritmo.

Experimentos em bases de dados sintéticas e bases de dados de problemas reais, utilizadas como *benchmark* para problemas de aprendizagem de máquina, demonstram a eficácia do algoritmo CSCDR como técnica de classificação.

Quando comparado a outros classificadores conhecidos da literatura, o CSCDR apresenta desempenho similar e, quando comparado a algoritmos baseados em instâncias, o mesmo utiliza menores quantidades de protótipos para representar os dados, mantendo o desempenho.

Palavras-chave: Sistema imunológico artificial, seleção clonal, classificação de dados.

ABSTRACT

The vertebrate immune system is an extremely complex system, being responsible for protecting the body against disease causing agents. To function properly, it is necessary its components effectively recognize the pathogens in order to neutralize them, and also elements of the body itself, so as not to react to these.

These and other features are similar to those required solutions to problems in engineering and computing. Thus, artificial immune systems use biological counterpart as a metaphor for development of several computational tools used in various tasks.

This dissertation uses the concepts presented by the artificial immune systems to develop a new supervised learning algorithm, based mainly on the mechanism of clonal selection.

The method proposed in this work, named clonal selection classifier with data reduction (CSCDR), uses a fitness function based on the number of correct and incorrect classifications made by each antibody. The algorithm tries to maximize this value through the clonal selection process, involving mutation, affinity maturation and selection of the best individuals, turning the training phase in an optimization problem. This leads to more representative antibodies and therefore decreases the amount of prototypes generated at the end of the algorithm.

Experiments on synthetic databases and real problem databases, used as benchmark to machine learning problems, demonstrate the effectiveness of the CSCDR algorithm as a classification technique.

When compared to other well known classifiers in literature, CSCDR shows similar performance and when compared to instance based algorithms, CSCDR utilizes a smaller amount of prototypes to represent the data maintaining the same performance.

Keywords: Artificial immune system, clonal selection, data classification.

1 INTRODUÇÃO

Este capítulo introduz e discute os principais aspectos que levaram ao desenvolvimento de um novo classificador de dados baseado nos princípios da imunologia. Os problemas considerados, as hipóteses, objetivos e estrutura da dissertação fazem parte deste capítulo.

1.1 Motivação

Atualmente vivenciamos uma era onde a informação é imprescindível. Criamos e consumimos informação em quantidades muito superiores ao passado. A quantidade total de dados armazenados no mundo, nos mais diversos meios digitais (discos rígidos de computadores, cartões de memória, CDs, DVDs e outros) e analógicos (livros, radiografias, fotografias, vídeo analógico e outros) cresceu de 2,6 *exabytes* (EB), com compressão ótima, em 1986 para 15,8 EB em 1993, 54,5 EB em 2000 e para 295 EB, com compressão ótima, em 2007 (HILBERT; LÓPEZ, 2011).

Entretanto, os dados armazenados precisam ser convertidos em informação e conhecimento para tornarem-se úteis. Tradicionalmente, a tarefa de extrair informações úteis era realizada por analistas, que interpretavam os dados para dali retirarem informações e conhecimentos implícitos. Contudo, com o salto gigantesco no volume de dados gerados, desenvolveu-se uma necessidade urgente por novas técnicas e ferramentas automatizadas, que pudessem inteligentemente auxiliar-nos na transformação de vastas quantidades de dados em informações úteis e conhecimento. Isto levou à geração de uma nova área de pesquisa em ciência da computação denominada de *aprendizagem de máquina* (MITCHELL, 1997).

Um dos problemas clássicos de aprendizagem de máquina é o aprendizado supervisionado, que compreende as tarefas de classificação e regressão. Na classificação, apresenta-se um conjunto de dados de treinamento, normalmente representado por tuplas, e seus respectivos rótulos (classes), definidos pelo domínio do problema, e o sistema deve gerar um modelo que generalize as informações e possa classificar corretamente novos dados não rotulados, pertencentes ao conjunto de teste. Na regressão o processo é semelhante. No lugar dos rótulos, porém, valores contínuos são utilizados.

Dentre os diversos algoritmos e métodos utilizados para classificação, muitos têm inspiração em metáforas da natureza. É o caso dos Sistemas Imunológicos Artificiais (SIA), baseados em princípios e teorias do sistema imunológico dos vertebrados, que têm

despertado o interesse de pesquisadores da computação nos últimos anos.

O sistema imunológico biológico é um sistema robusto, complexo e adaptativo que defende o corpo de agentes patógenos, utilizando diferentes mecanismos de resposta para neutralizar os efeitos patogênicos ou para destruir as células infectadas, dependendo do tipo de agente e da sua forma de entrada no organismo (AICKELIN; DASGUPTA, 2005). É composto por uma grande variedade de moléculas, células e órgãos espalhados pelo corpo todo e, ao contrário de outros sistemas como o circulatório e o nervoso, não possui um órgão de controle central.

Sua função principal é vistoriar o organismo, em busca de células com algum mal funcionamento pertencentes ao próprio corpo (células de tumor, por exemplo) ou elementos estrangeiros causadores de doença (vírus e bactérias, por exemplo). Cada elemento que pode ser reconhecido pelo sistema imunológico é chamado de antígeno. As células pertencentes ao próprio organismo e inofensivas ao seu funcionamento são denominadas antígenos próprios, enquanto os elementos causadores de doenças são denominados antígenos não-próprios. Sendo assim, o sistema imunológico deve ser capaz de distinguir entre antígenos próprios e não-próprios (DE CASTRO; TIMMIS, 2002b).

Tarefa semelhante é realizada pelos sistemas de classificação. O modelo gerado pelo sistema deve ser capaz de distinguir as várias classes do domínio do problema, de forma a rotular corretamente novos dados. Esta dissertação aproveita algumas das semelhanças entre os sistemas imunológicos e as ferramentas de aprendizado supervisionado e propõe um novo algoritmo de SIA para classificação de dados.

1.2 Trabalhos Relacionados

Um dos primeiros classificadores baseados em imunologia foi proposto por Watkins (2001), denominado AIRS (*artificial immune recognition system*). O algoritmo baseia-se nos princípios de seleção clonal e redes imunológicas, ambos provenientes da imunologia, para gerar protótipos (células de memória) que representem o conjunto de treinamento. Estes protótipos são utilizados como referência por um classificador baseado nos vizinhos mais próximos, que realiza a classificação de novas instâncias.

Desde então, diversos outros algoritmos e métodos foram propostos. Os trabalhos de Watkins, Timmis e Boggess (2004), Watkins e Timmis (2004), Secker e Freitas (2007) e Golzari et al. (2009) utilizam o AIRS como base e propõe melhorias ou adaptações ao algoritmo original. Baseando-se ainda no princípio de seleção clonal, alguns autores propuseram modificações no algoritmo CLONALG de de Castro e Von Zuben (2002) a fim de aplicá-lo à classificação de dados (WHITE; GARRETT, 2003; BROWNLEE, 2005).

Além de seleção clonal, o princípio da seleção negativa também é utilizado em classificadores imuno-inspirados. M-NSA (MARKOWSKA-KACZMAR; KORDAS, 2006), MINSA (MARKOWSKA-KACZMAR; KORDAS, 2008), ANCS (IGAWA; OHASHI, 2009) e RNS (OLIVEIRA; DRUMMOND, 2011) são alguns dos algoritmos que utilizam este princípio para tarefas de classificação.

1.3 Hipótese e Objetivos

Normalmente, algoritmos de SIA para classificação utilizam estruturas denominadas células de memória (protótipos) para representar os dados de treinamento (DE CASTRO; VON ZUBEN, 2000; WATKINS, 2001; BROWNLEE, 2005). Quando novas instâncias não rotuladas são apresentadas ao algoritmo, verifica-se qual o protótipo mais semelhante a cada instância, classificando-a com o mesmo rótulo. Se o rótulo atribuído corresponde à classificação real, diz-se que a instância foi classificada corretamente.

Uma taxa de representatividade pode ser definida para cada protótipo, proporcional ao número de instâncias corretamente representadas por ele e inversamente proporcional aos erros de classificação. Desta forma, a tarefa de geração das células de memória pode ser tratada como um problema de otimização, onde a representatividade média dos protótipos deve ser maximizada (BROWNLEE, 2005).

A hipótese deste trabalho é a de que podemos aumentar a taxa de representatividade das células de memória, geradas pelos algoritmos de SIA para classificação de dados, através de um controle do tamanho máximo da sua população. Com isto, uma quantidade menor de protótipos seria capaz de representar corretamente um maior número de instâncias.

A partir da hipótese apresentada, o objetivo principal deste trabalho é prover um classificador, CSCDR (*clonal selection classifier with data reduction*), que apresente estas características e verificar seu desempenho em relação a acurácia e número de células de memória geradas. Bases de dados simuladas e bases de dados de problemas reais, utilizadas por outros trabalhos da literatura como *benchmark*, são utilizadas para avaliar o desempenho do algoritmo proposto e investigar o comportamento das células de memória geradas.

1.4 Organização da Dissertação

Além da introdução, esta dissertação conta com mais seis capítulos, organizados como se segue:

- O Capítulo 2 apresenta uma discussão a respeito dos principais componentes, teorias e princípios da imunologia, essenciais para o desenvolvimento de sistemas imunológicos artificiais.
- O Capítulo 3 investiga alguns sistemas imunológicos artificiais para classificação de dados baseados em seleção clonal, utilizados como inspiração para o modelo proposto nesta dissertação.
- O Capítulo 4 provê uma fundamentação sucinta a respeito de aprendizagem de máquina, aprendizado baseado em instância e métodos para comparação de modelos.
- O Capítulo 5 apresenta o algoritmo CSCDR e discute fatores importantes a respeito do funcionamento do método.
- O Capítulo 6 investiga o funcionamento do CSCDR em conjuntos de dados sintéticos e reais e compara os resultados obtidos com os resultados obtidos por outros

algoritmos classificadores.

- Finalmente, o Capítulo 7 conclui a dissertação com a apresentação das conclusões, contribuições e propostas de trabalhos futuros.

2 SISTEMA IMUNOLÓGICO BIOLÓGICO

Imunidade é definida como o estado de um organismo que resiste a infecções. O conjunto de células, tecidos e moléculas responsáveis pela imunidade é denominado *Sistema Imunológico* (SI) e a resposta coordenada destas células e moléculas a agentes infecciosos é denominada *resposta imune* (ABBAS; LICHTMAN, 2004).

O principal objetivo do SI humano é inspecionar constantemente o organismo à procura de células com mau comportamento. Para tanto, uma série de células, moléculas e outras estruturas do SI devem ser capazes de identificar antígenos patogênicos de forma eficiente e responder de maneira rápida a infecções.

Nesta seção, alguns conceitos a respeito do sistema imunológico serão abordados, dando ênfase às teorias e ideias utilizadas como inspiração para esta dissertação.

2.1 Constituintes do Sistema Imunológico

O sistema imunológico pode ser divido em dois: o *sistema imune inato*, responsável por uma resposta imunológica rápida e efetiva, e o *sistema imune adaptativo*, um pouco mais lento, porém mais eficaz. Ambos são mediados pelas *células brancas* ou *leucócitos*, que subdividem-se segundo a figura 2.1.

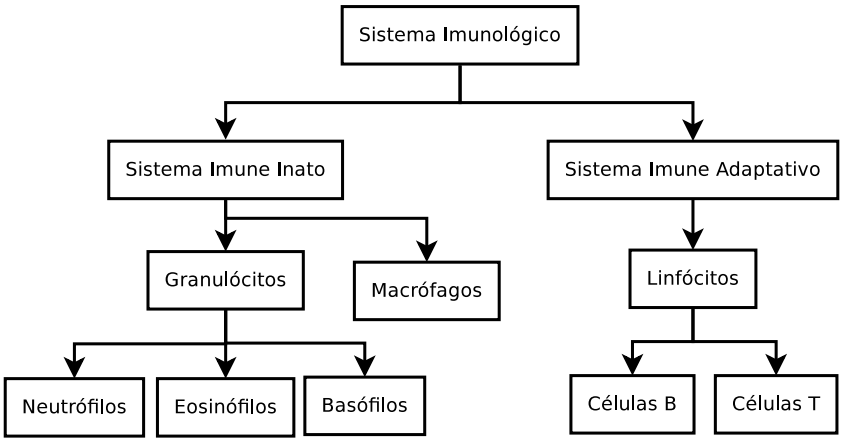


Figura 2.1: Constituintes do sistema imunológico. Adaptado de (DE CASTRO; TIMMIS, 2002a).

As células do sistema imune inato estão imediatamente disponíveis para o combate

contra uma ampla variedade de patógenos, sem exigir prévia exposição aos mesmos, e atuam do mesmo modo em todos os indivíduos saudáveis. Os *macrófagos* e *neutrófilos* possuem a capacidade de ingerir microrganismos ou partículas (*fagocitose*). O macrófago também possui a habilidade de apresentar antígenos a outras células, sendo portanto denominado de *célula apresentadora de antígeno* (APC - *antigen presenting cell*). Os *granulócitos* ou *leucócitos polimorfonucleares* constituem um grupo de células com núcleos multilobulados contendo grânulos citoplasmáticos. São divididos em *neutrófilos*, *eosinófilos* e *basófilos*. Os neutrófilos são fagocitários e exercem importante papel na ingestão e na morte de germes extracelulares, os eosinófilos são importantes especialmente na defesa contra infecções parasitárias e os basófilos acredita-se ter papel nas reações alérgicas (MURPHY; TRAVERS; WALPORT, 2010).

O sistema imune adaptativo consiste de linfócitos (com exceção dos linfócitos NK) e seus produtos, tais como os anticorpos. Os linfócitos são as únicas células que possuem receptores específicos para os antígenos em sua superfície. São gerados a partir das células-tronco na medula óssea e divididos em três principais grupos, os *linfócitos T* que amadurecem em um órgão denominado *Timo*, os *linfócitos B* que amadurecem na própria *medula óssea* (em inglês, *bone marrow*) e as células exterminadoras naturais (NK - *natural killer*). Apesar de serem morfologicamente semelhantes, são bastante diferentes em relação à função que desempenham (ABBAS; LICHTMAN, 2004).

As células B estão relacionadas com a imunidade humoral. Possuem receptores em sua membrana que podem ligar-se com antígenos solúveis ou antígenos na superfície de microrganismos ou outras células e gerar respostas imunológicas. Além disso, as células B são as únicas células capazes de produzir anticorpos. Anticorpos (ou imunoglobulinas) são proteínas responsáveis por neutralizar e eliminar microrganismos e suas toxinas.

As células T são as células responsáveis pela imunidade celular. Seus receptores só reconhecem fragmentos de peptídeos de antígenos proteicos que são apresentados por moléculas especializadas denominadas complexo principal de histocompatibilidade (MHC - *major histocompatibility complex*). As *células T auxiliares* secretam proteínas denominadas citocinas cujas funções são estimular a proliferação e diferenciação das células T e ativar outras células, incluindo células B, macrófagos e outros leucócitos. Já as *células T citotóxicas*, podem matar outras células que abrigam microrganismos intracelulares.

Linfócitos NK são mediadores do sistema imune inato, sendo capazes de reconhecer e matar algumas células anormais, como, por exemplo, células tumorais e células infectadas com o vírus da herpes (MURPHY; TRAVERS; WALPORT, 2010).

2.2 Anatomia do Sistema Imunológico

O sistema imunológico consiste de diferentes órgãos e tecidos distribuídos pelo corpo. Estes órgãos podem ser classificados funcionalmente em dois grupos principais, os órgãos linfoides *primários*, que proveem microambientes apropriados para o desenvolvimento e maturação dos linfócitos, e órgãos linfoides *secundários*, para onde são direcionados os antígenos de tecidos ou espaços vasculares de outras partes do corpo, para que os linfócitos maduros possam interagir efetivamente com eles (GOLDSBY et al., 2003). Estes órgãos fazem parte do sistema linfático.

O timo e a medula óssea são os órgãos linfoides primários (centrais), onde ocorre

a maturação das células T e B, respectivamente. As respostas imunes do sistema adaptativo ocorrem nos órgãos linfoides secundários (periféricos), podendo ser classificados de acordo com a região do corpo que defendem. O baço responde predominantemente a antígenos transportados pelo sangue. Os linfonodos respondem a antígenos circulando na linfa, entrando através da pele (linfonodos subcutâneos) ou pelas superfícies das mucosas. Amígdalas, placas de Peyer e outros tecidos linfoides associados a mucosas reagem a antígenos que entram pela superfície das barrerias mucosas. A medula óssea também é classificada como órgão linfoide secundário, pois ela dá origem a células B e NK, além de servir como local de diferenciação terminal para as células B. Os órgãos linfoides são apresentados pela figura 2.2.

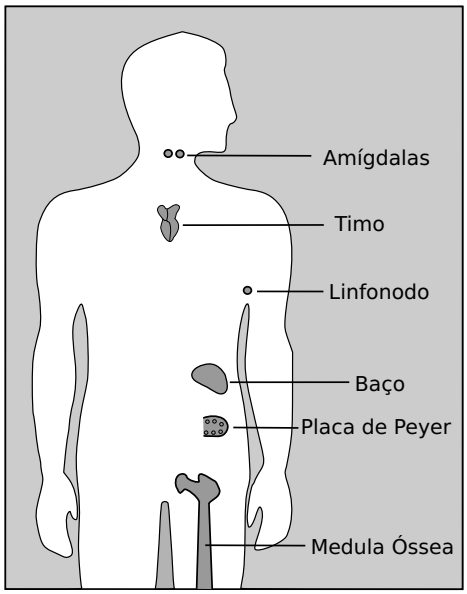


Figura 2.2: Principais órgãos linfoides

O sistema imunológico constitui-se de múltiplas camadas com vários mecanismos de interação entre si e com os agentes patógenos (MALE et al., 2006; MURPHY; TRAVERS; WALPORT, 2010). A principal distinção ocorre entre o sistema imune inato, que compreende os mecanismos de defesa presentes no organismo do indivíduo desde o seu nascimento e pré-existentes à entrada de qualquer antígeno, e o sistema imune adaptativo, que adapta-se ao longo da vida do indivíduo, na presença de microrganismos invasores:

- *Sistema Imune Inato*: media a primeira proteção contra muitos microrganismos comuns. É formado por células fagocitárias, como macrófagos e neutrófilos, além de fatores solúveis como o complemento e algumas enzimas. As células do sistema imune inato desempenham um papel crucial na iniciação e posterior direcionamento das respostas imunes adaptativas, principalmente devido ao fato de que as respostas adaptativas demoram certo período de tempo para exercer seus efeitos. O nome *inato* refere-se ao fato de que este tipo de defesa está sempre presente em indivíduos saudáveis (ABBAS; LICHTMAN, 2004).

- *Sistema Imune Adaptativo*: é capaz de se prevenir contra qualquer tipo de antígeno que possa ser encontrado. Os linfócitos são as principais células do sistema imune adaptativo. São capazes de desenvolver uma memória imunológica, ou seja, reco-

nhecer o mesmo estímulo antigênico caso ele entre novamente em contato com o organismo.

A primeira linha de defesa da imunidade inata é provida pelas barreiras epiteliais, células especializadas e antibióticos naturais presentes no epitélio (tecido que reveste as superfícies externas e internas), com função de bloquear a entrada de micróbios. Caso os microrganismos consigam atravessar o epitélio e cheguem aos tecidos ou à circulação, são atacados por fagócitos, células exterminadoras naturais e proteínas do plasma, incluindo as proteínas do sistema complemento. Além de prover uma defesa mais imediata, as respostas do sistema imune inato melhoram a resposta do sistema imune adaptativo contra agentes infecciosos (ABBAS; LICHTMAN, 2004).

Apesar de o sistema imune inato prover um combate efetivo contra muitas infecções, microrganismos patogênicos tem evoluído para lhe resistir. A defesa contra estes agentes infecciosos é tarefa do sistema imune adaptativo, representado pelos linfócitos e pelos seus produtos, tais como anticorpos. Enquanto os mecanismos do sistema imune inato reconhecem estruturas comuns a classes de micróbios, os linfócitos apresentam receptores que reconhecem, especificamente, diferentes substâncias produzidas por microrganismos.

Devido à forte ligação dos sistemas imunológicos artificiais com as teorias do sistema imune adaptativo, este será descrito em mais detalhes na seção seguinte.

2.3 Sistema Imune Adaptativo

A imunidade adaptativa é capaz de reconhecer e eliminar seletivamente microrganismos específicos e moléculas. Ao contrário da resposta imune inata, as respostas imunes adaptativas não são as mesmas em todos os membros de uma espécie, mas são reações ao contato com antígenos específicos (GOLDSBY et al., 2003).

É dividida em *imunidade humoral* e *imunidade mediada por células*. A imunidade humoral é mediada por anticorpos, produzidos pelos linfócitos B. Anticorpos são secretados nos fluidos corporais (circulação e fluidos das mucosas), também denominados humores, para neutralizar e eliminar microrganismos e toxinas microbiais presentes no sangue ou nas cavidades dos órgãos mucosos, tais como os tratos gastrointestinal e respiratório (ABBAS; LICHTMAN, 2004). A imunidade humoral é o principal mecanismo de defesa contra micróbios extracelulares e suas toxinas (ABBAS; LICHTMAN; PILLAI, 2007).

Os anticorpos não tem acesso aos microrganismo que vivem no interior celular. A defesa contra estes patógenos é provida pela imunidade mediada por células, realizada pelos linfócitos T. Alguns linfócitos T ativam fagócitos para destruir os micróbios ingerindo-os. Outras células T matam qualquer tipo de célula hospedeira que esteja abrigando algum agente patógeno no citoplasma. Os anticorpos são gerados de forma a reconhecer especificamente antígenos microbiais extracelulares, enquanto os linfócitos T reconhecem antígenos produzidos por micróbios intracelulares.

2.4 Reconhecimento Antigênico

Antígenos são geralmente muito grandes e complexos e não são reconhecidos inteiramente pelas células B e T. Em vez disso, apenas certas partes do antígeno são reconhecidas, denominadas *epítopos* ou *determinantes antigênicos*. Epítopos são as regiões imunologicamente ativas em um antígeno complexo, onde ocorre a ligação entre o antígeno e o receptor da célula B ou T.

O receptor das células B (BCR - *B cell receptor*) é constituído por proteínas, que iniciam os eventos sinalizadores, e uma molécula de anticorpo capaz de reconhecer epítopos de antígenos livres. Enquanto os anticorpos possuem um único tipo de receptor, os antígenos podem possuir múltiplos epítopos diferentes, podendo ser reconhecido por diferentes anticorpos (DE CASTRO, 2001).

A molécula de anticorpo é composta por duas regiões distintas. Uma *região constante* que pode assumir uma de apenas quatro ou cinco formas distintas bioquimicamente e uma *região variável* que pode tomar uma variedade aparentemente infinita de formas sutilmente diferentes, que permitem que os anticorpos se liguem especificamente a uma variedade igualmente grande de antígenos distintos (MURPHY; TRAVERS; WALPORT, 2010). A figura 2.4 apresenta uma molécula de imunoglobulina com suas regiões variáveis (V) e constantes (C).

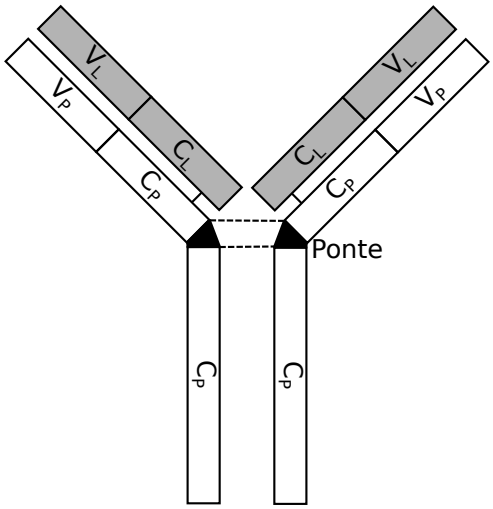


Figura 2.3: Molécula de imunoglobulina, formada por regiões constantes (C) e variáveis (V) com cadeias leves (L) e cadeias pesadas (P). A ligação entre as partes da molécula são realizadas por uma ponte dissulfeto. Adaptado de (GOLDSBY et al., 2003).

Ao contrário do BCR, o receptor da célula T (TCR - *T cell receptor*) necessita que os antígenos sejam previamente processados (fragmentos na forma de peptídeos) e ligados a moléculas de MHC, antes de serem reconhecidos.

2.4.1 Especificidade Antigênica

Cada *linfócito virgem* que penetra na corrente circulatória é portador de receptores de antígeno com especificidade única, determinada por um mecanismo de rearranjo genético especial que atua durante o desenvolvimento linfocitário, a fim de gerar centenas de diferentes variantes dos genes codificadores das moléculas receptoras. Assim, embora um

linfócito seja portador de um receptor de especificidade única, a especificidade de cada célula linfocitária é diferente, e os milhões de linfócitos do organismo podem apresentar milhões de especificidades distintas (DE CASTRO, 2001).

A *especificidade antigênica* permite que o sistema imune adaptativo identifique diferenças sutis entre antígenos, de forma que os anticorpos possam distinguir moléculas de proteínas que diferem entre si por um único aminoácido (GOLDSBY et al., 2003). Estima-se que o número total de especificidades antigênicas dos linfócitos em um indivíduo, denominado repertório linfocitário, seja capaz de discriminar entre 10^7 e 10^9 epítopos diferentes (ABBAS; LICHTMAN; PILLAI, 2007).

2.4.2 Espaço de Formas

Segundo Perelson e Oster (1979), para que as moléculas de anticorpo e os antígenos possam interagir, devem existir extensivas regiões de complementaridade em suas superfícies. O conjunto de características destas regiões, relevantes para a ligação entre as moléculas, é denominado *forma generalizada* (*generalized shape*) (PERELSON, 1989).

Considerando que possamos descrever a forma generalizada do paratopo de um anticorpo ou de um determinante antigênico qualquer através de N parâmetros (tamanho das regiões ligantes, carga, etc), organizados em um vetor $m = \langle m_N, m_{N-1}, ..., m_2, m_1 \rangle$; desta forma, um ponto em um espaço Euclidiano N-dimensional, denominado *espaço de formas* (*shape space*), especifica a forma generalizada de um paratopo ou de um determinante antigênico quaisquer. Espera-se que estes pontos localizem-se dentro de um volume V do espaço uma vez que as características das regiões ligantes são restritas a certos valores.

Assume-se que cada anticorpo interage especificamente com todos os antígenos cujo complemento encontra-se dentro de uma pequena região denominada *região de reconhecimento*. Estas regiões, de volume V_ε, definido por um limiar de reatividade-cruzada ε, são apresentadas na figura 2.4.2.

2.5 Seleção Clonal

Como cada célula linfocitária apresenta um receptor com especificidade diferente, o número de linfócitos capazes de se ligar a um dado antígeno é limitado. Entretanto, quando um antígeno entra no organismo, o SI seleciona as células específicas e ativa-as. Este conceito, denominado *seleção* (ou *expansão*) *clonal*, foi sugerido pela primeira vez por Jerne (1955) e enunciado mais claramente por Burnet (1957), como uma hipótese para explicar como o sistema imunológico poderia responder a uma grande variedade de antígenos (ABBAS; LICHTMAN; PILLAI, 2007).

Durante o processo, um antígeno liga-se a uma célula B ou T particular, ativando-a. Estas células passam por um processo de transformação estrutural, transformando-se em linfoblastos que começam a dividir-se duas a quatro vezes a cada 24 horas, durante três a cinco dias, de forma que o linfócito original dá origem a uma prole de cerca de 1000 clones de especificidade idêntica. Então, estes diferenciam-se em células efetoras. No caso das células B, as células efetoras são denominadas *plasmócitos* e secretam anticorpos em altas taxas; no caso das células T, as células efetoras são capazes de destruir células

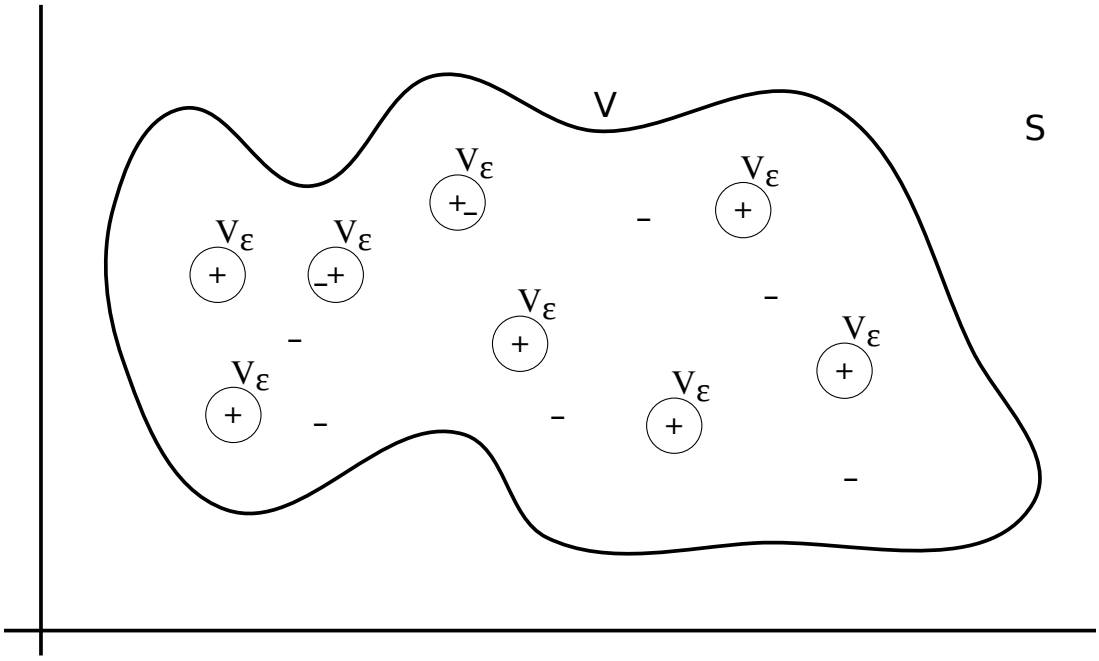


Figura 2.4: Representação de um espaço de formas S. Existe um volume V onde as formas dos paratopos $(+)$ e dos complementos dos determinantes antigênicos $(-)$ estão localizadas. Um anticorpo pode reconhecer qualquer antígeno cujo complemento encontra-se dentro do volume V_ε.

infectadas ou ativar outras células do SI (MURPHY; TRAVERS; WALPORT, 2010).

Alguns dos linfoblastos B e T podem diferenciar-se em *células de memória* de vida longa, garantindo imunidade a longo prazo contra o agente patógeno que iniciou a seleção clonal (GOLDSBY et al., 2003).

Durante o processo de seleção clonal, os clones dos linfócitos podem apresentar alta afinidade a células e elementos do próprio organismo (antígenos próprios). Estes, denominados células auto-reativas, são removidos durante seu desenvolvimento para evitar respostas autoimunes (ABBAS; LICHTMAN; PILLAI, 2007). Desta forma, a seleção clonal provê um mecanismo de reconhecimento próprio/não-próprio. A figura 2.5 ilustra o funcionamento da seleção clonal.

A seleção clonal pode ser vista como um microcosmo da evolução darwiniana. Os linfócitos multiplicam-se em altas taxas e passam por um processo de variação natural, provido pelas regiões genéticas variáveis, responsável pela produção de populações de anticorpos altamente diversificadas. Um tipo de seleção natural ocorre quando apenas os anticorpos capazes de ligarem-se aos antígenos reproduzem-se (CZIKO, 1997; ADAMS, 1996).

2.6 Memória Imunológica

Quando o SI é exposto a um dado antígeno, alguns linfócitos diferenciam-se em células de memória, como foi visto na seção anterior. Estas células circulam pelo sangue, vasos linfáticos e tecidos, prontos para responder a novas exposições a este mesmo antígeno, ao invés de "partir do começo" a cada reexposição. Isto garante que a velocidade e

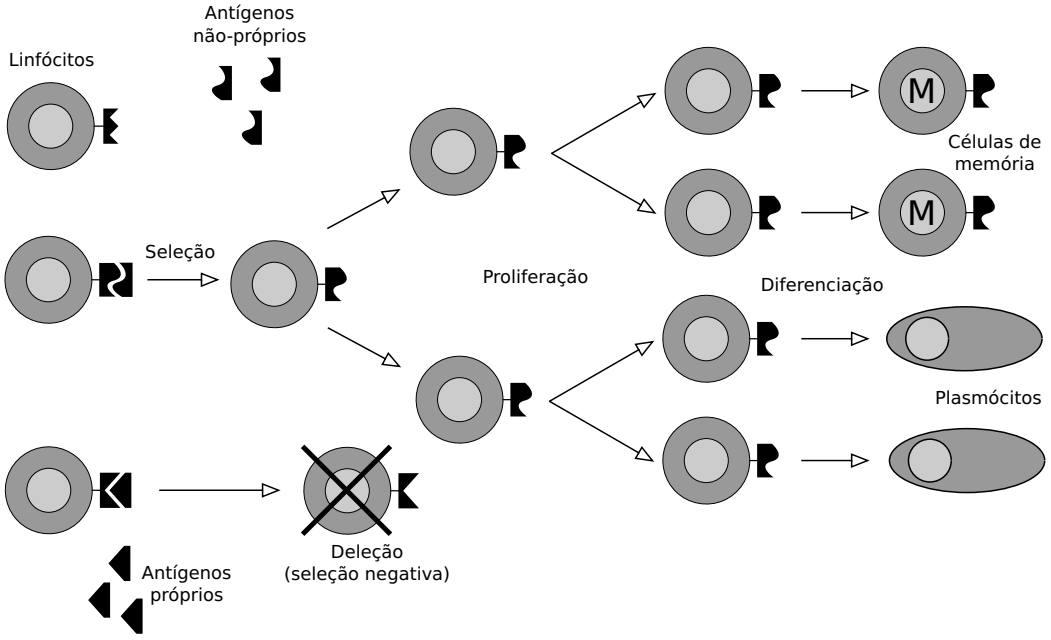


Figura 2.5: Princípio de funcionamento da seleção clonal. Os linfócitos que reconhecem antígenos próprios são eliminados. Aqueles que reconhecem os antígenos não-próprios são estimulados a se proliferar e diferenciar em células de memória e plasmócitos. Adaptado de (DE CASTRO; TIMMIS, 2002a).

eficácia da resposta imunológica se aperfeiçoe após cada infecção.

Para ilustrar o funcionamento deste mecanismo de *memória*, considera-se que o organismo é exposto a um antígeno *X*. Poucos anticorpos específicos a *X* estão presentes na circulação. Entretanto, após um período de latência, a quantidade de linfócitos anti-*X* começa a aumentar em concentração e afinidade gradativamente até um certo nível, de forma a responder à exposição antigênica. Assim que a infecção é eliminada sua concentração começa a cair. Este primeiro encontro com um antígeno estranho é denominado *resposta primária*.

Após algum tempo, o organismo é exposto novamente ao antígeno *X*, juntamente com um novo antígeno *Y*, diferente de *X*. Como o organismo está sendo exposto pela primeira vez ao antígeno *Y*, este desencadeia uma resposta primária. No caso do antígeno *X*, as células de memória remanescentes da primeira infecção possibilitam uma resposta mais rápida (menor período de latência) e mais eficiente (maior concentração de anticorpos), denominada *resposta secundária*.

Outra característica do SI é a capacidade de generalizar. Quando o organismo é exposto a *X'*, um novo antígeno estruturalmente parecido com *X*, os linfócitos anti-*X* são ativados e ocorre uma resposta imune semelhante a resposta secundária, denominada *resposta reativa cruzada* (DE CASTRO, 2001). A figura 2.6 apresenta estas diferentes respostas imunes.

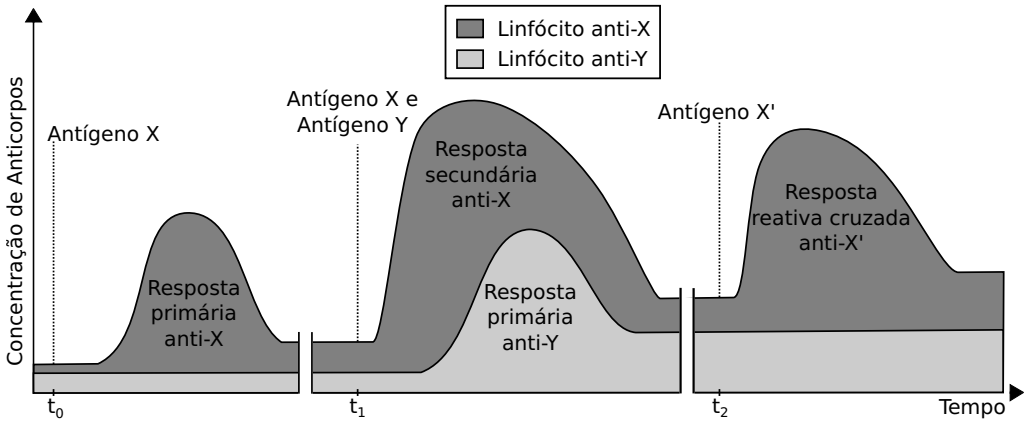


Figura 2.6: Resposta imunológica primária (t_0), secundária (t_1) e reativa cruzada (t_2) para três antígenos diferentes (*X*, *Y* e *X'*). Adaptado de (ABBAS; LICHTMAN; PILLAI, 2007).

2.7 Maturação de Afinidade

O mecanismo para geração de anticorpos de alta afinidade envolve mudanças sutis na estrutura das regiões variáveis dos anticorpos (regiões V). Estas mudanças ocorrem através do processo de *hipermutação somática* nos linfócitos B estimulados por um antígeno, que geram novas variações da região V. Algumas destas variações apresentam maior afinidade com o antígeno que a região V original. As células B com maior afinidade são preferencialmente escolhidas e acabam tornando-se dominantes a cada exposição subsequente (ABBAS; LICHTMAN; PILLAI, 2007). As mutações ocorrem com frequência cerca de cem mil vezes maior que as mutações espontâneas (daí o nome *hipermutação*) (GOLDSBY et al., 2003).

Este processo de aumento da afinidade média da população de anticorpos ao longo das exposições é denominado de *maturação de afinidade*.

2.8 Não Reatividade a Antígenos Próprios

O sistema imunológico é capaz de reconhecer tanto antígenos próprios como não-próprios. Contudo, é imprescindível que apenas antígenos não-próprios ativem respostas imunes, a fim de evitar que células e outros constituintes do organismo sejam atacados pelo SI.

Grande parte da capacidade de *tolerar* antígenos próprios surge durante o desenvolvimento linfocitário, onde os linfócitos imaturos são expostos a componentes próprios. Antígenos que não tenham sido expostos aos linfócitos imaturos, durante este período crítico, podem ser mais tarde reconhecidos como não-próprios pelo sistema imunitário, desencadeando uma resposta imune (GOLDSBY et al., 2003). Quando um linfócito reconhece um elemento próprio, ele é selecionado negativamente.

A *seleção negativa* é o processo pelo qual os linfócitos em desenvolvimento, que apresentam receptores antigênicos auto-reativos, são eliminados, contribuindo para a não reatividade própria. No timo, por exemplo, grandes quantidades de APCs apresentam complexos MHC-próprios aos linfócitos T em desenvolvimento (timócitos). Aqueles ti-

mócitos que expressam receptores de alta afinidade para os ligantes MHC-próprios são eliminados, resultando na seleção negativa do repertório de células T. Este processo elimina os linfócitos T auto-reativos potencialmente mais perigosos para o organismo e é um dos mecanismos que garante que o SI não responda a antígenos próprios, uma propriedade denominada *auto-tolerância*.

2.9 Rede Imunológica

Jerne (1974) propôs uma teoria a respeito de um mecanismo que regula as respostas imunes adaptativas, denominada *teoria da rede imunológica*. Esta ideia é baseada na demonstração de que os animais podem ser estimulados a produzir anticorpos capazes de reconhecer partes de anticorpos produzidos por outros animais da mesma espécie ou raça. Segundo Jerne (1974), a partir disto é razoável assumir que, dentro do sistema imunológico de um indivíduo, qualquer molécula de anticorpo pode ser reconhecida por um conjunto de outros anticorpos. Esta teoria foi refinada e formalizada nos trabalhos posteriores de Farmer, Packard e Perelson (1986), Perelson (1989) e Bersini e Varela (1994).

A porção da molécula de anticorpo responsável por reconhecer (complementarmente) um epítopo é denominada *paratopo*, o conjunto de epítopos exibido pelas regiões variáveis de um conjunto de moléculas de anticorpo é denominado *idiotipo* e cada epítopo idiotípico único é denominado *idiotopo*.

A teoria sugere que o SI é composto de uma rede enorme e complexa de paratopos, que reconhece conjuntos de idiotopos, e de idiotopos que são reconhecidos por conjuntos de paratopos. Assim, cada elemento pode tanto ser reconhecido como pode reconhecer. Depois de um dado anticorpo reconhecer um epítopo ou um idiotopo, ele pode responder positivamente ou negativamente a esse sinal de reconhecimento. Uma resposta positiva resulta na ativação e proliferação da célula e na secreção de anticorpos, enquanto uma resposta negativa leva à tolerância e supressão (DE CASTRO; TIMMIS, 2002a). A figura 2.9 apresenta as respostas positiva e negativa em uma rede imunológica.

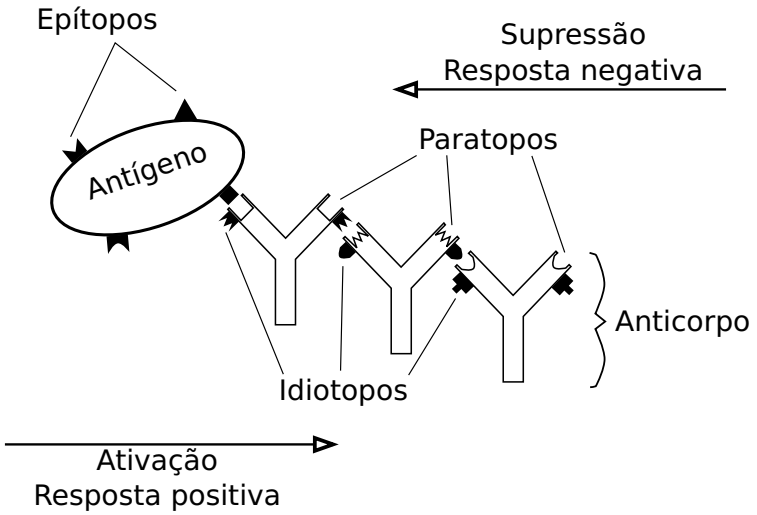


Figura 2.7: Respostas negativa e positiva como resultado da interação paratopo/epítopo e paratopo/idiotopo. Adaptado de (DE CASTRO; TIMMIS, 2002a).

3 SISTEMAS IMUNOLÓGICOS ARTIFICIAIS

Sistemas imunológicos artificiais (SIA) é um campo de estudo devotado ao desenvolvimento de modelos computacionais baseados nos princípios do sistema imunológico biológico. É uma área emergente que explora e emprega diferentes mecanismos imunológicos para resolver problemas computacionais. A poderosa capacidade de processamento de informação do sistema imunológico, tal como extração de características, reconhecimento de padrões, memória e sua natureza distributiva proveem metáforas ricas para sua contraparte artificial (DASGUPTA; NIÑO, 2008).

Os sistemas imunológicos artificiais são aplicados a diversas áreas. Uma revisão bibliográfica dos desenvolvimentos da área nos últimos anos pode ser encontrada nos trabalhos de Garrett (2005), Timmis et al. (2008), Dasgupta, Yu e Nino (2011) e Ulutas e Kulturel-Konak (2011). Dentre as principais aplicações de SIA, destacam-se (DE CASTRO, 2001; DE CASTRO; TIMMIS, 2002a; HART; TIMMIS, 2008):

- *Clustering* (agrupamento), classificação de dados e outros métodos de aprendizado;
- Detecção de falhas e anomalias;
- Sistemas baseados em agente;
- *Scheduling*;
- Aprendizagem de máquina;
- Controle e navegação autônoma;
- Métodos de busca e otimização;
- Processamento de imagens;
- Bioinformática;
- Vida artificial e
- Segurança de sistemas de informação.

Dentre estas, uma das aplicações mais conhecidas e estudadas é a classificação de dados. Diversos algoritmos de SIA já foram propostos e aplicados em tarefas de classificação de dados nas mais diversas áreas, incluindo previsão de ações da bolsa de valores

(BUTLER; KAZAKOV, 2010), detecção de fraudes (BRABAZON et al., 2010), pontuação de crédito de consumidores (HING; CHEONG; CHEONG, 2011), diagnóstico clínico (ZHAO; DAVIS, 2011; DELIBASIS et al., 2009), concursos de classificação de dados (OLIVEIRA; DRUMMOND, 2010) ou para propósitos gerais, em testes com bases de dados sintéticas ou da literatura (DE CASTRO; VON ZUBEN, 2000; WATKINS, 2001; SECKER; FREITAS, 2007; GOLZARI et al., 2009; IGAWA; OHASHI, 2009; CARTER, 2000).

Neste capítulo, são apresentados os principais classificadores baseados no princípio de seleção clonal, destacando os principais aspectos que serviram como inspiração para o desenvolvimento do CSCDR.

3.1 Considerações Iniciais

Com intuito de criar uma base comum para o campo de SIA, de Castro e Timmis (2002a) propuseram uma ideia de *framework* construído sobre três elementos básicos:

- Uma forma de representação para os componentes do sistema;
- Um conjunto de mecanismos para avaliar as interações dos indivíduos entre si e com o ambiente; e
- Procedimentos de adaptação que governam a dinâmica do sistema, isto é, como seu comportamento varia através do tempo.

O *framework* pode ser imaginado como uma abordagem em camadas, conforme apresentado na figura 3.1, onde a composição das camadas está diretamente relacionada ao domínio da aplicação. Normalmente, as soluções candidatas para o problema são representadas como anticorpos e o problema é representado como um conjunto de antígenos. A medida de afinidade é utilizada para medir o desempenho das soluções e o algoritmo imunológico é responsável por adaptar estas soluções. Ao final do SIA, as soluções devidamente melhoradas são apresentadas.

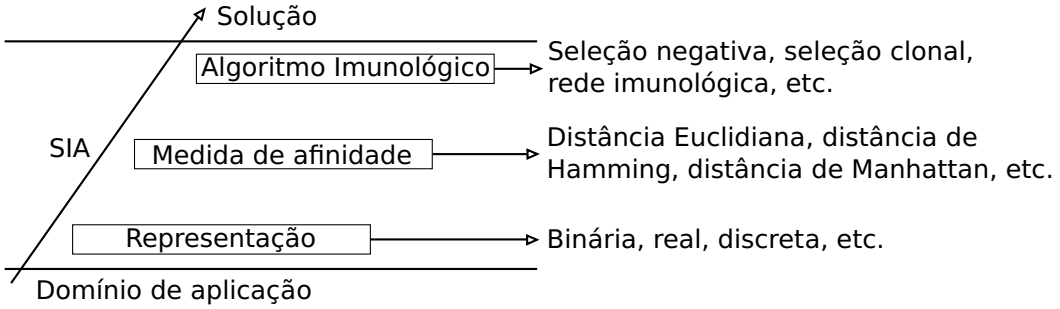


Figura 3.1: Framework em camadas para SIAs. Adaptado de (DE CASTRO; TIMMIS, 2002a)

No caso particular dos algoritmos de classificação, normalmente os antígenos representam os dados dos conjuntos de treinamento e teste, utilizados como entrada para o classificador, e os anticorpos representam centroides responsáveis por descrever o conjunto de entrada.

Neste trabalho, os problemas de classificação são definidos em espaços S^d, constituídos por um conjunto de instâncias de treinamento Ag e um conjunto de instâncias de teste Ag'. Ao final da fase de treinamento, o algoritmo retorna um conjunto de centroides Ab, utilizados para classificar os elementos de Ag' durante a fase de teste. Desta forma, define-se:

- *Representação*: Os antígenos (instâncias de entrada) e anticorpos (centroides) são compostos por um rótulo $rotulo \in \{1, 2, ..., nc\}$ e um vetor de atributos $atributos = \langle at_1, at_2, ..., at_d \rangle$, onde $at_i \in S$ e nc é o número total de classes do conjunto de dados. O vetor de atributos representa a forma generalizada do anticorpo ou antígeno (Seção 2.4.2) e define um ponto no espaço de formas S^d.

- *Medida de afinidade*: A afinidade entre um antígeno e um anticorpo é definida com base na medida da distância entre eles. A medida de afinidade realiza um mapeamento da interação entre dois vetores de características em um valor real que corresponde a sua afinidade ou grau de correspondência, $S^d \times S^d \rightarrow \mathbb{R}$. Para problemas definidos em $\mathbb{R}^d$, normalmente utiliza-se a distância Euclidiana (equação 4.1) ou de Manhattan (equação 4.2) e para problemas definidos em espaços discretos, onde S é um alfabeto finito, normalmente utiliza-se a distância de Hamming (equação 4.3). Outras medidas de distância também podem ser utilizadas.

- *Algoritmo imunológico*: Os algoritmos descritos nesta seção, assim como o classificador desenvolvido neste trabalho, são baseados no princípio da seleção clonal. Alguns destes algoritmos e técnicas serão descritos nas próximas seções.

3.2 CLONALG

A análise das propriedades computacionais do sistema imunológico realizada por Chowdhury, Stauffer e Choudary (1990) e Chowdhury e Stauffer (1992) sugeriu a possibilidade de aplicar o princípio de seleção clonal em inteligência computacional. Os trabalhos preliminares de Hightower, Forrest e Perelson (1996) e Forrest et al. (1993) consideraram a seleção clonal do ponto de vista do efeito Baldwin e reconhecimento de padrões, respectivamente.

Contudo, a forma artificial da seleção clonal foi popularizada principalmente pelo algoritmo CSA (*clonal selection algorithm*) de de Castro e Von Zuben (1999) e de Castro e Von Zuben (2000), aplicado a problemas de reconhecimento de padrões e otimização multi-modal, mais tarde denominado CLONALG (DE CASTRO; VON ZUBEN, 2002). O método foi adaptado por Brownlee (2005) para classificação de dados, de forma a receber como entrada um conjunto de instâncias de treinamento e retornar um conjunto de células de memória utilizadas para classificar as instâncias de teste. As células de memória correspondem a protótipos, comumente utilizados em métodos de aprendizado supervisionado baseado em instância.

3.2.1 Inspiração Biológica

O CLONALG aproveita-se de algumas metáforas da seleção clonal do sistema imunológico em aplicações computacionais de reconhecimento de padrões e otimização multi-

modal, tais como (DE CASTRO; TIMMIS, 2003):

- Um antígeno seleciona vários linfócitos para proliferar, com taxa proporcional à afinidade entre eles;
- A taxa de mutação aplicada a um clone linfocitário é inversamente proporcional à afinidade entre ele e o antígeno;
- A proliferação das células imunes é assexuada (processo mitótico). As células dividem-se sem *crossover*;
- Durante a reprodução, os clones linfocitários passam por um processo de hipermutação que, junto com uma forte pressão seletiva, resulta em linfócitos com receptores antigênicos apresentando afinidades mais altas em relação ao antígeno apresentado.

3.2.2 Algoritmo

O Algoritmo 3.1 apresenta o princípio de funcionamento do CLONALG para reconhecimento de padrões, onde $|.|$ representa a cardinalidade de um conjunto, $mutOp(Pop)$ é a função responsável por aplicar o operador de mutação a uma população Pop e n_g é um parâmetro que define o número máximo de gerações.

Algoritmo 3.1: Princípio de funcionamento do CLONALG

Entrada: Ag: conjunto de dados de treinamento (antígenos)
Saída: P_m: conjunto de células de memória

1 **início**
  // $P = P_r + P_m$: População de anticorpos
  // P_m: População de células de memória
  // P_r: População restante
2 $P \leftarrow$ soluções candidatas aleatórias;
3 **para** $i \leftarrow 1$ **até** n_g **faça**
4 **para** *cada antígeno* ag *de* Ag **faça**
5 $P^* \leftarrow n$ indivíduos de P com maior afinidade com ag;
6 $C \leftarrow$ Clones de P^* (proporcional a afinidade com ag);
7 $C^* \leftarrow mutOp(C)$ (inversamente proporcional a afinidade com ag);
8 $ab_1 \leftarrow$ elemento de C^* com maior afinidade com ag ;
9 $ab_2 \leftarrow$ elemento de P_m com maior afinidade com ag ;
10 **se** $ab_1.afinidade > ab_2.afinidade$ **então**
11 $P_m \leftarrow P_m$ + ab_1 ;
12 Substitui s indivíduos de P_r com menor afinidade com ag por novos anticorpos;

O primeiro passo do algoritmo é gerar uma população inicial P de anticorpos de tamanho N, normalmente escolhida a partir do próprio conjunto de treinamento. Esta população é formada pelo repertório de anticorpos de memória (P_m) e pelo repertório de

anticorpos restantes (P_r). A cada iteração do algoritmo, denominada geração, cada antígeno ag do conjunto de instâncias de treinamento Ag é apresentado à população P. Os n anticorpos de P com maior afinidade em relação a ag são selecionados para a clonagem (conjunto P^*), resultando em uma população de clones C. A quantidade de clones gerados por anticorpo é definida pela equação 3.1, onde i é posição do anticorpo no conjunto P^*, ordenado descendentemente em relação à afinidade dos seus elementos com ag, $round(.)$ é uma função que arredonda um valor real dado para o valor inteiro mais próximo e β é um fator multiplicativo.

$$n_{clones} = round\left(\frac{\beta.N}{i}\right) \tag{3.1}$$

Cada anticorpo ab do repertório C passa então por um processo de mutação, resultando no conjunto C^*. Cada componente do vetor de atributos $ab.atributos$ varia segundo uma taxa de hipermutação α. Esta taxa é definida por de Castro e Von Zuben (2002) e de Castro e Timmis (2003) segundo as equações 3.2 e 3.3, respectivamente, onde ρ é um parâmetro que controla o decaimento da exponencial inversa, af é a afinidade normalizada entre o anticorpo em questão e ag e $af^* = \frac{af}{af_{max}}$ é a afinidade normalizada do anticorpo em relação a afinidade máxima encontrada no conjunto C.

$$\alpha = exp(-\rho.af) \tag{3.2}$$

$$\alpha = exp(-\rho.af^*) \tag{3.3}$$

Seleciona-se os anticorpos ab_1 e ab_2, das populações C^* e P_m, respectivamente, que possuem a maior afinidade em relação a ag. Se a afinidade de ab_1 for maior que a afinidade de ab_2, o algoritmo adiciona o anticorpo ab_1 ao repertório P_m. Além disso, os s anticorpos com menor afinidade da população P_r são substituídos por novos elementos escolhidos aleatoriamente a partir da população de antígenos ou gerados a partir de algum modelo.

A fase de treinamento completa-se ao fim das n_g gerações, retornando o conjunto final de células de memória, utilizado para classificar novos elementos. Um novo padrão p, de um conjunto de teste, é classificado pela classe da célula de memória mais próxima dele, em um procedimento semelhante ao algoritmo kNN com $k = 1$.

3.3 CSCA

Brownlee (2005) propôs um algoritmo baseado no CLONALG, mas com algumas melhorias, denominado CSCA (*clonal selection classifier algorithm*). O pseudocódigo do método pode ser visto no algoritmo 3.2.

3.3.1 Inicialização e Particionamento do Conjunto de Entrada

O classificador utiliza um parâmetro p para dividir o conjunto de dados de entrada Ag em lotes. O valor de p indica em quantas partições o repertório Ag será dividido. Esta medida tem por objetivo diminuir o processamento necessário para treinar conjuntos de

Algoritmo 3.2: Princípio de funcionamento do CSCA

Entrada: Ag: conjunto de dados de treinamento (antígenos)
Saída: Ab: conjunto de células de memória (anticorpos)

```
1  início
       // Inicialização da população
2      Ab ← Escolhe S antígenos aleatórios para formar a população inicial;
3      para i ← 1 até G faça
4          Ag* ← Seleciona uma partição de Ag;
5          Expõe Ab à partição Ag*;
6          Verifica a possibilidade de trocar a classe do anticorpo;
7          Calcula o valor de aptidão de cada membro de Ab;
8          Poda os anticorpos de Ab com aptidão menor que ε;
9          Ab' ← Anticorpos de Ab com pelo menos um erro de classificação;
10         se |Ab'| = 0 então
               // Força o fim do laço
11             i ← G + 1;
12         senão
13             Cl ← clonar(Ab');
14             Cl ← mutacao(Cl);
15             Ab ← Ab + Cl;
16             Insere |Ab'| anticorpos aleatórios em Ab;
17     Expõe Ab à população Ag;
18     Poda os anticorpos de Ab com aptidão menor que ε;
19     retorna Ab
```

entrada de tamanho muito grande. Se $p = 1$, os dados não são divididos e o algoritmo trabalha com uma única partição. Para valores maiores que 1, o classificador alterna entre as partições a cada geração do algoritmo.

Antes de iniciar as gerações, uma população inicial Ab de anticorpos é selecionada a partir de Ag. O tamanho de Ab é determinado pelo parâmetro S, definido pelo usuário.

3.3.2 Fase Iterativa

Nesta fase, alguns passos são repetidos por um determinado número de gerações. Em cada ciclo, a população de anticorpos é apresentada à partição corrente de antígenos (Ag^*) e verifica-se qual antígeno é reconhecido (classificado) por qual anticorpo. Os números de classificações corretas e incorretas são contados apenas para as unidades de maior correspondência (BMU - *best match unit*), isto é, o anticorpo com maior afinidade para um dado antígeno.

Ao fim desta contagem, aqueles anticorpos com nenhuma classificação correta e mais de uma classificação incorreta têm suas classificações (rótulos) trocados pela classe com maior contagem.

Com base nos valores das contagens, uma medida de aptidão (*fitness*) é calculada, definida pela equação 3.4, onde $nr_{mesmaClasse}$ é o número de itens reconhecidos da mesma

classe do anticorpo em questão e $nr_{outraClasse}$ é o número de antígenos reconhecidos de outras classes. Por convenção, quando $nr_{outraClasse} = 0$, $aptidao(ab) = nr_{mesmaClasse}$.

$$aptidao(ab) = \frac{nr_{mesmaClasse}}{nr_{outraClasse}} \tag{3.4}$$

3.3.2.1 Poda e Seleção

Os anticorpos com aptidão menor que o limiar de poda ε são removidos de Ab. Feito isso, os anticorpos remanescentes da população Ab com pelo menos uma classificação errada ($nr_{outraClasse} > 0$) são selecionados para formarem o conjunto Ab'.

Caso nenhum anticorpo seja selecionado para compor o repertório Ab', o laço termina. Caso contrário, ocorrem os processos de clonagem e mutação.

3.3.2.2 Clonagem

Durante a clonagem, os elementos de Ab' são copiados, compondo a população Cl. O número de clones gerados por cada anticorpo ab é definido pela equação 3.5, onde λ é um fator de escala opcional e $r(ab)$ é a aptidão relativa de ab, definido pela equação 3.6.

$$n_{clones}(ab) = r(ab) * |Ag^*| * \lambda \tag{3.5}$$

$$r(ab) = \frac{aptidao(ab)}{\sum_{j=1}^{|Ab|} aptidao(ab_i)} \tag{3.6}$$

3.3.2.3 Mutação

Após a clonagem, cada anticorpo ab da população Cl sofre um processo de mutação que muda os valores de seu vetor de atributos. Cada atributo at_i é modificado segundo a equação 3.7, onde $rnd(a, b)$ é uma função que gera um número real aleatório no intervalo $[a, b]$ e δ é a taxa de mutação calculada pela equação 3.8. A taxa de mutação é definida em relação à aptidão relativa do anticorpo e à variação possível do atributo em questão ($variacao_{at_i}$), isto é, a diferença entre seus valores máximo e mínimo possíveis.

$$at_i^{novo} = rnd(at_i^{antigo} - \delta, at_i^{antigo} + \delta) \tag{3.7}$$

$$\delta = \frac{variacao_{at_i}}{2} * r(ab) \tag{3.8}$$

3.3.3 Fase Final

Ao final da fase iterativa, o algoritmo expõe os anticorpos ao conjunto de antígenos completo (Ag), calcula os valores de aptidão e realiza a última poda. Os anticorpos restantes formam o conjunto de células de memória, utilizado durante a classificação.

Diferentemente do CLONALG, o CSCA utiliza um algoritmo kNN para inferir as classes das instâncias de teste, com o valor de k definido pelo usuário. Desta forma, pode-se atribuir um valor maior que 1 à variável k.

3.4 AIRS

O AIRS (*artificial immune recognition system*) é um dos algoritmos de SIA mais estudados e aplicados em classificação de dados (MCEWAN; HART, 2009). Além de seleção clonal, o AIRS possui inspiração de várias outras fontes dentro do campo de sistemas imunológicos artificiais, incluindo conceitos representacionais das células B, ARBs (*artificial recognition balls*) e limitação de recursos.

Uma ARB é uma estrutura de dados que representa múltiplos anticorpos idênticos, de forma que uma população de ARBs pode representar uma população muito maior de anticorpos de maneira eficiente.

Durante cada exposição a um dado antígeno, cada célula imune tenta adquirir recursos baseada no seu nível de estímulo. Entretanto há um número limitado de recursos no sistema. Se mais recursos são utilizados do que a quantidade existente, as células tem seus recursos removidos, começando pelas células menos estimuladas, até a quantidade de recursos atingir um valor adequado, que não ultrapasse o limite do sistema. A competição por recursos do sistema gera uma seleção, tendendo a manter apenas as células melhor adaptadas.

O AIRS é um algoritmo incremental *one-shot*, isto é, executa suas operações uma única vez para cada um dos antígenos, como pode ser visto no algoritmo 3.3 (WATKINS, 2001).

3.4.1 Inicialização

Inicialmente, os dados são normalizados de tal forma que as distâncias Euclideanas entre quaisquer antígenos esteja no intervalo [0,1] e o limiar de afinidade AT (*Affinity threshold*) é calculado pela equação 3.9, onde n é o número de antígenos do conjunto de treinamento e $dist(a, b)$ é a distância Euclidiana entre a e b. O limiar de afinidade é o valor de afinidade médio de todo o conjunto de treinamento.

$$AT = \frac{\sum_{i=1}^{n} \sum_{j=i+1}^{n} dist(ag_i, ag_j)}{\frac{n(n-1)}{2}} \tag{3.9}$$

Existem dois conjuntos principais utilizados ao longo do algoritmo, o conjunto Mc de células de memória e o conjunto Ab de ARBs. Ambos são inicializados com instâncias aleatórias do conjunto de treinamento. Após a inicialização, o treinamento continua de forma incremental, executando os passos uma única vez para cada antígeno ag.

3.4.2 Geração de ARBs

Após a inicialização, o primeiro passo é a geração de ARBs. O elemento mc_{melhor} de Mc com maior estímulo em relação a ag é escolhido e submetido ao operador de

Algoritmo 3.3: Versão simplificada do funcionamento do AIRS

Entrada: Ag: conjunto de dados de treinamento (antígenos)
Saída: Mc: conjunto de células de memória (anticorpos)

início
  Normaliza os dados (Ag);
  Calcula AT;
  $Mc \leftarrow$ Seleciona instâncias aleatórias de Ag;
  $Ab \leftarrow$ Seleciona instâncias aleatórias de Ag;
  para cada ag *em* Ag **faça**
    $mc_{melhor} \leftarrow$ Elemento de Mc com maior estímulo em relação a ag;
    $Mu \leftarrow clonar(mc_{melhor})$;
    $Mu \leftarrow mutacao(Mu)$;
    $Ab \leftarrow Ab + Mu$;
    $Parada \leftarrow F$;
    $PrimeiraExecucao \leftarrow V$;
    enquanto $Parada = F$ **faça**
      para cada ab *em* Ab **faça**
        $ab.estimulo \leftarrow estimulo(ab, ag)$;
        $ab.recursos \leftarrow$ Recursos alocados para ab;
      `// nc é o número de classes do problema`
      para $i \leftarrow 1$ **até** nc **faça**
        `// Em relação à classe` i
        enquanto *Recursos alocados* > *Recursos permitidos* **faça**
          Remove o elemento de Ab com menor estímulo;
      se $s_i \geq ST, \ \ \forall i \in \{1, 2, .., nc\}$ **então**
        $Parada \leftarrow V$;
      se $PrimeiraExecucao = V \vee Parada = F$ **então**
        $Mu \leftarrow clonar(Ab)$;
        $Mu \leftarrow mutacao(Mu)$;
        $Ab \leftarrow Ab + Mu$;
        $PrimeiraExecucao \leftarrow F$;
        se $s_i \geq ST, \ \ \forall i \in \{1, 2, .., nc\}$ **então**
          $Parada \leftarrow V$;
    Verifica a inserção de $mc_{candidata}$ em Mc;
  retorna *Mc*

clonagem, gerando uma população Mu de clones. Estes clones são então submetidos ao operador de mutação e, em seguida, adicionados à população Ab. A probabilidade de cada um dos elementos do vetor de atributos mudar é definida pelo parâmetro $mutationRate$.

O estímulo entre dois elementos e a quantidade de clones gerados são definidos pelas equações 3.10 e 3.11, onde $hyperClonalRate$ e $clonalRate$ são parâmetros definidos pelo usuário.

$$estimulo(x, y) = 1 - dist(x, y) \tag{3.10}$$

$$n_{clones}(mc) = hyperClonalRate * clonalRate * estimulo(mc, ag) \tag{3.11}$$

3.4.3 Competição por Recursos

A população Ab passa por um processo de competição por recursos. Para cada elemento $ab \in Ab$, são alocados recursos proporcionalmente ao seu estímulo em relação ao antígeno. Feito isso, caso a quantidade de recursos totais alocados seja superior ao limite do sistema, são removidos as ARBs menos estimuladas, até que o valor total de recursos seja menor ou igual ao valor limiar do sistema.

Depois de gerar e selecionar as ARBs, o algoritmo verifica se estas foram estimuladas suficientemente e podem seguir para o próximo antígeno, aplicando a equação 3.12 para cada uma das nc classes. O critério de parada só é atingido se e somente se $s_i \geq ST$ para todos os elementos do vetor $\vec{s} = s_1, s_2, ..., s_{nc}$, com ST representando o limiar de estimulação, definido pelo usuário, e $ab_j.estim$ denotando o estímulo de ab_j em relação a ag.

$$s_i \leftarrow \frac{\sum_{j=1}^{|Ab_i|} ab_j.estim}{|Ab_i|}, ab_j \in Ab_i \tag{3.12}$$

Independentemente se o critério de parada foi atingido ou não, o algoritmo procede passando por mais um processo de clonagem seguido de mutação, desta vez aplicados aos elementos de Ab. Ao final, as ARBs geradas são concatenadas com Ab.

Se o critério de parada não for atingido, o treinamento retorna ao passo de alocação de recursos, de onde continua a sua execução até que a desigualdade $s_i \geq ST$ seja verdadeira. Neste caso, se o critério de parada for atingido logo após a alocação de recursos, o procedimento de clonagem e mutação realizado na sequência não ocorre.

3.4.4 Introdução da Célula de Memória

Quando o critério de parada é atingido, a célula de memória $mc_{candidata} \in Ab$ com maior estímulo é selecionada como candidata a entrar para o conjunto Mc. Se o estímulo de $mc_{candidata}$ for maior que o estímulo de mc_{melhor}, $mc_{candidata}$ é adicionada ao conjunto Mc e se a afinidade (distância Euclidiana) entre mc_{melhor} e $mc_{candidata}$ for menor que $AT * ATS$, mc_{melhor} é removida do conjunto Mc. ATS é o limiar de afinidade escalar (*affinity threshold scalar*), definido pelo usuário.

O algoritmo continua a partir da fase de apresentação do antígeno até que não restem mais antígenos a serem aprendidos, retornando o conjunto Mc em seu término. Assim como no CSCA, os elementos de Mc são utilizados para definir as classes das instâncias de teste através do método kNN.

3.4.5 Outras Versões

A primeira versão do AIRS foi descrita por Watkins (2001), seguida por uma versão mais simples e mais eficiente, denominada AIRS2 (WATKINS; TIMMIS; BOGGESS, 2004) e uma versão distribuída (WATKINS; TIMMIS, 2004).

Outras versões do AIRS incluem o uso de seleção por torneio do mundo real (RWT-SAIRS - *real world tournament selection* AIRS) (GOLZARI et al., 2009) e o uso de atributos ponderados (WAIRS - *weighted* AIRS) (SECKER; FREITAS, 2007).

3.5 Outros Algoritmos

Ulutas e Kulturel-Konak (2011) realizaram uma revisão do estado da arte dos algoritmos baseados em seleção clonal e suas aplicações, incluindo problemas de engenharia industrial, *scheduling*, *design*, problema do caixeiro viajante, reconhecimento de padrões e otimização de funções.

Dentre os algoritmos apresentados, aqueles utilizados como classificadores incluem as versões paralela (WATKINS; BI; PHADKE, 2003) e sem parâmetros (CLONCLAS - *clonal classification*) (WHITE; GARRETT, 2003) do CLONALG, um método para reconhecimento de numerais escritos a mão (GARAIN; CHAKRABORTY; DASGUPTA, 2006) e um híbrido entre algoritmo de seleção clonal e SVM (*support vector machine*) (DING; LI, 2009).

4 APRENDIZADO BASEADO EM INSTÂNCIA

Este capítulo apresenta uma revisão de tópicos específicos a respeito de aprendizagem de máquina, destacando os sistemas de aprendizado baseado em instância. O objetivo é não apenas introduzir os conceitos relativos a este método de aprendizado específico, mas também apresentar maneiras de avaliar seu funcionamento, que serão utilizadas no Capítulo 6.

4.1 Aprendizagem de Máquina

Aprendizagem de máquina é a área da computação que investiga como computadores podem aprender baseando-se em dados. O conceito de aprendizado, neste caso, pode ser definido para incluir qualquer programa de computador que melhore seu desempenho em alguma tarefa através de experiência. Formalmente define-se da seguinte maneira (MITCHELL, 1997):

Definição. *Um computador é dito aprender da experiência E com respeito a uma classe de tarefas T e uma medida de performance P, se sua performance nas tarefas de T, medida por P, melhoram com a experiência E.*

Em um jogo de damas, ser humano contra computador, por exemplo, a tarefa T corresponde ao ato de o computador jogar damas contra um adversário, a medida de performance P corresponde ao resultado final do jogo (-1 se o oponente humano ganhar, 1 se o computador ganhar e 0 em caso de empate, por exemplo) e o conjunto de informações relativas à experiência corresponde às jogadas realizadas pelos dois jogadores durante uma partida.

Em aprendizagem de máquina, o conjunto de entrada, composto pelas experiências (E) e a saída (medida de desempenho P), é denominado conjunto de *treinamento*. Os métodos de aprendizagem de máquina podem ser classificados pelo uso de instrutor, formas de conhecimento e dados utilizadas e pela forma como o sistema aprende. Normalmente, os métodos de aprendizagem de máquina são classificados como (MITCHELL, 1997; MARSLAND, 2009; WOJTUSIAK, 2011):

- *Aprendizado supervisionado*: Um conjunto de entrada com as respectivas saídas é disponibilizado e, a partir destes dados, o algoritmo deve generalizar para responder corretamente a novas entradas.

- *Aprendizado não supervisionado*: O conjunto de dados provido não possui as saídas disponíveis e o algoritmo deve procurar por similaridade entre as instâncias, de maneira a categorizá-las da melhor forma.

- *Aprendizado por reforço*: Pode ser considerado uma forma intermediária entre o aprendizado supervisionado e o não supervisionado. O algoritmo utiliza uma função *recompensa* que determina a qualidade da saída obtida. Desta forma, o algoritmo deve explorar e tentar diferentes possibilidades, de forma a maximizar o valor da *recompensa*.

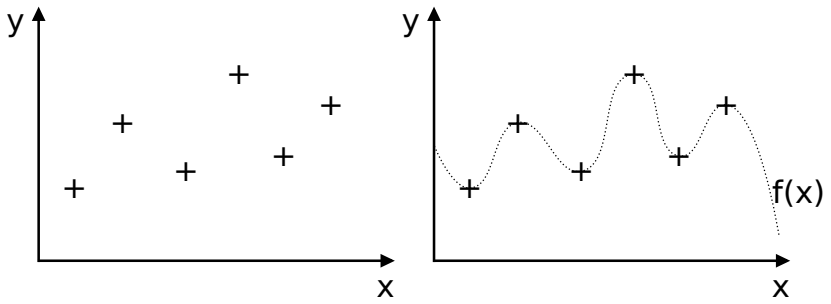
4.2 Aprendizado Supervisionado

Em aprendizado supervisionado, o conjunto de treinamento é composto por amostras de entrada-saída conhecidas. A entrada representa as características (atributos) das instâncias, normalmente representada na forma de vetores ($\vec{x}$), e a saída representa o valor mapeado para cada instância (t). O problema consiste em *induzir* um modelo que represente instâncias do espaço do problema desconhecidas para o sistema.

Em Filosofia, indução consiste em definir, a partir de uma amostra de exemplos de um conjunto C, uma conclusão geral sobre a totalidade de C, que não esteja rigorosamente relacionada a ele. Pode-se definir a tarefa de indução no aprendizado supervisionado como o processo de estimar uma dependência entrada-saída desconhecida, utilizando um conjunto limitado de observações ou medidas de entradas e saídas deste sistema (MITCHELL, 1997).

As tarefas mais comuns do aprendizado supervisionado compreendem *regressão* e *classificação*. Na regressão, as entradas estão relacionadas a saídas contínuas; na classificação, estão relacionadas a saídas discretas, denominadas rótulos ou classes.

A figura 4.2 apresenta um exemplo de regressão. O conjunto de treinamento é formado por seis instâncias representadas por coordenadas cartesianas. No caso, o vetor de características é composto por um único elemento (x) e o valor a ser mapeado é o valor de y. A função $f(x)$ é um exemplo de função que pode ser aprendida a partir das entradas, capaz de predizer o valor de y a partir do valor de x.

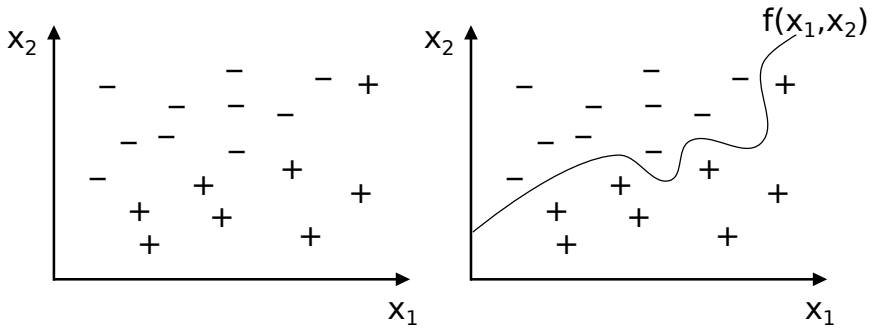


(a) Conjunto de treinamento (b) Função de regressão

Figura 4.1: Interpretação gráfica de regressão.

A figura 4.2 exibe um exemplo de classificação, em um espaço de características bidimensional ($\vec{x} = \{x_1, x_2\}$), composto por duas classes ("+" e "−"). Neste exemplo,

o classificador gera a função (modelo), $f : \mathbb{R} \times \mathbb{R} \rightarrow \{+, -\}$, a partir do conjunto de treinamento. Esta função descreve uma curva que particiona o espaço em dois, de acordo com a classe mapeada pela função.



(a) Conjunto de treinamento (b) Função classificadora

Figura 4.2: Interpretação gráfica de classificação.

Diversos métodos são utilizados para realizar tarefas de aprendizado supervisionado. Um deles é o aprendizado baseado em instância, que utiliza as instâncias de entrada (ou novas instâncias geradas pelo classificador) para atribuir o valor ou a classe referente às novas instâncias apresentadas. Os exemplos de treinamento são armazenados e uma função de distância é utilizada para determinar qual membro do conjunto de treinamento está mais próximo a uma instância de teste desconhecida. Uma vez que a instância de treinamento mais próxima é localizada, seu valor de saída determina o valor de saída da instância de teste (MITCHELL, 1997).

Métodos convencionais de aprendizado baseado em instância armazenam todos os dados de treinamento e comparam os dados de teste com estes. A cada nova instância de teste apresentada, todas as instâncias de treinamento armazenadas são checadas a fim de encontrar os vizinhos mais próximos, o que acaba exigindo muito tempo de processamento. Não há uma fase de treinamento explícita, adiando todo o processamento para a fase de teste. Por isso, normalmente são denominados *lazy learners* (aprendizes preguiçosos).

4.2.1 k-Vizinhos Mais Próximos

Os algoritmos baseados em instância para classificação de dados são derivados do classificador de padrões por k-vizinhos mais próximos (kNN - *k-nearest neighbor*) (COVER; HART, 1967), um método que atribui a uma nova instância a classe majoritária entre os k pontos mais próximos desta.

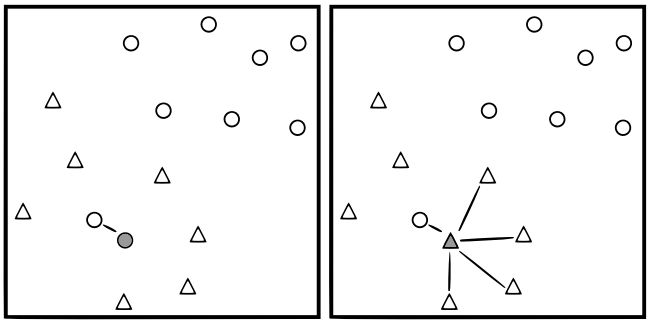
O método kNN assume que todas as instâncias correspondem a pontos em um espaço n-dimensional $\mathbb{R}^n$ e utiliza funções para calcular a distância entre estes pontos. As distâncias mais utilizadas são a distância Euclidiana, distância de Manhattan e distância de Hamming, apresentadas nas equações 4.1, 4.2 e 4.3, respectivamente, onde x e y são duas tuplas de dados, representadas por pontos no espaço $\mathbb{R}^n$. As distâncias Euclidiana e de Manhattan normalmente são utilizadas para problemas com valores reais e a distância de Hamming normalmente é utilizada para valores binários ou categóricos.

$$dist_e(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{4.1}$$

$$dist_m(x,y) = \sum_{i=1}^{n}|x_i - y_i| \tag{4.2}$$

$$dist_h(x,y) = \sum_{i=1}^{n}\delta, \text{ onde } \delta = \begin{cases} 1 & \text{se } x_i \neq y_i \\ 0 & \text{caso contrário} \end{cases} \tag{4.3}$$

O valor de k define o número de vizinhos utilizados para determinar a classe de uma instância e pode influenciar os resultados da classificação. Quando $k > 1$, utiliza-se a classe majoritária. A figura 4.2.1 apresenta classificações diferentes para um mesmo exemplo, dependendo do valor de k. Existem duas classes, os triângulos e os círculos e uma nova instância (em destaque) deve ser classificada em uma destas classes. Para $k = 1$, a nova instância é classificada como círculo e para $k = 5$, é classificada como triângulo.



(a) $k = 1$ (b) $k = 5$

Figura 4.3: kNN e classificação de uma instância (em destaque) para $k = 1$ e $k = 5$.

4.2.2 1NN e o Diagrama de Voronoi

A superfície de decisão de um classificador 1NN (kNN com $k = 1$) é uma combinação de poliedros convexos circundando cada um dos exemplos de treinamento. Os limites destes poliedros determinam a fronteira de decisão de cada instância de treinamento. Dentro desta fronteira, qualquer ponto tem como vizinho mais próximo a instância referente e, portanto, é classificada por esta.

A combinação destes poliedros é denominada de *diagrama de Voronoi* e é exemplificada na figura 4.2.2, para um conjunto de instâncias qualquer.

4.2.3 Seleção e Construção de Protótipos

No método kNN, as instâncias utilizadas para treinar o classificador são armazenadas indiscriminadamente. Nenhum processo de seleção é realizado e, como resultado,

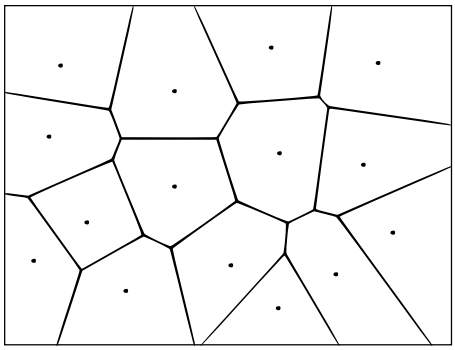


Figura 4.4: Exemplo de um diagrama de Voronoi

instâncias prejudiciais e supérfluas são armazenadas desnecessariamente (BRIGHTON; MELLISH, 2002).

Alguns métodos baseados no kNN tentam utilizar a menor quantidade de instâncias possível, que sejam aptas a predizer a classe de uma instância de teste com a mesma ou maior acurácia que o conjunto original de treinamento. Para tanto, utilizam-se de métodos de seleção e construção de protótipos (REINARTZ, 2002).

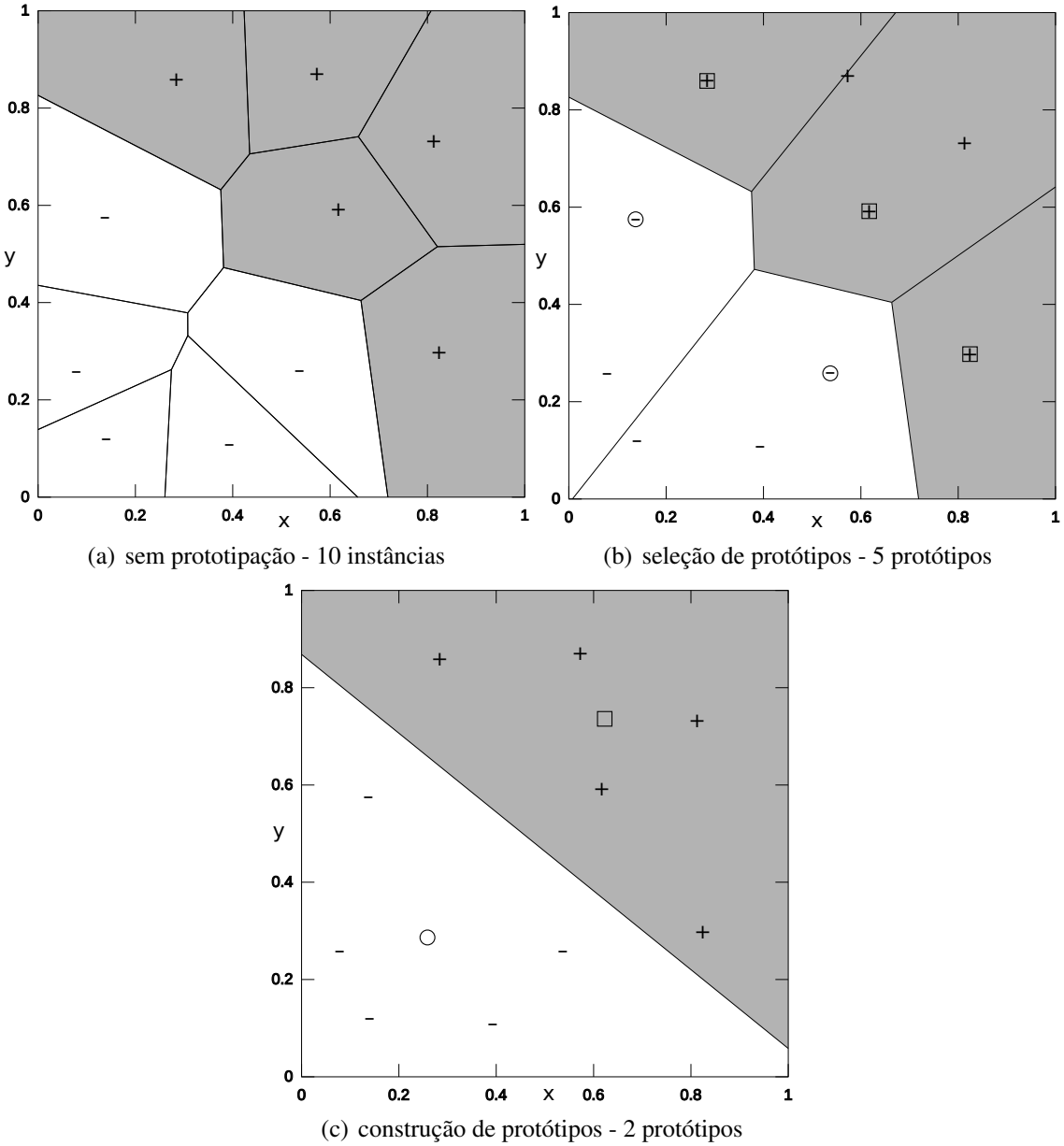
Protótipos são conjuntos de instâncias capazes de representar os dados de treinamento de forma mais condensada. Desta forma, pode-se reduzir o número de instâncias armazenadas pelo algoritmo. Os métodos de seleção e construção de protótipos são descritos a seguir:

- *Seleção de protótipos*: seleciona o conjunto de instâncias mais representativas do conjunto de treinamento, evitando aquelas que sejam supérfluas ou prejudiciais. O algoritmo IB2 (AHA; KIBLER; ALBERT, 1991) é um exemplo.
- *Construção de protótipos*: protótipos são gerados a partir do conjunto de treinamento sem que necessariamente correspondam a tuplas deste conjunto. Este tipo de prototipagem usa uma função específica para explicitamente construir novas instâncias que representem informações de um subconjunto inteiro de tuplas. Muitos classificadores baseados em SIA utilizam este método, tais como o CLONALG (DE CASTRO; VON ZUBEN, 2000), AIRS (WATKINS, 2001) e WAIRS (SECKER; FREITAS, 2007).

A figura 4.2.3 apresenta três diagramas de Voronoi representando as instâncias utilizados pelo classificador kNN, definidas a partir de um algoritmo sem prototipagem, com seleção de protótipos e com geração de protótipos, respectivamente. Os quadrados e os círculos representam os protótipos das classes 1 e 2, respectivamente, e os sinais $+$ e $-$ representam as instâncias do conjunto de treinamento das classes 1 e 2, respectivamente.

4.3 Avaliação de Modelos

Além dos classificadores citados nesta dissertação, existem muitos outros trabalhos da literatura que sugerem métodos diferentes de classificar instâncias de dados, cada qual com suas particularidades. Desta forma, é importante definir uma métrica de avaliação, a fim de definir qual o melhor classificador para o problema proposto.



(a) sem prototipação - 10 instâncias

(b) seleção de protótipos - 5 protótipos

(c) construção de protótipos - 2 protótipos

Figura 4.5: Diagramas de Voronoi de instâncias utilizadas por um classificador 1NN, definidas (a) por um algoritmo sem prototipagem, (b) por seleção de protótipos e (c) por construção de protótipos.

Para avaliar um classificador, deve-se treiná-lo e testá-lo com conjuntos de dados diferentes, provenientes de um mesmo domínio de problema e, a partir dos resultados obtidos, calcular uma métrica de comparação. Contudo, normalmente dispõe-se apenas de uma quantidade limitada de dados a respeito do problema e, mesmo que os dados sejam distribuídos segundo uma distribuição de probabilidade, esta não é conhecida a priori. Desta forma, deve-se definir uma forma eficaz de estimar os valores utilizados pela métrica.

4.3.1 Validação Cruzada

Uma medida comumente utilizada para avaliar a qualidade de um modelo é a acurácia preditiva, que pode ser estimada pela taxa de erro verdadeiro. Estatisticamente, a taxa de erro verdadeiro é definida como a taxa de erro do modelo, em um número de novos casos assintoticamente grande que converge para a distribuição da população real (MITCHELL, 1997).

A taxa de erro verdadeiro deve ser estimada a partir de um conjunto grande de amostras, que seja independente do conjunto de instâncias utilizadas para treinamento, ou seja, deve-se dividir o conjunto de entrada, que normalmente já é pequeno, devido a dificuldade em se obter os dados, em duas partições. Se a quantidade de amostras destinadas ao teste for grande, os dados de treinamento podem ser insuficientes para gerar um modelo com boa capacidade preditiva. Caso contrário, se poucas instâncias forem utilizadas durante o teste, a estimativa do erro verdadeiro pode não ser confiável.

Para contornar este problema, diversas técnicas de *amostragem* são sugeridas pela literatura. Uma das mais utilizadas é a *validação cruzada*. Considerando um conjunto D de entrada, de tamanho n, a validação cruzada com k partições consiste em dividir as amostras disponíveis em k subconjuntos disjuntos ($D_1, D_2, ..., D_k$), onde $1 \leq k \leq n$. O classificador é treinado e testado k vezes; a cada passo $t \in \{1, 2, ..., k\}$, o conjunto $D - D_t$ é utilizado para treinamento e o conjunto D_t é utilizado como teste. Segundo Kohavi (1995), a validação cruzada com dez pastas ($k = 10$) é uma das melhores opções para comparação de modelos.

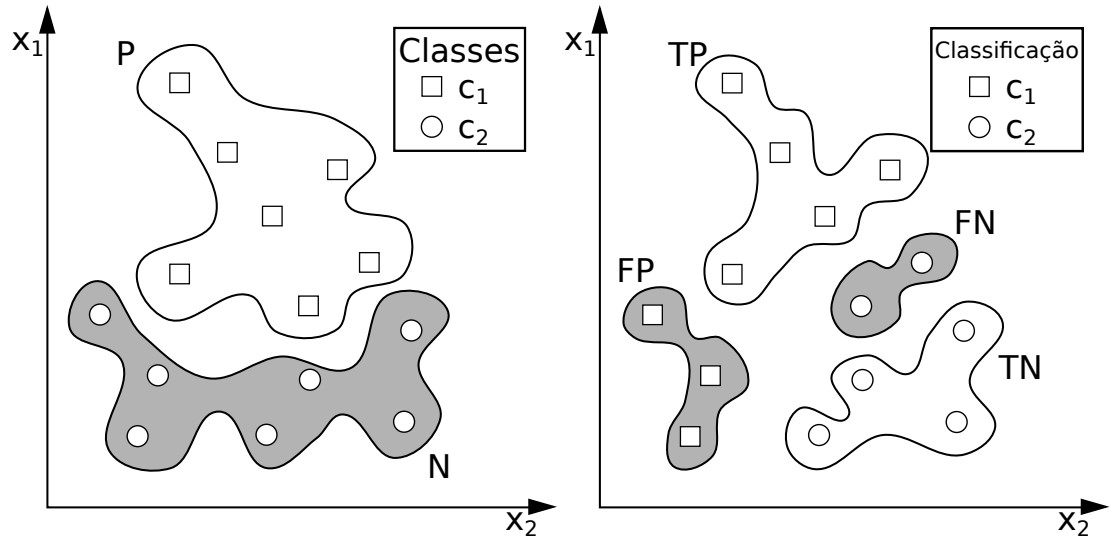
4.3.2 Métricas Avaliativas

Considerando um conjunto de dados de entrada para classificação, de tamanho n, dividido em nc classes diferentes, com $nc > 1$, dada uma classe $i \in \{1, 2, .., nc\}$, denomina-se os elementos pertencentes a essa classe de instâncias *positivas* (P) e os elementos das outras $nc - 1$ classes de instâncias *negativas* (N). Após a classificação, denomina-se as instâncias classificadas, em relação a uma classe i, segundo a forma como foram classificadas pelo modelo. As amostras positivas correta e incorretamente classificadas são denominadas *verdadeiros positivos* (TP - true positive) e *falsos negativos* (FN - false negative), respectivamente, e as amostras negativas correta e incorretamente classificadas são denominadas *verdadeiros negativos* (TN - true negative) e *falsos positivos* (FP - false negative), respectivamente. A figura 4.3.2 apresenta um exemplo de classificação e as respectivas nomenclaturas, em relação à classe c_1.

Dadas estas definições, algumas medidas de avaliação utilizadas na literatura são apresentadas na tabela 4.1 (HAN; KAMBER; PEI, 2011). Os valores TP, TN, FP, FN, P e N representam a quantidade de verdadeiros positivos, verdadeiros negativos, falsos positivos, falsos negativos, amostras positivas e amostras negativas, respectivamente.

4.3.3 Comparação de Modelos

Ao comparar dois modelos gerados por classificadores, o erro estimado deve ser levado em conta. Desta forma, nenhuma conclusão pode ser levantada sobre qual modelo é melhor que o outro, com significância estatística.



(a) Exemplo de um conjunto de dados e suas respectivas classes.

(b) Classificação segundo um modelo qualquer e as respectivas nomenclaturas.

Figura 4.6: Nomenclatura das instâncias, em relação à classe c_1.

Tabela 4.1: Métricas avaliativas utilizadas na literatura

Medida	**Fórmula**
Acurácia, taxa de acerto	$acuracia = \frac{TP+FP}{P+N}$
Taxa de erro	$erro = \frac{FP+FN}{P+N}$
Sensibilidade, *recall*, taxa de verdadeiros positivos	$recall = \frac{TP}{P} = \frac{TP}{TP+FN}$
Especificidade, taxa de verdadeiros negativos	$especificidade = \frac{TN}{N} = \frac{TN}{TN+FP}$
Precisão	$precisao = \frac{TP}{TP+FP}$
F, F_1, $F - score$	$F_1 = \frac{2*precisao*recall}{precisao+recall}$
F_β, onde β é um valor real não-negativo	$F_\beta = \frac{(1+\beta^2)*precisao*recall}{\beta^2*precisao+recall}$

O *teste-t corrigido* (NADEAU; BENGIO, 2003) é utilizado para determinar se a média de uma medida de desempenho de um algoritmo é significativamente maior que, ou menor que, a média de outro algoritmo. É uma modificação do *teste-t* padrão que funciona bem na prática (WITTEN; FRANK, 2011). Esta modificação do *teste-t* utiliza um valor t diferente calculado pela equação 4.4, onde n_1 é o número de instâncias utilizadas para treinamento, n_2 é a quantidade utilizada para teste, k é o número de execuções, σ_d^2 é a estimativa da variância das k diferenças e $\bar{d}$ é definido pela equação 4.5, onde a_j e b_j são os valores a serem estimados para os algoritmos A e B, respectivamente, medidos na execução j ($1 \leq j \leq k$) (BOUCKAERT; FRANK, 2004).

$$t = \frac{\bar{d}}{\sqrt{\left(\frac{1}{k} + \frac{n_2}{n_1}\right)\sigma_d^2}} \tag{4.4}$$

$$\bar{d} = \frac{1}{k} \sum_{j=1}^{k} a_j - b_j \tag{4.5}$$

Se as médias são as mesmas, a diferença $\bar{d}$ é zero (denominada *hipótese nula*); se elas são significativamente diferentes, a diferença será significativamente diferente de zero. Assim, para um determinado nível de confiança (α), é verificado se a diferença real excede o limite de confiança (z). O valor de z é definido a partir da distribuição *t* de Student, para $k - 1$ graus de liberdade com confiança α bicaudal.

Caso o valor de t seja maior que z, ou menor que $-z$, rejeita-se a *hipótese nula* de que as médias são iguais e conclui-se que há uma diferença significativa entre os valores medidos.

5 UM NOVO CLASSIFICADOR BASEADO EM SELEÇÃO CLONAL

Este capítulo descreve o funcionamento do classificador desenvolvido, denominado *clonal selection classifier with data reduction* (CSCDR), apresentando as principais rotinas do algoritmo e os princípios relacionados. O algoritmo utiliza uma função de aptidão baseada nos números de instâncias correta e incorretamente classificadas por cada anticorpo e tenta maximizar este valor através da seleção clonal.

5.1 A Base de Funcionamento do CSCDR

O algoritmo desenvolvido neste trabalho é baseado em outros trabalhos da literatura de AIS e em princípios e teorias da imunologia. É importante salientar as contribuições para o desenvolvimento do CSCDR e as melhorias realizadas em relação aos outros algoritmos utilizados como inspiração.

5.1.1 Metáfora Biológica

No sistema imunológico natural, os linfócitos B são estimulados a se reproduzirem e passarem por hipermutações somáticas como resposta à exposição a um antígeno desconhecido, durante o processo de seleção clonal. Por decorrência deste processo, a afinidade entre os receptores linfocitários (anticorpos, no caso das células B) e o antígeno tende a aumentar (maturação de afinidade). Alguns destes linfócitos diferenciam-se em células de memória, responsáveis por garantir uma resposta mais rápida e efetiva a posteriores exposições antigênicas.

No caso do CSCDR, os conceitos de linfócito B e anticorpo (receptor de superfície) se misturam. Ambos são tratados como uma única entidade, representada pela estrutura de dados *anticorpo*. Desta forma, os anticorpos passam pelas fases de exposição antigênica, seleção (poda), clonagem e hipermutação, de maneira semelhante ao que ocorre na contraparte biológica, a fim de aumentar a afinidade entre a população de anticorpos e os antígenos de entrada.

O objetivo do classificador é gerar centroides que aprendam a partir do conjunto de treinamento, a fim de classificarem novas instâncias de dados. Utilizando a metáfora imunológica, pode-se dizer que o CSCDR deve gerar um conjunto de células de memória que aprendam a partir da interação entre os anticorpos e os antígenos (instâncias de

treinamento) a fim de responderem com alta afinidade a encontros posteriores com novos antígenos (instâncias de teste).

5.1.2 Inspiração e Melhorias

O CSCDR é fortemente baseado no algoritmo CSCA. Este, por sua vez, é fortemente baseado no CLONALG e possui algumas ideias retiradas do AIRS.

O CSCDR apresenta várias mudanças quando comparado ao CSCA. A principal delas é a substituição do limiar de afinidade ε pelo parâmetro N como controle do tamanho do repertório principal de anticorpos. Além de possibilitar que o usuário defina o tamanho da população final de células de memória, esta substituição produz um mecanismo de poda mais robusto.

Oliveira, Mota e Barone (2012) realizaram um estudo preliminar dos efeitos do parâmetro ε sobre os resultados do CSCA. Se o valor escolhido for baixo, a quantidade de anticorpos selecionados para a poda diminui e, consequentemente, a população final de células de memória cresce. Se o valor for grande, o processo de seleção torna-se muito exigente, ocorrendo podas em excesso. Isto origina repertórios com poucas ou nenhuma células de memória, incapazes de representar de forma eficaz os dados de entrada. Desta forma, o valor de ε tem que ser escolhido com cuidado a fim de balancear o tamanho do repertório e a qualidade dos resultados.

O parâmetro N, por outro lado, é mais versátil. Além de definir o tamanho da população inicial e final, poupando um parâmetro de entrada, possibilita um mecanismo de poda mais controlado. O usuário ainda tem que gerir a relação entre o tamanho do repertório e a qualidade dos resultados; porém, pode definir exatamente o tamanho da população desejada, ao contrário do parâmetro ε, que se mostrou bastante instável como forma de controle populacional.

Outra vantagem sobre o CSCA é que o CSCDR trabalha com uma população de tamanho fixo, definido por N, e tenta maximizar a representatividade das células de memória, isto é, aumentar o número de instâncias de treinamento representadas por cada centroide, através de um algoritmo de seleção clonal. Desta forma, teoricamente, uma quantidade menor de células de memória é necessária para representar os dados.

O método de mutação utilizado pelo CSCDR é baseado no CLONALG desenvolvido por de Castro e Von Zuben (2002), onde a taxa de mutação é inversamente proporcional à medida de aptidão normalizada do anticorpo. A ideia por trás desta abordagem consiste em que candidatos próximos a um ótimo local (aptidão alta) devem ser ajustados minuciosamente, enquanto candidatos longe de ótimos locais (aptidão baixa) podem realizar saltos grandes em busca de regiões do espaço com valores de aptidão mais altos. Já no CSCDR, a taxa de mutação é proporcional à aptidão do anticorpo.

5.2 Parâmetros de Entrada

O algoritmo CSCDR possui ao todo quatro parâmetros de entrada definidos pelo usuário, apresentados abaixo:

- N: Este parâmetro é um dos mais importantes. Ele define a quantidade de anti-

corpos gerados no início do algoritmo e o tamanho do repertório final de células de memória. Além disso, é utilizado durante as podas para controlar o tamanho da população de anticorpos.

- n_g: Define o número máximo de gerações executadas pelo CSCDR.

- d: Determina a quantidade de anticorpos aleatórios adicionados à cada geração. Esta inserção permite que o algoritmo explore possíveis soluções (células de memória) em outras regiões do espaço de formas.

- β: Constante multiplicativa utilizada para definir a quantidade de clones gerados.

Comparando-se o CSCDR com os SIAs classificadores mais conhecidos da literatura, verifica-se que o mesmo possui o menor número de parâmetros de entrada. A tabela 5.1 apresenta um comparativo entre esses valores. Normalmente, os parâmetros de entrada estão diretamente relacionados com o desempenho do algoritmo e, para melhorar os resultados é necessário ajustar os valores de entrada de acordo com cada problema. Quanto maior a quantidade de parâmetros, mais difícil torna-se esta tarefa de ajuste.

Tabela 5.1: Número de parâmetros de entrada de SIAs classificadores.

Algoritmo	**Nro. de parâmetros**	**Artigo original**
AIRS (versão 1)	8	(WATKINS, 2001)
AIRS (versão 2)	8	(WATKINS; TIMMIS; BOGGESS, 2004)
CLONALG	6	(DE CASTRO; VON ZUBEN, 2002)
CSCA	6	(BROWNLEE, 2005)
CSCDR	4	-

5.3 O Algoritmo

O CSCDR é baseado principalmente nos algoritmos CLONALG e CSCA, com algumas modificações para aumentar o desempenho e diminuir o número de células geradas pelo processo. O algoritmo 5.1 apresenta o seu funcionamento.

5.3.1 Inicialização

O primeiro passo do algoritmo é gerar uma população inicial de anticorpos (Ab). A função $GeraAnticorpos(N)$ seleciona N antígenos de Ag, de forma aleatória e sem reposição, e copia os conteúdos dos vetores de atributos e dos rótulos destes antígenos para um conjunto de novos anticorpos. O parâmetro N é responsável por controlar o tamanho da população de anticorpos durante a execução do algoritmo.

Após a inicialização da população Ab, os anticorpos passam por ciclos de exposição antigênica e seleção clonal. Cada ciclo é denominado *geração* e o número de gerações é determinado por um parâmetro n_g, definido pelo usuário.

Algoritmo 5.1: Princípio de funcionamento do CSCDR

Entrada: Ag: conjunto de dados de treinamento (antígenos)
Saída: Ab: conjunto de células de memória (anticorpos)

```
1  início
       // Inicialização da população
2      Ab ← GeraAnticorpos(N) ;
3      para i ← 1 até n_g faça
4          Expõe Ab à população Ag;
5          Calcula o valor de aptidão de cada membro de Ab;
6          Poda os piores anticorpos de Ab;
7          Ab ← Ab + GeraAnticorpos(d);
8          Cl ← Clonar(Ab);
9          Cl ← Mutacao(Cl);
10         Ab ← Ab + Cl;
11     Poda Ab uma última vez;
12     retorna Ab
```

5.3.2 Valor de Aptidão

Durante a exposição, cada antígeno é apresentado à população de anticorpos. A distância entre o antígeno e cada membro de Ab é calculada e o anticorpo mais próximo (maior afinidade) é selecionado como *unidade com maior correspondência* (UMC). A distância é definida de acordo com a forma de representação adotada, conforme discutido na Seção 3.1.

Na imunologia, a medida de afinidade é uma medida proporcional à complementaridade entre dois elementos. Para que os pares paratopo/epítopo ou paratopo/idiotopo possam interagir entre si deve existir uma quantidade mínima de regiões complementares em sua forma generalizada (PERELSON, 1989). Já no CSCDR, a medida de afinidade é proporcional à similaridade entre o anticorpo ab e o antígeno ag; isto é, quanto mais próximos se encontrarem no espaço de formas maior é a afinidade entre eles. A maior afinidade possível entre ab e ag ocorre quando ambos encontram-se na mesma posição no espaço, ou seja, $distancia(ab, ag) = 0$ e a menor afinidade ocorre quando $distancia(ab, ag) \to \infty$. A figura 5.3.2 mostra a relação entre distância e afinidade entre dois elementos do algoritmo.

Apesar de espaços binários ou discretos poderem ser eventualmente explorados pelo algoritmo, a implementação corrente do CSCDR é mais adequada para espaços contínuos, definidos em $\mathbb{R}^d$. Desta forma, utiliza-se a distância Euclidiana como medida de proximidade entre antígenos e anticorpos.

Cada anticorpo possui um contador para cada classe possível do problema e quando é selecionado como UMC, seu contador para a classe do antígeno em questão é incrementado em um. Após a fase de exposição, o algoritmo verifica se cada anticorpo deve alterar seu rótulo. Se houver contadores de classe com valores maiores do que a contagem da própria classe, o anticorpo tem seu rótulo alterado para a classe com contador com maior valor. Esta providência visa maximizar os valores de aptidão dos anticorpos.

A equação 5.1 apresenta a função utilizada para calcular a aptidão de cada anti-

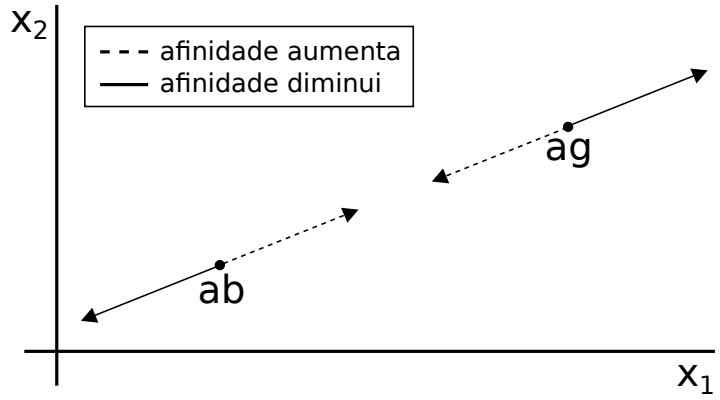


Figura 5.1: Relação entre distância e afinidade em um espaço com dois atributos (x_1 e x_2). Se a distância entre ab e ag aumenta, a afinidade diminui e vice-versa.

corpo. A função $TP(ab)$ (equação 5.2) e $FP(ab)$ (equação 5.3) retornam o número de itens corretamente e incorretamente classificados pelo anticorpo ab, respectivamente. Isto assemelha-se aos valores de verdadeiro positivo (*true positive*) e falso positivo (*false positive*), apresentados na Seção 4.3, relativos a cada anticorpo. O vetor $ab.contador$ é o vetor de contadores utilizado para armazenar as contagens de UMC, nc é o número de classes do problema e $ab.rotulo$ é o rótulo da classe à qual pertence ab, tal que $ab.rotulo \in \{1, 2, ..., nc\}$. Em um caso particular, quando $TP(ab) = 0$, $aptidao(ab) = 0$.

$$aptidao(ab) = \frac{TP(ab) + 1}{FP(ab) + 1} \tag{5.1}$$

$$TP(ab) = ab.contador[ab.rotulo] \tag{5.2}$$

$$FP(ab) = \sum_{i=1}^{nc} ab.contador[i],\ i \neq ab.rotulo \tag{5.3}$$

5.3.3 Poda

Depois de calcular os valores de aptidão, os anticorpos com os valores mais baixos são removidos da população Ab. No total, $|Ab| - N$ anticorpos são removidos, igualando o tamanho de Ab ao seu tamanho inicial N. Na primeira geração, $|Ab| = N$ e, portanto, a poda não ocorre. Contudo, nas gerações seguintes, a poda ocorre para controlar o crescimento populacional em decorrência do processo de clonagem.

5.3.4 Adição de Novos Anticorpos

Antes de iniciar o processo de clonagem e mutação, d novos anticorpos são adicionados à população Ab, gerados a partir da função $GeraAnticorpos(d)$, definida na Seção 5.3.1. Os novos anticorpos permitem que outros pontos do espaço de busca possam ser explorados e, portanto, a função $GeraAnticorpos$ deve ser definida de forma que os antígenos selecionados para gerar os anticorpos não se repitam.

Uma forma de garantir isto é utilizando uma lista de ponteiros $Ag_{ponteiros}$ apontando para os elementos de Ag. Os antígenos que originam os novos anticorpos são selecionados a partir de $Ag_{ponteiros}$ e removidos da lista. Quando $|Ag_{ponteiros}| = 0$, reinicia-se a lista com novos ponteiros ligados a Ag.

5.3.5 Clonagem e Mutação

Neste passo, os anticorpos de Ab são submetidos à clonagem e mutação, resultando na população Cl de clones. O número de clones gerados para cada anticorpo ab é definido pela equação 5.4, onde β é uma constante multiplicativa, definida pelo usuário, $round(.)$ é uma função que arredonda um valor real dado para o valor inteiro mais próximo, $|.|$ é a cardinalidade de um conjunto e i_{ab} é a posição do anticorpo ab na população Ab ordenada do maior para o menor valor de aptidão.

$$n_{cl}(ab) = round\left(\frac{|Ab| * \beta}{i_{ab}}\right) \tag{5.4}$$

Cada anticorpo ab de Cl é submetido ao processo de hipermutação, onde os componentes de seu vetor de características mudam segundo a equação 5.5. O novo valor de um atributo at_j é calculado a partir do seu antigo valor e de uma taxa de variação α, definida pela equação 5.6. A função $rnd(a, b)$ gera um número real aleatório no intervalo $[a, b]$, D é a aptidão relativa do anticorpo ab, calculado a partir da equação 5.7 e var_{at_j} é o intervalo de variação possível da característica, isto é, o maior menos o menor valor possível para o atributo at_j.

$$at_j^{novo} = rnd(at_j^{velho} - \alpha, at_j^{velho} + \alpha) \tag{5.5}$$

$$\alpha = exp(-5 * D) * \frac{var_{at_j}}{2} \tag{5.6}$$

$$D = \frac{aptidao(ab)}{\arg\max\limits_{ab \in Ab} aptidao(ab)} \tag{5.7}$$

5.3.6 Última Poda

Após o processo de mutação, a população Cl resultante é adicionada ao repertório Ab. Com este último passo, o algoritmo conclui uma geração. Se o número máximo de gerações ainda não tiver sido atingido, o CSCDR retorna ao início do laço (linha 4 do algoritmo 5.1). Caso contrário, o algoritmo executa uma última poda.

A última poda é semelhante às podas que ocorrem durante uma geração. A população Ab, gerada ao final da última geração, é exposta aos antígenos do repertório Ag e os valores de aptidão dos anticorpos são calculados. Os $|Ab| - N$ anticorpos com valor de aptidão mais baixo são removidos de Ab. Com isso, o algoritmo encerra sua execução e retorna a população de células de memória Ab.

5.3.7 Classificação

Durante a fase de teste, as células de memória são utilizadas para classificar novas instâncias de dados. Dado um padrão p de entrada, a sua classe é definida pela classe do anticorpo mais próximo de p, isto é, $p.rotulo = ab.rotulo \,|\, \arg\min_{ab \in Ab} distancia(p, ab)$.

6 ESTUDO EXPERIMENTAL

Este capítulo apresenta uma investigação a respeito do comportamento do algoritmo CSCDR em bases reais e sintéticas. Duas bases sintéticas são utilizadas para apresentar a distribuição espacial dos detectores gerados pelo algoritmo, e seu desempenho em bases de dados reais, utilizadas pela literatura como *benchmarks*, é medido e comparado ao desempenho de outros classificadores da literatura.

Os algoritmos deste capítulo, inclusive o CSCDR, são desenvolvidos em Java como módulo para o pacote de software WEKA (*Waikato environment for knowledge analysis*) (WITTEN; FRANK, 2011).

6.1 Comportamento do Algoritmo

Dentre as questões a serem investigadas a respeito do funcionamento do algoritmo CSCDR, destacam-se nesta seção: como as células de memória distribuem-se pelo espaço, os efeitos do tamanho do repertório final (N) e os diagramas de Voronoi formados pelos centroides.

6.1.1 Conjuntos de Dados

Para demonstrar e investigar o comportamento do CSCDR no espaço cartesiano, duas bases de dados sintéticas simples (A e B) são utilizadas, representando problemas bidimensionais com apenas duas classes.

Para cada base de dados, 500 pontos bidimensionais são gerados aleatoriamente, segundo uma distribuição uniforme. As componentes p_x e p_y de um ponto qualquer p são geradas de forma que ambos valores pertençam ao intervalo $[0, 10]$. Estes valores compõem seu vetor de características e representam a sua posição em um plano cartesiano 10×10.

As classes das instâncias da base de dados A são definidas segundo a equação 6.1. O espaço é dividido em dois pela equação 6.2 e cada porção correspondendo a uma classe.

$$classe_A(p) = \begin{cases} 0 & \text{se } p_y < sen(\pi * p_x) + p_x \\ 1 & \text{caso contrário} \end{cases} \tag{6.1}$$

$$p_y = sen(\pi * p_x) + p_x \tag{6.2}$$

No caso da base B, a classificação das instâncias é dada pela equação 6.3. Os pontos da classe 0 são separados da classe 1 por cinco regiões retangulares.

$$classe_B(p) = \begin{cases} 0 & \text{se } 0 \leq p_x \leq 2 \text{ e } 0 \leq p_y \leq 5 \\ 0 & \text{se } 1 \leq p_x \leq 3 \text{ e } 6,5 \leq p_y \leq 10 \\ 0 & \text{se } 3,5 \leq p_x \leq 5,5 \text{ e } 2,5 \leq p_y \leq 5,5 \\ 0 & \text{se } 7,5 \leq p_x \leq 10 \text{ e } 0 \leq p_y \leq 3 \\ 0 & \text{se } 5 \leq p_x \leq 10 \text{ e } 6.5 \leq p_y \leq 10 \\ 1 & \text{caso contrário} \end{cases} \tag{6.3}$$

Os pontos gerados para compor as bases de dados A e B e suas fronteiras de decisão são apresentados nas figuras 6.1.1 e 6.1.1, respectivamente. Os círculos e os asteriscos correspondem às instâncias das classes 0 e 1, respectivamente.

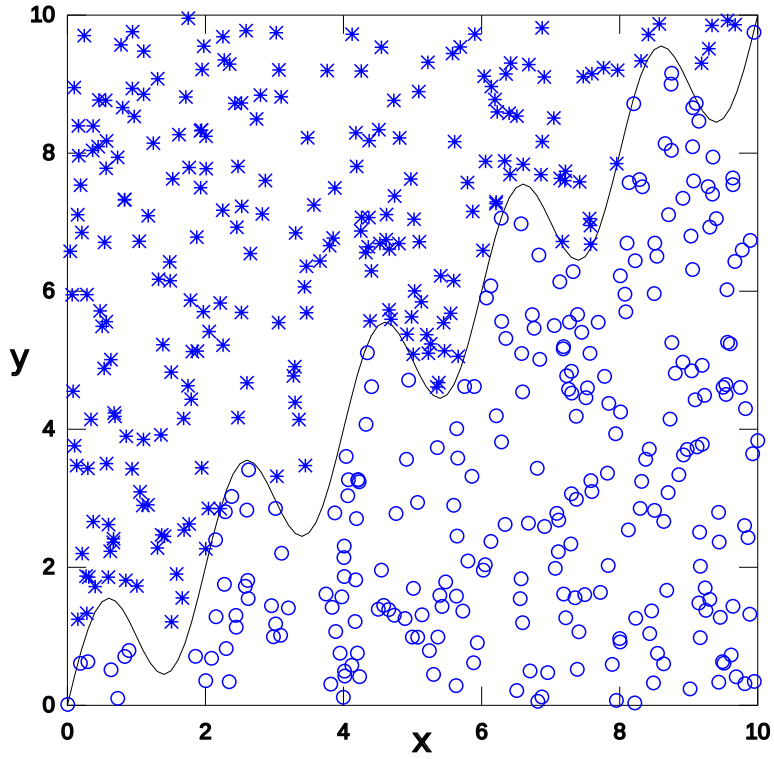


Figura 6.1: Pontos gerados para compor a base A de dados sintéticos e sua fronteira de decisão.

6.1.2 Organização dos Experimentos

Para os experimentos com as bases sintéticas, os valores de entrada do CSCDR foram mantidos invariáveis, com exceção de N. Os parâmetros n_g, d e β foram fixados com os valores 20, 3 e 0,5, respectivamente. Já o parâmetro N foi ajustado para 10, 30 e 50, para a base A, e para 50, 100 e 150, para a base B.

As bases sintéticas foram utilizadas integralmente para treinamento e teste. Cada combinação de parâmetros e base de dados foi testada 20 vezes e a que resultou em melhor

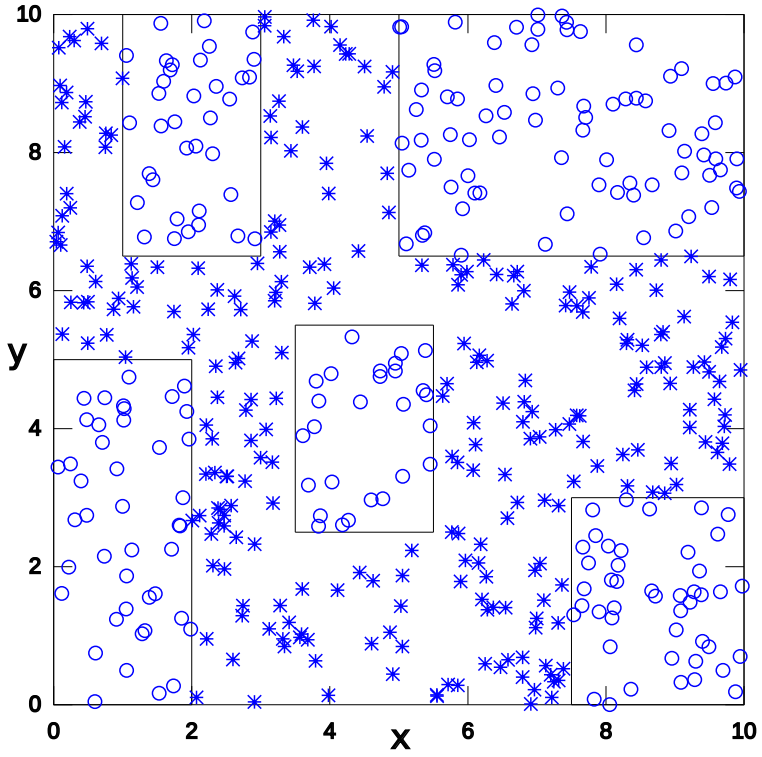


Figura 6.2: Pontos gerados para compor a base B de dados sintéticos e suas fronteiras de decisão.

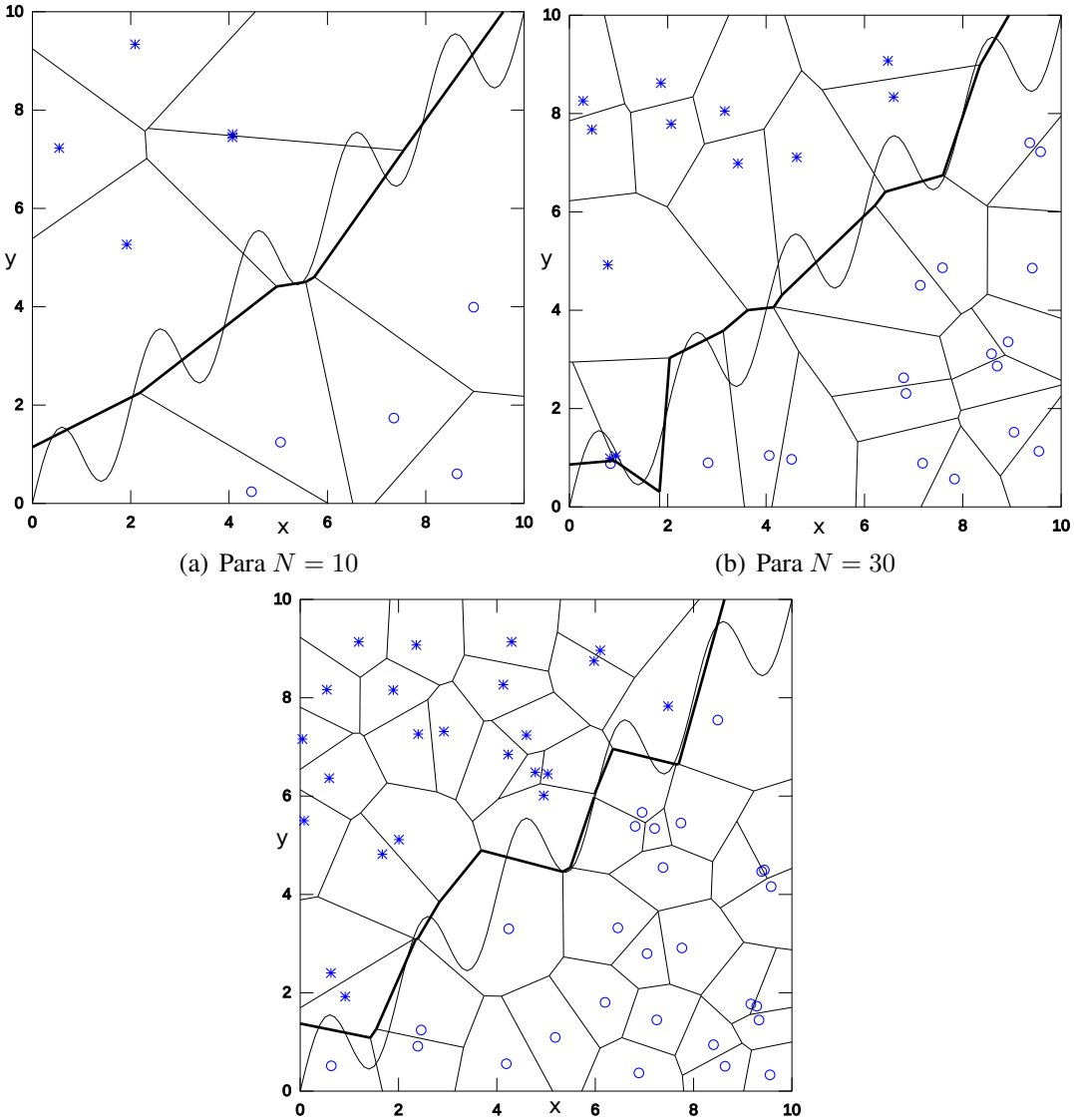
acurácia foi utilizada. Os valores comparativos gerados pelos experimentos limitam-se à posição das células de memória e à acurácia sobre os dados de treinamento.

6.1.3 Resultados

As figuras 6.1.3 e 6.1.3 apresentam as células de memória geradas pelas execuções do algoritmo com as bases A e B como entrada, respectivamente. As figuras apresentam as fronteiras de decisão das bases de dados, definidas nas equações 6.1 e 6.3, os diagramas de Voronoi definidos pelas células de memória e as fronteiras de decisão geradas por elas, representadas por linhas mais escuras.

Como pode ser visto nestas figuras, as células de memória geradas pelo CSCDR conseguiram uma representação crível das classes nestas bases de dados. Pelos diagramas de Voronoi percebe-se que mesmo valores de N pequenos podem gerar centroides capazes de representar as fronteiras de decisão do problema. Na base A especificamente, percebe-se que, a medida que o valor de N aumenta, a fronteira de decisão gerada pelas células de memória se aproxima da fronteira imposta pelo domínio do problema.

A tabela 6.1 apresenta as acurácias obtidas nos testes. Para estas bases em particular, a taxa de acurácia aumentou juntamente com o valor de N. Contudo, isto deve-se ao grau de simplicidade das bases e ao fato de que os conjuntos de treinamento e teste eram os mesmos. Empregando conjuntos diferentes ou bases mais complexas, isto nem sempre acontece.



(a) Para $N = 10$

(b) Para $N = 30$

(c) Para $N = 50$

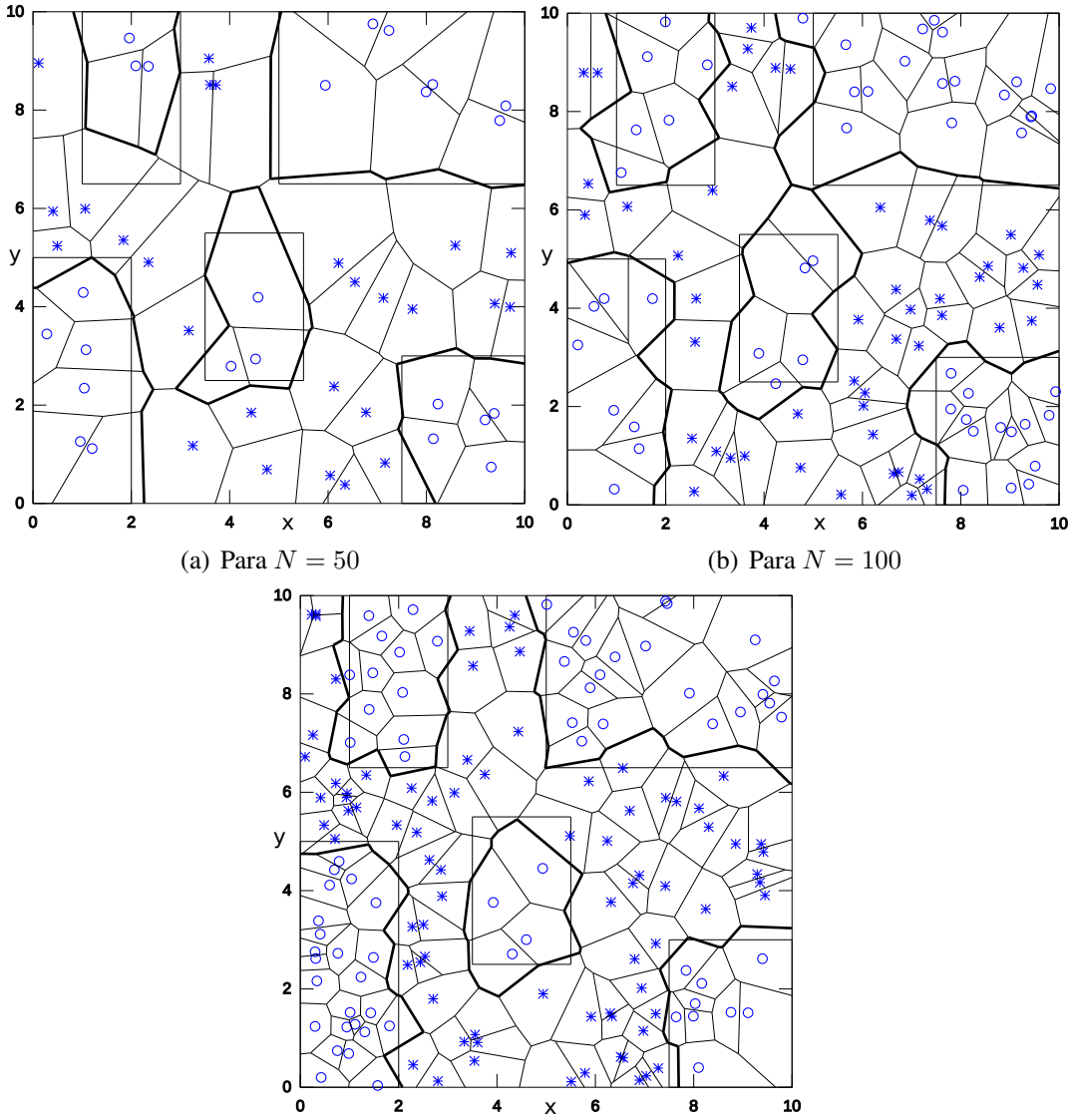
Figura 6.3: Células de memória e diagramas de Voronoi obtidos para o conjunto de dados sintéticos A.

Tabela 6.1: Acurácias obtidas para as bases sintéticas.

Base de dados	N	**Acurácia (%)**
Base A	10	94,40
	30	95,20
	50	96,60
Base B	50	94,20
	100	96,20
	150	96,80

6.2 Estudo Experimental com Bases de Dados de *Benchmark*

Esta seção explora o comportamento do CSCDR em conjuntos de dados utilizados como *benchmark* em problemas de classificação, comparando os resultados obtidos pelo



(a) Para $N = 50$ (b) Para $N = 100$

(c) Para $N = 150$

Figura 6.4: Células de memória e diagramas de Voronoi obtidos para o conjunto de dados sintéticos B.

CSCDR com os resultados obtidos por outros algoritmos da literatura.

Os algoritmos utilizados são o MLP (*multi layer perceptron*) com *backpropagation*, baseado em redes neurais artificiais, o C4.5 (QUINLAN, 1993), baseado em árvores de decisão, o método *k*NN simples (AHA; KIBLER; ALBERT, 1991), baseado em instâncias, e o CSCA (BROWNLEE, 2005) e o AIRS2 (WATKINS; TIMMIS; BOGGESS, 2004), ambos baseados em sistemas imunológicos artificiais.

Os dois primeiros classificadores foram escolhidos por serem comumente utilizados em publicações de aprendizagem de máquina e os três últimos por serem diretamente relacionados ao funcionamento do CSCDR (aprendizado baseado em instância), de forma a comparar a quantidade de instâncias utilizadas pelos algoritmos para representar os dados.

Os algoritmos MLP, C4.5 e *k*NN são providos pelo pacote WEKA (WITTEN; FRANK,

2011) com os nomes *MultilayerPerceptron*, *J48* e *IBk*, respectivamente, e os algoritmos CSCA e AIRS2 são providos por Brownlee (2009).

6.2.1 Bases de Dados

Os conjuntos de dados utilizados nesta seção foram obtidos do repositório de aprendizagem de máquina UCI (University of California, Irvine) (FRANK; ASUNCION, 2010). No total, foram selecionadas seis bases de dados com atributos contínuos, comumente utilizadas em testes com SIAs classificadores (WATKINS, 2001; SECKER; FREITAS, 2007; GOLZARI et al., 2009; OLIVEIRA; MOTA; BARONE, 2012), descritas a seguir.

6.2.1.1 Conjunto de Dados Íris

O conjunto de dados íris, descrito por Fisher (1936), é uma das bases mais conhecidas na literatura de aprendizagem de máquina.

O conjunto é formado por três classes de 50 instâncias cada, onde cada classe corresponde a uma espécie da planta íris: *setosa*, *viginica* e *versicolor*. Uma classe é linearmente separável das outras duas e as outras duas não o são.

É constituído de quatro atributos medidos em centímetros: comprimento e largura da sépala e comprimento e largura da pétala.

6.2.1.2 Conjunto de Dados WDBC

O diagnóstico de tumores na mama tem sido tradicionalmente realizado por biópsias completas, um procedimento cirúrgico invasivo. Em alguns casos, este precedimento pode ser evitado utilizando outro método de diagnóstico, denominado FNA (*fine needle aspiration*), que analisa pequenas quantidades de tecido do tumor.

A base de dados WDBC (*Wisconsin diagnostic breast cancer*) corresponde ao trabalho realizado por Street, Wolberg e Mangasarian (1993), na Universidade de Wisconsin, responsável por analisar imagens obtidas por FNA, processá-las e gerar a base. No total, o conjunto é composto por 569 instâncias, com 30 atributos reais cada, distribuídos em duas classes: 357 instâncias pertencem à classe correspondente aos diagnósticos de tumores benignos e 212 ao diagnóstico de tumores malignos.

6.2.1.3 Conjunto de Dados Ionosfera

Esta base de dados consiste em informações sobre elétrons livres na ionosfera, coletadas de um sistema de radar em Goose Bay, Labrador, Canadá. O sistema é composto por uma matriz faseada de 16 antenas de alta frequência, com o poder total de transmissão na ordem de 6,4 kilowatts. Os retornos "bons" do radar são aqueles mostrando evidências de algum tipo de estrutura na ionosfera e os "maus" retornos são aqueles que não mostram (seus sinais passam pela ionosfera) (FRANK; ASUNCION, 2010).

O conjunto é constituído por 225 instâncias da classe *g* (*good*) e 126 instâncias da classe *b* (*bad*), compostas por 34 atributos numéricos.

6.2.1.4 Conjunto de Dados E. coli

A localização de uma proteína dentro de uma célula é primariamente determinada por sua sequência de aminoácidos. Nakai e Kanehisa (1991) exploraram este fato para desenvolver um sistema especialista baseado em regras para classificar proteínas em suas possíveis localizações celulares, através de sua sequência de aminoácidos, em bactérias Gram-negativas.

As proteínas da bactéria *Escherichia coli* foram classificadas em oito classes diferentes, distribuídas segundo a tabela 6.2, totalizando 336 instâncias, compostas por sete atributos numéricos.

Tabela 6.2: Distribuição das classes do conjunto de dados *E. coli*.

Classe	**Quantidade**
im	77
cp	143
imL	2
omL	5
imU	35
imS	2
om	20
pp	52
TOTAL	**336**

6.2.1.5 Conjunto de Dados Vidros

Fragmentos de vidro são encontrados frequentemente quando cientistas forenses examinam a roupa de um suspeito de um crime, tal como invasão domiciliar, sendo possível determinar a composição elementar e o índice de refração de fragmentos muito pequenos. Evett e Spiehler (1987) descreveram uma base de dados utilizada para classificar amostras de vidros através de indução de regras.

A base é composta por 214 instância de amostras de vidro, divididas em seis classes: 70 correspondentes a janelas de construção (*float*), 17 a janelas de veículos (*float*), 76 a janelas de construção (não *float*), 13 a recipientes, 9 a utensílios de mesa e 29 a faróis.

Cada instância é descrita por nove atributos numéricos, representando o índice de refração do vidro e a porcentagem de oito elementos químicos na composição do vidro.

6.2.1.6 Conjunto de Dados Diabetes

A população de Índios Pima, moradora de uma região próxima a Phoenix, Arizona, foi estudada pelo *National Institute of Diabetes and Digestive and Kidney Diseases* devido à alta taxa de incidência de diabetes (KNOWLER et al., 1981). Cada membro da comunidade com mais de cinco anos de idade passou por um processo de exame padrão, incluindo um teste oral de tolerância à glicose. As informações obtidas foram utilizadas na base de dados descrita por Smith et al. (1988), utilizando informações de pacientes do

sexo feminino, no período de cinco anos.

Os dados são compostos por 768 instâncias, compostas por oito atributos numéricos e divididas em duas classes. A classe 1, com 500 instâncias, representa resultados positivos e a classe 0, com 268 elementos, representa resultados negativos nos testes para o diagnóstico de diabetes.

6.2.2 Métricas Comparativas

São utilizadas duas métricas para comparação do desempenho dos algoritmos. A acurácia é utilizada para comparar o CSCDR a todos os outros algoritmos investigados nesta seção. Já a quantidade de protótipos utilizados pelo algoritmo é utilizada apenas para comparar o CSCDR aos outros classificadores baseados em instâncias.

Existem diversas métricas para avaliar o desempenho preditivo de classificadores, conforme apresentado na Seção 4.3.2. Dentre estas métricas, a mais utilizada pela literatura para comparar classificadores em domínios não-específicos é a acurácia, sendo, desta forma, adotada também para comparar os resultados obtidos nesta seção.

Uma das hipóteses deste trabalho é a de que é possível diminuir a quantidade de protótipos necessários por um classificador baseado em SIA para representar o conjunto de dados de entrada. Desta forma, é adotada a quantidade de células de memória geradas pelos sistemas imunológicos artificiais como forma de verificar a validade da hipótese.

6.2.3 Metodologia

Para estimar a precisão da previsão dos algoritmos, uma abordagem utilizando validação cruzada com dez pastas (*folds*) foi adotada, como sugerido por Kohavi (1995). Devido ao não-determinismo presente, tanto no particionamento aleatório dos dados como na execução dos algoritmos, a validação cruzada com dez pastas foi repetida dez vezes para cada combinação de algoritmo e conjunto de dados.

O *teste-t corrigido* é utilizado para determinar se a média da medida de desempenho de um algoritmo é significativamente maior do que, ou menor que, a média obtida pelo CSCDR. O valor de k das equações 4.4 e 4.5, apresentadas na Seção 4.3.3, é definido como 100, equivalente a dez execuções de validação cruzada com dez pastas. Para 99 graus de liberdade $(k - 1)$ e utilizando um nível de confiança de 95% bicaudal, tem-se $z = \pm 1,98421695$.

6.2.4 Seleção de Parâmetros

Todos os algoritmos apresentados possuem parâmetros de entrada, que normalmente influenciam no resultado dos testes, devendo ser ajustados para cada problema especificamente. A tabela 6.3 apresenta as variações utilizadas para escolher a melhor combinação de parâmetros para cada caso. O parâmetro inicia com o valor da coluna *Início* e é incrementado com o valor do *Passo* até atingir o valor da coluna *Fim*. Todas as variações geradas para cada parâmetro são combinadas entre si e testadas através de validações cruzadas com dez pastas. Aquela que apresenta a maior acurácia é selecionada para os testes.

Tabela 6.3: Variação de parâmetros de entradas dos algoritmos testados.

Algoritmo	Parâmetro	Tipo	Início	Passo	Fim
C4.5	C (fator de confiança)	Real	0,1	0,05	0,5
	M (nº. mín. de instâncias por folha)	Int.	1	1	50
MLP	L (taxa de aprendizagem)	Real	0,1	0,1	0,9
	M (momento)	Real	0,1	0,1	0,9
	N (número máximo de épocas)	Int.	300	100	500
***k*NN**	K (nº. de vizinhos considerados)	Int.	1	1	10
CSCDR	n_g	Int.	5	5	20
	d	Int.	3	1	10
	β	Real	0,1	0,1	2,5

O MLP utilizado não realiza decaimento da taxa de aprendizado e possui apenas uma camada oculta, com a quantidade de neurônios definida segundo a equação 6.4, utilizada por Witten e Frank (2011). Vale notar também que o parâmetro N do algoritmo CSCDR depende do tamanho da conjunto de dados de entrada e, portanto, deve ser ajustado segundo cada conjunto. A tabela 6.4 apresenta as variações deste parâmetro para cada uma das bases de dados.

$$n_{neuronios} = \left\lfloor \frac{n_{atributos} + n_{classes}}{2} \right\rfloor \tag{6.4}$$

Tabela 6.4: Variação do parâmetro N para cada base de dados

Base de dados	Início	Passo	Fim
Íris	30	2	80
WDBC	38	2	80
Ionosfera	36	2	90
E. coli	40	2	70
Vidros	120	2	170
Diabetes	60	2	120

Para os algoritmos CSCA e AIRS2, os mesmos parâmetros utilizados no trabalho de Oliveira, Mota e Barone (2012) são adotados. Os autores utilizam um procedimento de ajuste dos parâmetros semelhante ao adotado neste trabalho. A tabela 6.5 apresenta as combinações de parâmetros selecionados para cada algoritmo/base de dados.

6.2.5 Resultados

Nesta seção, os resultados adquiridos nos experimentos são apresentados e discutidos. Os valores significantemente diferentes daqueles obtidos pelo CSCDR, dentro do intervalo de confiança do *teste-t corrigido*, são grafados em negrito.

A tabela 6.6 mostra as médias das acurácias obtidas nos testes com todos os clas-

Tabela 6.5: Combinações de parâmetros utilizadas para cada algoritmo/base de dados.

Base de dados	**Parâmetros**									
	CSCDR				**MLP**			**C4.5**		***k*NN**
	N	β	d	n_g	N	M	L	M	C	k
Íris	40	1,1	10	10	300	0,9	0,8	1	0,35	6
WDBC	74	1,3	6	10	300	0,8	0,3	5	0,5	9
Ionosfera	70	0,1	7	25	300	0,9	0,6	2	0,2	3
E. coli	54	1,4	10	5	400	0,6	0,8	4	0,3	9
Vidros	160	2,2	8	25	400	0,5	0,1	7	0,35	1
Diabetes	84	1,4	5	10	400	0,7	0,1	44	0,45	7

sificadores e a tabela 6.7 apresenta as médias das quantidades de protótipos utilizados para realizar a classificação e a diferença relativa em relação ao CSCDR, referentes aos resultados adquiridos nos experimentos com os algoritmos baseados em instâncias.

Tabela 6.6: Comparação das acurácias médias obtidas para os algoritmos CSCDR, MLP, C4.5, *k*NN, CSCA e AIRS2

Base de dados	**CSCDR**	**MLP**	**C4.5**	***k*NN**	**CSCA**	**AIRS2**
Íris	96,27	95,93	94,27	96,73	95,87	93,73
WDBC	92,01	**96,54**	93,45	**97,14**	92,15	**95,61**
Ionosfera	89,15	92,20	89,74	**86,02**	87,44	85,55
E. coli	84,38	83,61	81,44	86,90	84,41	84,26
Vidros	70,74	66,14	68,75	70,30	71,05	65,63
Diabetes	73,16	75,27	75,05	74,45	72,88	73,09

Tabela 6.7: Comparação das quantidades médias de protótipos utilizados nos testes pelos algoritmos baseados em instâncias (valores absolutos e diferenças relativas)

Base de dados	**CSCDR**	**kNN**		**CSCA**		**AIRS2**	
		Abs.	Dif.	Abs.	Dif.	Abs.	Dif.
Íris	40,00	**135,00**	237,50%	**28,45**	-28,88%	**25,16**	-37,10%
WDBC	74,00	**512,10**	592,03%	**112,61**	52,18%	**419,66**	467,11%
Ionosfera	70,00	**315,90**	351,29%	**57,60**	-17,71%	**89,82**	28,31%
E. coli	54,00	**302,40**	460,00%	55,40	2,59%	**143,18**	165,15%
Vidros	160,00	**192,60**	20,38%	**150,93**	-5,67%	**127,05**	-20,59%
Diabetes	84,00	**691,20**	722,86%	**108,39**	29,04%	**432,49**	414,87%

Comparando-se as acurácias obtidas pelo CSCDR aos valores obtidos pelo MLP e C4.5, verifica-se que o desempenho dos algoritmos é semelhante e, apesar de as diferenças não serem estatisticamente significantes, os valores médios das acurácias do CSCDR para as bases Íris, *E. coli* e Vidros são superiores aos valores obtidos pelos outros dois classificadores.

Em relação ao *k*NN, o CSCDR apresentou acurácia significativamente superior para os testes com a base Ionosfera, utilizando um número de protótipos cerca de 350% menor. Para as bases Íris, *E. coli*, Vidros e Diabetes, as acurácias dos dois classificadores não apresentam diferenças significativas, porém, o CSCDR utiliza quantidades de protótipos muito abaixo das quantidades utilizadas pelo *k*NN.

O algoritmo AIRS2 também apresentou acurácias similares às obtidas pelo CSCDR para as bases Ionosfera, *E. coli* e Diabetes, utilizando uma quantidade significativamente superior de células de memória, com diferenças variando entre 89,82% e 414,87%.

O algoritmo CSCA, uma das principais bases para o desenvolvimento do CSCDR, foi o algoritmo que apresentou resultados mais parecidos com os obtidos pelo CSCDR, tanto para os valores de acurácia quanto para as quantidades de células de memória geradas. Ainda assim, para as bases Diabetes e WDBC, o CSCA gerou cerca de 29% e 52% mais células de memória, respectivamente, que o algoritmo CSCDR.

6.2.6 Discussão

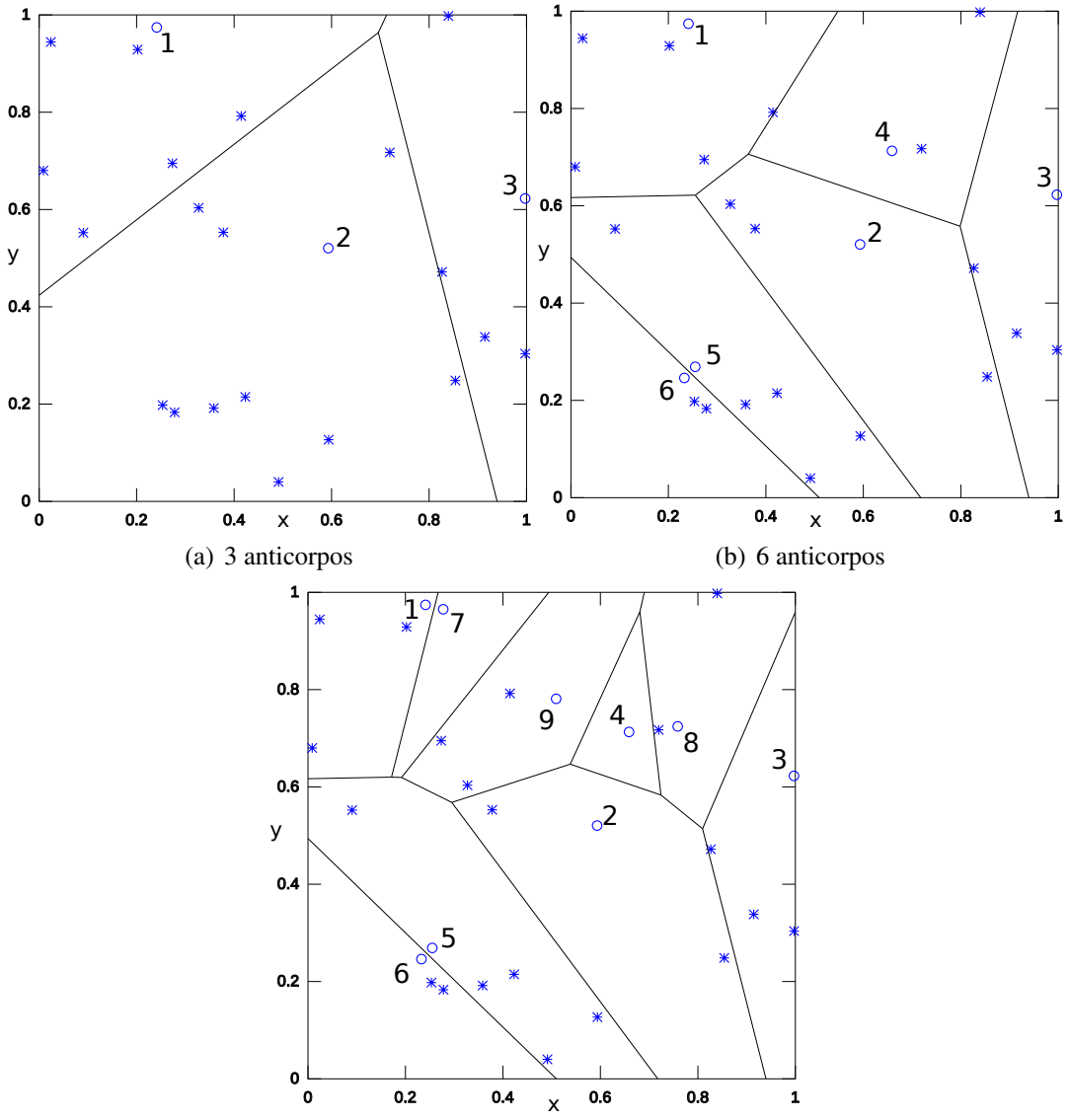
Através dos testes realizados, verifica-se que o CSCDR pode ser utilizado como alternativa para os algoritmos baseados em instância, com o intuito de diminuir a quantidade de protótipos necessários pelo modelo. Quanto maior o número de instâncias utilizadas para representar os dados de treinamento, maior a quantidade de armazenamento necessária para guardar o modelo e mais processamento é despendido para encontrar os vizinhos mais próximos.

Além disso, se o modelo utilizar todo o conjunto de treinamento para efetuar a classificação, como é o caso do *k*NN, ele torna-se mais suscetível a *outliers* do conjunto, o que gera inconsistências durante a classificação. Isto pode ser um dos motivos que levaram o CSCDR a obter melhor acurácia que o kNN para a base Ionosfera, mesmo utilizando uma quantidade muito inferior de instâncias para a classificação.

O trabalho de de Castro e Von Zuben (2002) analisa a sensibilidade do parâmetro N_c, que regula a quantidade de clones produzidos por geração, em um algoritmo CLONALG para otimização numérica. Os autores verificaram que, quanto maior o valor de N_c, mais rápido o algoritmo convergia para o máximo da função, em termos de número de gerações, ou seja, para um mesmo número de gerações, maiores valores de N_c geravam soluções mais próximas do máximo global.

Apesar de o CSCDR ser aplicado a problemas de classificação, ele realiza um processo de otimização numérica, a fim de maximizar o valor de aptidão médio da população de células de memória. Desta forma, espera-se que o parâmetro β, responsável por controlar o tamanho da população de clones no CSCDR, possua o mesmo comportamento de N_c, isto é, que valores maiores de β gerem populações de células de memória com aptidão média igual ou superior à aptidão média das populações geradas com valores menores de β.

Entretanto, durante os testes com o CSCDR, os valores do parâmetro β sofreram variações, de acordo com a tabela 6.3, apresentada na Seção 6.2.4, e, com exceção da base Vidros, os valores que geraram as melhores acurácias ficaram bem abaixo do valor máximo testado. Este fato pode ser explicado pela forma como a aptidão é calculada pelo algoritmo.



(a) 3 anticorpos

(b) 6 anticorpos

(c) 9 anticorpos

Figura 6.5: Diagramas de Voronoi mostrando a representação de 20 antígenos por diferentes quantidades de anticorpos.

A aptidão utiliza o conceito de unidade de melhor correspondência, apresentado na Seção 5.3.2, que está ligado diretamente ao diagrama de Voronoi formado pelos anticorpos. Quanto maior o número de anticorpos, mais fragmentado torna-se o diagrama e, consequentemente, os anticorpos passam a cobrir espaços menores. Desta forma, a média de antígenos cobertos por anticorpo diminui, acarretando problemas no cálculo da aptidão.

A figura 6.2.6 apresenta um exemplo, onde 20 antígenos são cobertos por 3, 6 e 9 anticorpos. Os anticorpos são numerados e representados por círculos e os antígenos são representados por asteriscos.

A tabela 6.8 mostra os valores de aptidão de cada um dos anticorpos para os três casos da figura 6.2.6. Quanto maior a quantidade de anticorpos, menores tornam-se os valores médios das aptidões, dificultando a escolha dos melhores indivíduos. A quantidade de

anticorpos utilizada durante o cálculo de aptidão está relacionada não só ao parâmetro β, mas também ao tamanho da população, definido por N, o que pode acarretar problemas em base de dados extensas, onde o valor de N deve ser suficientemente grande para representar os dados de treinamento.

Tabela 6.8: Aptidão dos anticorpos da figura 6.2.6

Anticorpo	**Aptidão**		
	Fig. (a)	**Fig. (b)**	**Fig. (c)**
1	6	5	3
2	10	3	2
3	4	3	3
4	-	2	0
5	-	5	5
6	-	2	2
7	-	-	0
8	-	-	2
9	-	-	3
Média	**6,67**	**3,33**	**2,22**

Este problema pode ser resolvido substituindo a função de aptidão atual por outra que não seja sensível à quantidade de anticorpos utilizados no treinamento. Com isso, o desempenho do classificador pode melhorar para populações maiores de anticorpos.

7 CONCLUSÕES E PROPOSTAS DE TRABALHOS FUTUROS

Neste capítulo são apresentadas as contribuições e conclusões decorrentes do trabalho realizado e sugestões para futuras pesquisas envolvendo o tema desta dissertação.

7.1 Contribuições

Semelhante a outros paradigmas computacionais biologicamente inspirados, como as redes neurais artificiais e os algoritmos genéticos, os sistemas imunológicos artificiais surgiram como um paradigma de inteligência computacional com o propósito de resolver problemas computacionais complexos como reconhecimento de padrões, otimização e controle (DE CASTRO; TIMMIS, 2002a).

Esta dissertação propõe um novo algoritmo para classificação de dados, baseado nos princípios dos sistemas imunológicos artificiais. Seleção clonal, hipermutação somática e maturação de afinidade são algumas das metáforas retiradas do sistema imunológico biológico e utilizadas nos trabalhos de de Castro (2001), de Castro e Von Zuben (2002) e Brownlee (2005), que foram aplicadas no desenvolvimento deste algoritmo.

A contribuição desta dissertação para o campo de Sistemas Imunológicos Artificiais é a introdução de um método de aprendizado supervisionado, baseado em instâncias, que emprega seleção clonal para otimizar a escolha das instâncias, em conjunto com uma função de aptidão e um parâmetro de controle de população. A dissertação proveu uma descrição detalhada do método, denominado *clonal selection classifier with data reduction* (CSCDR), juntamente com testes em bases de dados reais e sintéticos, utilizando os valores de acurácia e número de protótipos armazenados pelo modelo para comparar os resultados obtidos pelo CSCDR com os resultados obtidos pelos algoritmos MLP, C4.5, *k*NN, CSCA e AIRS2.

7.2 Conclusões

O algoritmo proposto, baseado principalmente no algoritmo CSCA, apresenta modificações em relação ao seu predecessor, garantindo melhorias no controle do tamanho populacional e na forma como o operador de mutação é aplicado. Além disso, quando comparado a outros SIAs classificadores, o CSCDR apresenta menor número de variá-

veis de entrada, facilitando a escolha da melhor combinação de parâmetros para cada problema.

Os resultados obtidos demonstraram que o CSCDR pode ser utilizado como alternativa a outros classificadores baseados em instâncias, com desempenho semelhante e menor número de células de memória. Quando comparado ao *k*NN, AIRS2 e CSCA, o CSCDR utilizou, em alguns casos, quantidades inferiores de protótipos, chegando a diferenças de cerca de 723%, 415% e 52%, respectivamente. Também com relação ao MLP e C4.5, o CSCDR conseguiu resultados suficientemente bons para ser tomado como abordagem alternativa.

Contudo, a partir de algumas observações no comportamento do algoritmo, percebeu-se que a função de aptidão utilizada durante o treinamento é suscetível ao tamanho da população de clones, o que pode afetar seu desempenho em conjuntos de dados maiores.

7.3 Proposta para Trabalhos Futuros

A função de aptidão utilizada pelo modelo implementado neste trabalho apresentou-se suscetível ao tamanho da população de anticorpos, o que acarreta problemas em conjuntos de treinamento muito grandes ou em repertórios extensos de clones. Desta forma, o desempenho do algoritmo pode ser otimizado através do uso de uma nova função de aptidão, que não seja fortemente influenciada pelo tamanho da população de anticorpos.

Uma possibilidade para solucionar este problema inclui o uso de alocação de recursos, apresentado nos trabalhos de Knight e Timmis (2001) e Timmis e Neal (2000) e utilizado no algoritmo AIRS (WATKINS, 2001).

Outra alternativa, é calcular a aptidão para cada clone gerado, excluindo-se do cálculo o anticorpo pai e os outros clones. Desta forma, cada clone é tratado como um candidato a célula de memória, em substituição ao anticorpo original, caso o valor de aptidão do clone seja maior que do indivíduo pai. Para diminuir o custo computacional inerente às buscas pelas UMCs, relacionadas a esta solução, estruturas de dados como as *k-d trees* (FRIEDMAN; BENTLEY; FINKEL, 1977) e *ball trees* (LIU; MOORE; GRAY, 2006) podem ser utilizadas.

Outra forma de aperfeiçoar o método pode ser conseguida através da remoção do parâmetro N. Uma forma automática de calcular o tamanho da população de anticorpos necessários para representar o conjunto de dados, pode apresentar melhorias tanto por subtrair uma variável de entrada a ser ajustada, como por possibilitar uma abordagem mais orientada ao problema.

REFERÊNCIAS

ABBAS, A.; LICHTMAN, A. *Basic immunology: functions and disorders of the immune system*. 2nd. ed. [S.l.]: WB Saunders Company, 2004.

ABBAS, A.; LICHTMAN, A.; PILLAI, S. *Cellular and molecular immunology*. 6th. ed. [S.l.]: Saunders Elsevier, 2007.

ADAMS, D. How the immune system works and why it causes autoimmune diseases. *Immunology today*, Elsevier, v. 17, n. 7, p. 300–302, 1996.

AHA, D.; KIBLER, D.; ALBERT, M. Instance-based learning algorithms. *Machine learning*, Springer, v. 6, n. 1, p. 37–66, 1991.

AICKELIN, U.; DASGUPTA, D. Artificial immune systems. In: BURKE, E.; KENDALL, G. (Ed.). *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*. [S.l.]: Springer, 2005.

BERSINI, H.; VARELA, F. The immune learning mechanisms: Reinforcement, recruitment and their applications. In: *Computing with Biological Metaphors*. [S.l.]: Chapman Hall, 1994. p. 166–192.

BOUCKAERT, R.; FRANK, E. Evaluating the replicability of significance tests for comparing learning algorithms. *Advances in knowledge discovery and data mining*, Springer, p. 3–12, 2004.

BRABAZON, A. et al. Identifying online credit card fraud using artificial immune systems. In: IEEE CONGRESS ON EVOLUTIONARY COMPUTATION (CEC). [S.l.]. Proceedings... 2010. p. 1–7.

BRIGHTON, H.; MELLISH, C. Advances in instance selection for instance-based learning algorithms. *Data mining and knowledge discovery*, Springer, v. 6, n. 2, p. 153–172, 2002.

BROWNLEE, J. *Clonal Selection Theory & CLONALG - The Clonal Selection Classification Algorithm (CSCA)*. Centre for Intelligent Systems and Complex Processes (CISCP), Faculty of Information and Communication Technologies (ICT), Swinburne University of Technology, n. 2-02, 2005.

BROWNLEE, J. *WEKA Classification Algorithms*. 2009. Disponível em: <http://wekaclassalgos.sourceforge.net>. Acesso em: 15 de Novembro de 2011.

BURNET, F. A modification of Jerne's theory of antibody production using the concept of clonal selection. *Australian Journal of Science*, v. 20, n. 3, p. 67–69, 1957.

BUTLER, M.; KAZAKOV, D. Modeling the behavior of the stock market with an artificial immune system. In: IEEE CONGRESS ON EVOLUTIONARY COMPUTATION (CEC). [S.l.]. Proceedings... 2010. p. 1–8.

CARTER, J. The immune system as a model for pattern recognition and classification. *Journal of the American Medical Informatics Association*, American Medical Informatics Association, v. 7, n. 1, p. 28–41, 2000.

CHOWDHURY, D.; STAUFFER, D. Statistical physics of immune networks. *Physica A: Statistical Mechanics and its Applications*, Elsevier, v. 186, n. 1, p. 61–81, 1992.

CHOWDHURY, D.; STAUFFER, D.; CHOUDARY, P. A unified discrete model of immune response. *Journal of theoretical Biology*, Elsevier, v. 145, n. 2, p. 207–215, 1990.

COVER, T.; HART, P. Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, IEEE, v. 13, n. 1, p. 21–27, 1967.

CZIKO, G. The immune system: Selection by the enemy. In: *Without miracles: Universal selection theory and the second Darwinian revolution*. [S.l.]: The MIT Press, 1997.

DASGUPTA, D.; NIÑO, L. *Immunological computation: theory and applications*. [S.l.]: CRC Press, 2008. 277 p.

DASGUPTA, D.; YU, S.; NINO, F. Recent advances in artificial immune systems: models and applications. *Applied Soft Computing*, Elsevier, v. 11, n. 2, p. 1574–1587, 2011.

DE CASTRO, L. N. *Engenharia Imunológica: Desenvolvimento e Aplicação de Ferramentas Computacionais Inspiradas em Sistemas Imunológicos Artificiais*. Tese (Doutorado) — Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, Brasil, 2001.

DE CASTRO, L. N.; TIMMIS, J. *Artificial immune systems: a new computational intelligence approach*. [S.l.]: Springer-Verlag, 2002.

DE CASTRO, L. N.; TIMMIS, J. Artificial immune systems: A novel paradigm to pattern recognition. In: CORCHADO, J.; ALONSO, L.; FYFE, C. (Ed.). *Artificial Neural Networks in Pattern Recognition*. [S.l.]: University of Paisley, 2002. p. 67–84.

DE CASTRO, L. N.; TIMMIS, J. Artificial immune systems as a novel soft computing paradigm. *Soft Computing*, Springer, v. 7, n. 8, p. 526–544, 2003.

DE CASTRO, L. N.; VON ZUBEN, F. Learning and optimization using the clonal selection principle. *Evolutionary Computation, IEEE Transactions on*, IEEE, v. 6, n. 3, p. 239–251, 2002.

DE CASTRO, L. N.; VON ZUBEN, F. J. *Artificial Immune Systems: Part I - Basic Theory and Applications*. Department of Computer Engineering and Industrial Automation, School of Electrical and Computer Engineering, State University of Campinas, Campinas, Brazil, 1999.

DE CASTRO, L. N.; VON ZUBEN, F. J. The clonal selection algorithm with engineering applications. In: GENETIC AND EVOLUTIONARY COMPUTATION CONFERENCE (GECCO). [S.l.]. Proceedings... 2000. p. 36–37.

DELIBASIS, K. et al. Computer-aided diagnosis of thyroid malignancy using an artificial immune system classification algorithm. *Information Technology in Biomedicine, IEEE Transactions on*, IEEE, v. 13, n. 5, p. 680–686, 2009.

DING, S.; LI, S. Clonal selection algorithm for feature selection and parameters optimization of support vector machines. In: INTERNATIONAL SYMPOSIUM ON KNOWLEDGE ACQUISITION AND MODELING (KAM). [S.l.]. Proceedings... 2009. v. 2, p. 17–20.

EVETT, I. W.; SPIEHLER, E. J. Rule induction in forensic science. In: KBS IN GOVERMENT. [S.l.]: Online Publications. Proceedings... 1987. p. 107–118.

FARMER, J.; PACKARD, N.; PERELSON, A. The immune system, adaptation, and machine learning. *Physica D: Nonlinear Phenomena*, Elsevier, v. 22, n. 1, p. 187–204, 1986.

FISHER, R. The use of multiple measurements in taxonomic problems. *Annals of Human Genetics*, Wiley Online Library, v. 7, n. 2, p. 179–188, 1936.

FORREST, S. et al. Using genetic algorithms to explore pattern recognition in the immune system. *Evolutionary computation*, MIT Press, v. 1, n. 3, p. 191–211, 1993.

FRANK, A.; ASUNCION, A. *UCI Machine Learning Repository*. 2010. Disponível em: <http://archive.ics.uci.edu/ml>. Acesso em: 29 de Novembro de 2012.

FRIEDMAN, J.; BENTLEY, J.; FINKEL, R. An algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematical Software (TOMS)*, ACM, v. 3, n. 3, p. 209–226, 1977.

GARAIN, U.; CHAKRABORTY, M.; DASGUPTA, D. Recognition of handwritten indic script using clonal selection algorithm. *Artificial Immune Systems*, Springer, p. 256–266, 2006.

GARRETT, S. How do we evaluate artificial immune systems? *Evolutionary Computation*, MIT Press, v. 13, n. 2, p. 145–177, 2005.

GOLDSBY, R. et al. *Immunology*. 5th. ed. [S.l.]: W.H. Freeman, 2003.

GOLZARI, S. et al. Improving the accuracy of AIRS by incorporating real world tournament selection in resource competition phase. In: IEEE CONGRESS ON EVOLUTIONARY COMPUTATION (CEC). [S.l.]. Proceedings... 2009. p. 3040–3044.

HAN, J.; KAMBER, M.; PEI, J. *Data Mining: Concepts and Techniques*. 3rd. ed. [S.l.]: Elsevier Science, 2011. (The Morgan Kaufmann Series in Data Management Systems).

HART, E.; TIMMIS, J. Application areas of ais: The past, the present and the future. *Applied Soft Computing*, Elsevier, v. 8, n. 1, p. 191–201, 2008.

HIGHTOWER, R.; FORREST, S.; PERELSON, A. S. The baldwin effect in the immune system: Learning by somatic hypermutation. In: ADAPTIVE INDIVIDUALS IN EVOLVING POPULATIONS: MODELS AND ALGORITHMS. [S.l.]: Addison-Wesley. Proceedings... 1996. p. 159–167.

HILBERT, M.; LÓPEZ, P. The world's technological capacity to store, communicate, and compute information. *Science*, American Association for the Advancement of Science, v. 332, n. 6025, p. 60, 2011.

HING, K. L. K.; CHEONG, F.; CHEONG, C. Consumer credit scoring using an artificial immune system algorithm. In: IEEE CONGRESS ON EVOLUTIONARY COMPUTATION (CEC). [S.l.]. Proceedings... 2011. p. 3377–3384.

IGAWA, K.; OHASHI, H. A negative selection algorithm for classification and reduction of the noise effect. *Applied Soft Computing*, Elsevier, v. 9, n. 1, p. 431–438, 2009.

JERNE, N. The natural-selection theory of antibody formation. *Proceedings of the National Academy of Sciences of the United States of America*, National Academy of Sciences, v. 41, n. 11, p. 849–857, 1955.

JERNE, N. K. Towards a network theory of the immune system. *Annales d'immunologie*, v. 125C, n. 1-2, p. 373–389, 1974.

KNIGHT, T.; TIMMIS, J. *Assessing the performance of the resource limited artificial immune system AINE*. Computing Laboratory, University of Kent, Kent, UK, 2001.

KNOWLER, W. et al. Diabetes incidence in pima indians: contributions of obesity and parental diabetes. *American Journal of Epidemiology*, Oxford Univ Press, v. 113, n. 2, p. 144–156, 1981.

KOHAVI, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE. [S.l.]. Proceedings... 1995. v. 14, p. 1137–1145.

LIU, T.; MOORE, A.; GRAY, A. New algorithms for efficient high-dimensional nonparametric classification. *The Journal of Machine Learning Research*, JMLR. org, v. 7, p. 1135–1158, 2006.

MALE, D. et al. *Immunology*. 7th. ed. [S.l.]: Mosby Elsevier, 2006.

MARKOWSKA-KACZMAR, U.; KORDAS, B. Negative selection based method for multi-class problem classification. In: INTERNATIONAL CONFERENCE ON INTELLIGENT SYSTEMS DESIGN AND APPLICATIONS. [S.l.]. Proceedings... 2006. v. 2, n. 6, p. 1165–1170.

MARKOWSKA-KACZMAR, U.; KORDAS, B. Multi-class iteratively refined negative selection classifier. *Applied Soft Computing*, Elsevier, v. 8, n. 2, p. 972–984, 2008.

MARSLAND, S. *Machine Learning: An Algorithmic Perspective*. [S.l.]: CRC Press, 2009.

MCEWAN, C.; HART, E. On AIRS and clonal selection for machine learning. *Artificial Immune Systems*, Springer, p. 67–79, 2009.

MITCHELL, T. *Machine Learning (Mcgraw-Hill International Edit)*. 1st. ed. [S.l.]: McGraw-Hill Education (ISE Editions), 1997. Paperback.

MURPHY, K.; TRAVERS, P.; WALPORT, M. *Imunobiologia de Janeway*. 7th. ed. [S.l.]: Artmed, 2010.

NADEAU, C.; BENGIO, Y. Inference for the generalization error. *Machine Learning*, Springer, v. 52, n. 3, p. 239–281, 2003.

NAKAI, K.; KANEHISA, M. Expert system for predicting protein localization sites in gram-negative bacteria. *Proteins: Structure, Function, and Bioinformatics*, Wiley Online Library, v. 11, n. 2, p. 95–110, 1991.

OLIVEIRA, L.; DRUMMOND, I. Real-valued negative selection (RNS) for classification task. In: ÜNAY, D.; ÇATALTEPE, Z.; AKSOY, S. (Ed.). In: RECOGNIZING PATTERNS IN SIGNALS, SPEECH, IMAGES AND VIDEOS. [S.l.]: Springer Berlin / Heidelberg. Proceedings... 2010. (Lecture Notes in Computer Science, v. 6388), p. 66–74.

OLIVEIRA, L.; DRUMMOND, I. Real-valued negative selection (rns) for mr brain image classification. In: HAMZA, M.; ZHANG, J. (Ed.). In: SIGNAL PROCESSING, PATTERN RECOGNITION, AND APPLICATIONS / COMPUTER GRAPHICS AND IMAGING. Innsbruck, Austria. Proceedings... 2011.

OLIVEIRA, L. O. V. B.; MOTA, R.; BARONE, D. Clonal selection classifier with data reduction: Classification as an optimization task. In: IEEE CONGRESS ON EVOLUTIONARY COMPUTATION (CEC). Brisbane, Austrália. Proceedings... 2012. p. 231–237.

PERELSON, A. Immune network theory. *Immunological Reviews*, Wiley Online Library, v. 110, n. 1, p. 5–36, 1989.

PERELSON, A.; OSTER, G. Theoretical studies of clonal selection: minimal antibody repertoire size and reliability of self-non-self discrimination. *Journal of theoretical biology*, Elsevier, v. 81, n. 4, p. 645–670, 1979.

QUINLAN, J. *C4. 5: programs for machine learning*. San Mateo, CA: Morgan kaufmann, 1993.

REINARTZ, T. A unifying view on instance selection. *Data Mining and Knowledge Discovery*, Springer Netherlands, v. 6, p. 191–210, 2002. ISSN 1384-5810.

SECKER, A.; FREITAS, A. WAIRS: Improving classification accuracy by weighting attributes in the AIRS classifier. In: IEEE CONGRESS ON EVOLUTIONARY COMPUTATION (CEC). [S.l.]. Proceedings... 2007. p. 3759–3765.

SMITH, J. et al. Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. In: ANNUAL SYMPOSIUM ON COMPUTER APPLICATION IN MEDICAL CARE. [S.l.]. Proceedings... 1988. p. 261–265.

STREET, N.; WOLBERG, W.; MANGASARIAN, O. Nuclear feature extraction for breast tumor diagnosis. In: INTERNATIONAL SYMPOSIUM ON ELECTRONIC IMAGING: SCIENCE AND TECHNOLOGY. San Jose, California: [s.n.]. Proceedings... 1993. v. 1905, p. 861–870.

TIMMIS, J. et al. Theoretical advances in artificial immune systems. *Theoretical Computer Science*, Elsevier, v. 403, n. 1, p. 11–32, 2008.

TIMMIS, J.; NEAL, M. Investigating the evolution and stability of a resource limited artificial immune system. In: SPECIAL WORKSHOP ON ARTIFICIAL IMMUNE SYSTEMS, GENTIC AND EVOLUTIONAY COMPUTTION CONFERENCE (GECCO). [S.l.]. Proceedings... 2000. p. 40–41.

ULUTAS, B. H.; KULTUREL-KONAK, S. A review of clonal selection algorithm and its applications. *Artificial Intelligence Review*, Springer, v. 36, n. 2, p. 117–138, 2011.

WATKINS, A. *AIRS: A resource limited artificial immune classifier*. Dissertação (Mestrado) — Mississippi State University, 2001.

WATKINS, A.; BI, X.; PHADKE, A. Parallelizing an immune-inspired algorithm for efficient pattern recognition. *Intelligent Engineering Systems through Artificial Neural Networks: Smart Engineering System Design: Neural Networks, Fuzzy Logic, Evolutionary Programming, Complex Systems and Artificial Life*, Citeseer, v. 13, p. 225–230, 2003.

WATKINS, A.; TIMMIS, J. Exploiting parallelism inherent in AIRS, an artificial immune classifier. *Artificial Immune Systems*, Springer, p. 427–438, 2004.

WATKINS, A.; TIMMIS, J.; BOGGESS, L. Artificial immune recognition system (AIRS): An immune-inspired supervised learning algorithm. *Genetic Programming and Evolvable Machines*, Springer, v. 5, n. 3, p. 291–317, 2004.

WHITE, J.; GARRETT, S. Improved pattern recognition with artificial clonal selection? *Artificial Immune Systems*, Springer, p. 181–193, 2003.

WITTEN, I.; FRANK, E. *Data Mining: Practical machine learning tools and techniques*. 3rd. ed. [S.l.]: Morgan Kaufmann, 2011.

WOJTUSIAK, J. Machine learning. In: SEEL, N. (Ed.). *Encyclopedia of the Sciences of Learning*. [S.l.]: Springer, 2011.

ZHAO, W.; DAVIS, C. A modified artificial immune system based pattern recognition approach - An application to clinical diagnostics. *Artificial Intelligence in Medicine*, Elsevier, 2011.

GLOSSÁRIO

A

Anticorpo
(ou imunoglobulina ou gamaglobulina) Glicoproteína sintetizada e excretada por células plasmáticas derivadas dos linfócitos B, os plasmócitos, presentes no plasma, tecidos e secreções, que atacam antígenos, realizando assim a defesa do organismo (imunidade humoral).

Antígeno
Molécula que se liga a um anticorpo ou ao receptor de antígeno de uma célula T (TCR).

B

Basófilo
Leucócito de núcleo lobulado, que contém, no citoplasma, grânulos que se coram pelos corantes básicos.

Benchmark
Ponto de referência estabelecido com o qual computadores ou programas podem ser medidos em testes comparando suas performances, confiança, etc.

C

Citocina
Pequenas moléculas de proteínas envolvidas na emissão de sinais entre as células durante o desencadeamento das respostas imunes.

Complexo principal de histocompatibilidade
Grande região genômica ou família de genes encontrada na maioria dos vertebrados. É a região mais densa de genes do genoma dos mamíferos e possui importante papel no sistema imune, auto-imunidade e no sucesso reprodutivo.

Crossover
Na *biologia*, é a troca de material genético entre cromossomos homólogos. Em *Algoritmos Genéticos*, é um operador genético utilizado para variar a programação de um ou mais cromossomos de uma geração para a outra, análogo ao crossover biológico.

Célula exterminadora natural
Tipo de linfócito que tem um papel importante no combate a infecções virais e células tumorais. Seu nome provem da sua atividade citotóxica contra células tumorais de diferentes linhagens.

Célula-tronco
Célula primitiva, produzida durante o desenvolvimento do organismo, que dá origem a outros tipos de células.

E

Efeito Baldwin
Resultado da interação entre evolução e aprendizado em animais individuais, durante sua vida. A teoria, proposta por James Mark Baldwin, propõe que a aprendizagem individual pode melhorar a aprendizagem evolutiva em nível de espécie.

Eosinófilo
Leucócito ou outro granulócito com inclusões citoplásmicas, facilmente corável com eosina.

Epitélio
Tecido formado por células intimamente unidas entre si. Sua principal função é revestir a superfície externa do corpo, os órgãos e as cavidades corporais internas. A perfeita união entre as células epiteliais fazem com que os epitélios sejam eficientes barreiras contra a entrada de agentes invasores e a perda de líquidos corporais.

Epítopo
(ou determinante antigênico) É a menor porção de antígeno com potencial de gerar a resposta imune. É a área da molécula do antígeno que se liga aos receptores celulares e aos anticorpos.

Exabyte
Unidade de medida de informação que equivale a 2^{60} bytes.

G

Granulócito
Célula que, em seu protoplasma, contém grânulos basófilos, eosinófilos e neutrófilos.

Grânulos citoplasmáticos
Áreas condensadas de material celular, que podem estar ligadas por uma membrana.

H

Humor
Qualquer líquido que atue normalmente no corpo (bílis, sangue, linfa, etc.).

I

Idiotipo
Conjunto de epítopos exibido pelas regiões variáveis de um conjunto de moléculas de anticorpo.

Idiotopo
Cada epítopo idiotípico único.

L

Linfoblasto
Linfócito estimulado por um antígeno, precursor das células efetoras.

Linfonodo
(ou gânglio linfático) Pequenos órgão perfurado por canais que existe em diversos pontos da rede linfática, uma rede de ductos que faz parte do sistema linfático.

Linfócito
(ou glóbulo branco) Célula produzida na medula óssea e presentes no sangue, linfa, órgãos linfoides e vários tecidos conjuntivos.

Linfócito B
(ou célula B) É um tipo de linfócito que constitui o sistema imune. Ele tem um importante papel na imunidade humoral e é um essencial componente do sistema imune adaptativo. A principal função das células B é a produção de anticorpos contra antígenos. Após sua ativação os linfócitos B podem sofrer diferenciação em plasmócitos ou células B de memória.

Linfócito T
(ou célula T) Pertence a um grupo de glóbulos brancos do sangue e é o principal efector da imunidade celular. É fabricado na medula óssea e sofre posterior maturação no timo a partir de precursores indiferenciados da medula óssea.

M

Macrófago
Célula fagocitária de grandes dimensões.

Medula óssea
Tecido gelatinoso que preenche a cavidade interna de vários ossos e fabrica os elementos figurados do sangue periférico, tais como hemácias, leucócitos e plaquetas.

Mitótico
Relativo a mitose, processo pelo qual as células eucarióticas dividem seus cromossomos entre duas células menores.

Morfologicamente
Relativo à forma.

Multilobulado
Dividido em muitos lóbulos, isto é, dividido em diversas estruturas de forma arredondada.

N

Neutrófilo
Leucócito fagocítico principal do sangue, com finos grânulos citoplasmáticos que se coloram indiferentemente com a fração ácida ou básica dos corantes de sangue comuns.

P

Paratopo
Porção da molécula de anticorpo responsável por reconhecer (complementarmente) um epítopo.

Patógeno
Causa específica de uma doença, como uma bactéria ou um vírus.

S

Sistema complemento
Proteínas do soro sanguíneo e superfície das células que interagem entre si e com outras moléculas do sistema imunológico, de forma altamente regulada, para eliminar microrganismos.

Sistema linfático
Sistema de vasos espalhados pelo corpo que coleta fluido dos tecidos (linfa), originalmente derivado do sangue e retorna-o, pelo duto torácico, para a circulação. Gânglios linfáticos são intercalados ao longo destes vasos e retém antígenos presentes nos gânglios.

T

Timo
Órgão linfático que está localizado na porção antero-superior da cavidade torácica. É vital contra a autoimunidade.