COPPE/UFRJ

UMA METODOLOGIA PARA EXTRAÇÃO DE INFORMAÇÃO SOBRE O SISTEMA IMUNOLÓGICO

Luciana Itida Ferrari

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia de Sistemas e Computação.

Orientadores: Luis Alfredo Vidal de Carvalho
Inês de Castro Dutra

Rio de Janeiro
Dezembro de 2008

UMA METODOLOGIA PARA EXTRAÇÃO DE INFORMAÇÃO SOBRE O SISTEMA IMUNOLÓGICO

Luciana Itida Ferrari

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Aprovada por:

Prof. Luis Alfredo Vidal de Carvalho, D.Sc.

Prof. Nelson Maculan Filho, D.Sc.

Prof. Felipe Maia Galvão França, Ph.D.

Prof. Rodrigo Varejão Andreão, D.Sc.

Dr. Beatriz Stransky Ferreira, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

DEZEMBRO DE 2008

Ferrari, Luciana Itida

Uma Metodologia para Extração de Informação sobre o Sistema Imunológico/ Luciana Itida Ferrari – Rio de Janeiro: UFRJ/COPPE, 2008.

XII, 90 p.: il.; 29,7 cm.

Orientadores: Luis Alfredo Vidal de Carvalho

Inês de Castro Dutra

Tese (doutorado) – UFRJ/ COPPE/ Programa de Engenharia de Sistemas e Computação, 2008.

Referencias Bibliográficas: p. 84-90.

1. Mineração de Dados. 2. Sistema Imunológico. 3. Bioinformática. I. Carvalho, Luis Alfredo Vidal de *et al*. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Titulo.

Ao meu Pai, Luiz Paiva Ferrari

Agradecimentos

Em primeiro lugar, tenho que agradecer ao meu Pai. Sem você eu não teria chegado ao fim deste trabalho. Obrigada por seu apoio emocional e financeiro durante todo esse processo.

Agradeço à COPPE Sistemas e Computação pela oportunidade de estudar lá. Em especial, agradeço ao Luis Alfredo e à Inês Dutra pela orientação deste trabalho e pela amizade desde antes de entrar pra COPPE.

Aos membros convidados para a banca, Prof. Nelson Maculan, Prof. Felipe França, Prof. Rodrigo Varejão e Beatriz Stransky, obrigada por aceitarem participar. Em especial, Bia, obrigada pela amizade e pela ótima parceria!

Aos amigos de Vitória, da UFES (Lorena, Prof. Rodrigo, Prof. Elias, Prof. Alimatéia, Prof. Marcelo, Prof. Attílio, Delu, Marisa, Rosa, e os alunos da Arqui e Biblio), e do CEET (Marcelo, Luziane, Zirlene, Francisca, Luciano, Silvio, Marcelo C., Débora, os outros professores, e os alunos do Técnico em Informática), que me ajudaram a lembrar aonde estava a minha motivação para terminar este projeto. Obrigada pela ajuda na fase final de confecção da tese, me emprestando sala, computador, e cobrindo minhas faltas!

Aos muitos amigos (Cris, Lê, Lê Leal, Zé Afonso, Paty, Fê, Rog, etc...) e familiares (Tias, Mauro, Débora, Márcia e Lelê, e os Itida), pela força, e por tolerarem meu sumiço nestes últimos tempos. Agradecimentos especiais a Paula e André por terem salvado minha vida algumas vezes!

Aos meus colegas de Mestrado, que me incentivaram a voltar e fazer o Doutorado... provavelmente sem esse incentivo o tema desta tese teria sido bem diferente!

Ao Prof. Alberto Nóbrega, sou muito grata por ter me acolhido em seu laboratório, por ter me aproximado do grupo de Imuno do Instituto de Microbiologia da UFRJ, e por me ajudar a fazer contatos para o estágio externo.

Agradeço à CAPES pela bolsa PDEE, pela oportunidade de passar um ano em Portugal, aprendendo muito, e fazendo muitas amizades. Um agradecimento especial ao Prof. Jorge Carneiro, por me acolher no seu grupo e orientar metade desta tese, e ao Rui Gardner, por dividir comigo seu trabalho. Obrigada aos amigos do IGC Tiago, Nuno e Andreas que me ajudaram diretamente na tese, e obrigada vocês 3 e mais o resto da turma (Daniel, Lurdes, Joaninha, Iris, José Afonso, etc...) que me ajudaram a conhecer e entender Portugal. Agradecimentos também para os brasileiros em solo luso pela amizade, e um agradecimento especial para Ju Lamaro e Ana Paula, que me ajudaram a manter a estabilidade emocional durante este processo.

Por fim, um agradecimento muito especial ao Carlos Vieira, companheiro de altos e baixos, parceiro de todas as horas, melhor amigo, e meu amor. Em Vitória o horizonte é bem maior do que em Petrópolis.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

UMA METODOLOGIA PARA EXTRAÇÃO DE INFORMAÇÃO SOBRE O SISTEMA IMUNOLÓGICO

Luciana Itida Ferrari

Dezembro/2008

Orientadores : Luis Alfredo Vidal de Carvalho
Inês de Castro Dutra

Programa: Engenharia de Sistemas e Computação

Utilizando técnicas inovadoras como Imunoblots e *microarrays,* especialistas em imunologia procuraram analisar de forma global o sistema imunológico. Um experimento investigou a capacidade de regeneração do repertório de anticorpos naturais de camundongos, após destruição de linfócitos. Este foi feito com Imunoblots, caracterizando a reatividade dos anticorpos frente à centenas/milhares de antígenos. Outro experimento teve como objetivo estimar a diversidade de linfócitos do repertório de um organismo, e para tal utilizou *microarrays* para consultar a reatividade de milhares de seqüências de uma só vez.

Experimentos deste tipo podem ser melhor investigados e analisados com o auxílio de ferramentas computacionais que não se restrinjam apenas à métodos básicos numéricos e estatísticos.

Neste trabalho, estudamos as características principais de experimentos como os acima mencionados, e com a colaboração e validação de especialistas na área, criamos uma metodologia para auxiliar profissionais não especialistas em matemática e computação, a obter resultados mais qualitativos a partir de seus dados experimentais. Aplicamos métodos de análise estatística (como histogramas, *box-plots*, e gráficos de função de distribuição cumulativa), algoritmos de agrupamento (*k-means*, *fuzzy c-means* e mapas auto-organizáveis) e modelagem computacional para simulação de dados experimentais, para estudar com mais profundidade dados relacionados aos experimentos de análise de repertório.

A utilização da metodologia criada permitiu a obtenção de resultados relevantes em imunologia, e criou a oportunidade de integração dos diferentes profissionais, facilitando a investigação interdisciplinar.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

A METHODOLOGY TO EXTRACT INFORMATION ABOUT THE IMMUNE SYSTEM

Luciana Itida Ferrari

December/2008

Advisors: Luis Alfredo Vidal de Carvalho
Inês de Castro Dutra

Department: Systems Engineering and Computer Science

Using innovative techniques like Immunoblots and microarrays, immunology specialists have done some global analysis of the immune system. One of these experiments investigated the regeneration capacity of the natural antibody repertoire from mice, after destruction of their lymphocytes. This experiment used Immunoblots, characterizing the antibody reactivity with hundreds/thousands of antigens. Another experiment aimed to estimate the diversity of the lymphocytes repertoire from a given organism, and used microarrays to interrogate the expression of thousands of sequences at once.

These experiments can be better investigated and analysed with the help of computational tools that do more than basic numeric and statistical analysis.

In the present work, we studied the main characteristics from experiments like the ones mentioned above, and with the colaboration and validation of specialists in this area, we created a methodology to help professionals that are not specialists in mathematics and computation, to obtain more qualitative results from their experimental data. We used statistical analysis (such as histograms, box-plots, and cumulative distribution function plots), clustering algorithms (k-means, fuzzy c-means and self-organizing maps) and simulation of experimental data though computational modeling, to study deeply these data related to the repertoire analysis experiments.

The use of the proposed methodology revealed relevant results in immunology, and created the opportunity to integrate different professionals, facilitating interdisciplinar research.

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

Sabe-se que a KDD (*Knowledge Discovery in Databases* - descoberta de conhecimento em bancos de dados) é uma metodologia para análise inteligente de dados que tem sido aplicada com sucesso em diversos domínios, como Engenharia, Biologia, Medicina, Mercado Financeiro, e Marketing, por exemplo. Uma das etapas da KDD é a Mineração de Dados, que se apóia em métodos vindos de diversas áreas, entre elas a Inteligência Artificial e a Estatística, para extrair informação oculta em Bancos de Dados. Além disso, pode ser utilizada também modelagem computacional para simular o comportamento de dados existentes, a fim de compreendê-los melhor.

No presente trabalho, será utilizada como base a metodologia da KDD, experimentando várias técnicas de mineração de dados e de criação de modelos computacionais em resultados de experimentos sobre o sistema imunológico feitos por equipes de Biólogos. O objetivo é criar metodologias para análise destes dados que forneçam informações relevantes sobre estes resultados obtidos anteriormente, que melhorem significativamente a visualização dos dados, e que indiquem direções para futuros experimentos. Estas metodologias recorrem tanto a métodos estatísticos, para estudar a natureza dos dados, quanto a métodos mais inteligentes de mineração de dados, como algoritmos de classificação e extração de regras.

Inicialmente, os dados, representados computacionalmente em forma de matrizes, foram pré-processados. Então, foram aplicados diferentes métodos de mineração de dados encontrados na literatura, baseados em:

- análise estatística básica: matrizes de correlação, *box-plots*, histogramas, funções de distribuição cumulativa, e indicadores como média e mediana;
- análise multivariada: análise de componentes principais;
- algoritmos de agrupamento: *k-means*, *fuzzy c-means* e mapas auto-organizáveis;
- extração de regras de associação *fuzzy*;

Também foi criado um modelo computacional capaz de simular o comportamento dos experimentos biológicos originais para posterior análise. Todos os resultados obtidos ficam então disponíveis para as equipes de Biólogos, para que estes façam uma avaliação da relevância e validação destes.

Devido a esse tipo de interação entre equipes, a área interdisciplinar conhecida como *Bioinformática* vem ganhando cada vez mais espaço nas pesquisas. O aumento no poder computacional e a impressionante melhora tecnológica nos equipamentos para realização de experimentos biológicos têm incentivado a convergência de pesquisadores de ambas as áreas para a Bioinformática. Pelo lado da biologia, a realização de experimentos cada vez mais complexos e específicos tem gerado uma enorme quantidade de dados, que necessita tratamento computacional (por exemplo, nas pesquisas sobre o seqüenciamento de genomas). Por outro lado, recentes descobertas da biologia inspiram a criação de novos algoritmos e modelos, como as redes neuronais, algoritmos genéticos, e sistemas imunológicos artificiais. A troca de idéias, informações e inspirações têm sido muito rica entre profissionais destas áreas.

A grande maioria dos trabalhos de bioinformática está voltado, atualmente, para genômica, principalmente no seqüenciamento de genomas e na busca por seu entendimento. Deste modo, a quantidade de profissionais ligados à computação dispostos a desenvolver pesquisas com as demais áreas da biologia é bastante escassa. Em especial, a imunologia recentemente tem despertado interesse dos pesquisadores de inteligência artificial, aprendizado de máquina e áreas correlatas, por ser uma fonte bastante rica de inspiração para criação de novos paradigmas de algoritmos inteligentes. Por exemplo, na referência [1] são citados diversos trabalhos relacionados com sistemas imunológicos

artificiais. Ainda assim, faltam pesquisadores da computação que estejam interessados em auxiliar os imunólogos a procurar novas teorias e a corrigir e acrescentar às antigas, sem as quais as fontes de inspiração rarearam.

Sobre essas fontes de inspiração, pode-se dizer que dentre os tópicos abordados pelos imunólogos, a dinâmica do repertório de anticorpos de um organismo permanece como uma questão em aberto na imunologia. Enquanto a maior parte dos especialistas nesta área está se voltando para análises pontuais, o estudo de um ponto de vista mais abrangente está muito pouco explorado. Certos tipos de resposta imunológica, que têm comportamento mais simples, estão relativamente bem estudados. Porém perturbações que geram respostas aparentemente caóticas ainda não possuem explicação evidente.

No presente trabalho, foram estudados dois tipos específicos de dados obtidos de experimentos biológicos, cuja extração de informação é motivo de intensa pesquisa na área de imunologia. O primeiro conjunto de dados foi gerado a partir de experimentos que procuram esclarecer como o sistema imunológico se comporta perante certas perturbações. Utilizando uma análise global do sistema imunológico, conhecida como Imunoblot, especialistas em imunologia analisaram o repertório de anticorpos de camundongos, caracterizando sua reatividade frente à centenas/milhares de antígenos [2]. No artigo da referência [3], foi feita uma análise da reatividade com antígenos autólogos, demonstrando que o repertório de anticorpos naturais é capaz de regenerar grande parte de seu formato original, após destruição significativa de linfócitos, indicando que a seleção do repertório autoreativo é um processo biológico robusto. No presente trabalho analisamos os dados obtidos em [3], utilizando: análise estatística básica, *k-means*, *fuzzy c-means*, mapas auto-organizáveis, e extração de regras de associação *fuzzy*. Dentre estes, destacam-se pela aceitação da equipe de biólogos os histogramas e os algoritmos de agrupamento *k-means*, *fuzzy c-means*, e mapas auto-organizáveis. Os resultados obtidos com estes três algoritmos de agrupamento foram muito semelhantes. Isto indica uma probabilidade alta de que os grupos/padrões encontrados nos dados não sejam fruto do acaso. Além disso, como os resultados dos três algoritmos são semelhantes, o especialista pode aplicar aquele que lhe for melhor interpretável e fácil de utilizar. Note ainda que o

fato de no uso de mapas auto-organizáveis não haver a necessidade de determinar a priori a quantidade de agrupamentos procurados, foi interpretado pelos Biólogos como uma vantagem, pois seria uma análise mais isenta de tendências.

O segundo tipo de dado vem de experimentos que procuram estipular a diversidade de linfócitos presente em determinados organismos. A técnica descrita em [4, 5] indica o uso de tecnologia de *microarrays* para testar, de uma só vez, uma população de receptores de linfócitos. A idéia é separar os RNAs de receptores específicos dos linfócitos e confrontá-los com um *GeneChip*. A diversidade então será estipulada pela quantidade de reações detectadas dentre as mais de 400.000 existentes em cada *GeneChip*. Após a conclusão de várias baterias de testes, os dados foram disponibilizados para tratamento computacional. Para este tipo de dado, os imunólogos estão interessados em medir o erro associado à estas estimativas de diversidade. Esta tarefa foi realizada em 2 partes. Na primeira parte, foi realizado um estudo estatístico dos dados disponíveis, que revelou características destes dados que dificultam comparações entre resultados de cada GeneChip. Na referência [4], foi indicado que existe uma diferença estatística significativa entre baterias de testes. No presente trabalho, investigamos as diferenças dentro de cada bateria de teste. Na segunda parte, foi criado um modelo computacional que é capaz de simular o comportamento dos dados originais, usando como medida a análise estatística feita na primeira parte. Em especial, nos concentramos em uma parte do conjunto de dados para calibrar os parâmetros do modelo e fazer testes. Para adaptar este modelo a outros conjuntos de dados, basta mudar alguns parâmetros. Se a equipe de biólogos associar os parâmetros do modelo com características bio-físicas das reações testadas, espera-se que este modelo seja útil para sugestão do que precisa ser tratado de forma diferente nos próximos experimentos.

Note que, embora a natureza dos dados trabalhados seja diferente, ambas pesquisas estão interessadas em análises globais do Sistema Imunológico, mais especificamente, em estudar a diversidade de repertórios de linfócitos de organismos. Acredita-se que a diversidade do repertório, mais do que a quantidade de linfócitos, seja crítica para montar a defesa contra microorganismos e até contra tumores.

Podemos resumir as contribuições deste trabalho em:

- Criação de uma metodologia para tratamento e análise de dados de imunologia, obtidos através de experimentos biológicos, mais especificamente, dados relacionados ao comportamento de anticorpos.
- Análise do grau de contribuição de métodos estatísticos e mais inteligentes de mineração de dados, no que diz respeito à extração de informação relevante sobre este tipo de dados.
- Criação de um modelo computacional, capaz de simular o comportamento dos experimentos biológicos originais, para reprodução de condições experimentais e posterior análise de resultados.
- Descoberta de informação relevante na análise da diversidade do repertório de linfócitos com a utilização da nossa metodologia.
- Introdução de métodos de mineração de dados mais inteligentes aos pesquisadores da área de Imunologia, facilitando a pesquisa interdisciplinar e interação entre pesquisadores de diferentes áreas.

Os Capítulos seguintes estão organizados da seguinte forma: Nos Capítulos 2 e 3, apresentamos conceitos fundamentais para a compreensão dos capítulos subseqüentes. No Capítulo 2, são apresentados principais conceitos em Biologia, com foco em Imunologia, explicando a técnica de Imunoblots e *Microarrays*, e descrevendo experimentos realizados com estas técnicas. No Capítulo 3, introduzimos os conceitos e métodos estatísticos utilizados nesta tese, assim como outros métodos inteligentes de mineração de dados. Apresentamos também a metodologia de tratamento de dados proposta nesta tese. Nos Capítulos 4 e 5, apresentamos o tratamento dos dois conjuntos de dados utilizados neste trabalho e os resultados obtidos aplicando os métodos estudados. Também neste capítulo, fazemos a análise dos resultados e a discussão sobre seu significado e relevância biológicos. No Capítulo 6, apresentamos as conclusões e perspectivas de trabalhos futuros.

Notas sobre tradução e terminologia

Alguns termos que aparecem no texto dessa tese, tais como *probe*, *perfect match*, e *hits*, estão em inglês pois é difícil encontrar uma tradução adequada que expresse o mesmo significado do termo em português. Outros termos, como *microarray*, *target* e *k-means*, tem tradução para o português, mas o termo em inglês é o mais conhecido na comunidade científica. Já *Standard*, *Sample* e Curva *Standard* optamos por deixar em inglês pois são os termos utilizados pelos criadores da técnica experimental descrita na seção 2.3.3.

Alguns termos foram traduzidos para o português, mas os acrônimos utilizados, tais como SOM, NAbs, TCR, etc. são os que correspondem aos termos em inglês. Esses acrônimos foram mantidos nessa forma por serem amplamente utilizados pela comunidade. A utilização desses termos e acrônimos não representa de forma alguma desconsideração à língua portuguesa, e somente foram utilizados pelos motivos acima apresentados.

Capítulo 2

Análises globais do Sistema Imunológico

Neste capítulo, serão apresentados alguns conceitos básicos de Biologia e Imunologia, além de detalhamentos sobre os métodos biológicos e computacionais realizados para obtenção dos dois tipos de dados que serão estudados neste tese. Na seção 2.2, são mostrados os objetivos de uso e a forma de obtenção dos dados de um Imunoblot, bem como um experimento específico feito com essa metodologia. Na seção 2.3, são abordados conceitos básicos de genética, e uma breve explicação de como funciona um *microarray*, além da descrição do experimento que utiliza *microarrays* para analisar o Sistema Imunológico.

2.1 Conceitos básicos sobre a Imunologia

A Imunologia é uma área de estudo bastante ampla, com diversas teorias, muitas vezes conflitantes entre si. À medida que as técnicas e equipamentos permitem experimentos cada vez mais precisos, os pesquisadores desta área procuram confirmar estas teorias, ou mesmo refutá-las à luz de novas informações. Além disso, há duas grandes abordagens nos experimentos atuais. Alguns cientistas visam conhecer os *componentes* do sistema imunológico, *um a um* (e o têm feito com impressionante precisão), usando técnicas como, por exemplo, seqüenciamento de proteínas que compõem a parte variável

dos anticorpos. Enquanto isso, outros procuram formular *teorias sistêmicas*, que visam explicar o comportamento do sistema como um todo, usando ferramentas como simulações computacionais. Os experimentos que são analisados neste trabalho seguem a segunda abordagem.

Em livros como em [6], conceitos de Imunologia estão descritos minuciosamente; no entanto, no presente trabalho será mostrado apenas o mínimo necessário para entendimento dos dados utilizados para análise computacional, e com o alerta de que a descrição aqui será bastante superficial e simplificada.

No sistema imunológico dos mamíferos encontra-se uma grande variedade de *anticorpos*, que são os responsáveis por proteger o corpo de elementos estranhos, os *antígenos*[1]. Toda e qualquer molécula que estimule o sistema imunológico é considerada um antígeno, por exemplo, substâncias solúveis, ou moléculas presentes em vegetais ou na superfície de organismos como bactérias e vírus. Estas moléculas são chamadas de *moléculas heterólogas*, ou seja, que não pertencem ao organismo em questão. Em alguns casos, moléculas do próprio organismo (chamadas de *moléculas self* ou *moléculas autólogas*) também podem atuar como antígenos.

No período peri-natal, ou logo após o nascimento (depende da espécie), os anticorpos vêm de células chamadas de *linfócitos B,* vindos do fígado. Posteriormente, a responsabilidade pela produção de anticorpos é dividida pela população de linfócitos B presentes no baço e na medula óssea. O *repertório de anticorpos* de um sistema imunológico representa o cadastro de anticorpos estruturalmente diferentes que conseqüentemente define o conjunto de antígenos suscetíveis de serem reconhecidos. Acredita-se que o repertório de anticorpos sofre mudanças drásticas quando imaturo, e que mesmo após sua maturidade ainda preserva a possibilidade de aprender a reconhecer novos antígenos (porém com menos plasticidade). Esta possibilidade de aprendizado na maturidade deve-se ao fato de que, durante toda a vida do indivíduo, novos linfócitos B são produzidos e liberados no corpo. Assim, estão sendo inseridos novos anticorpos que

[1] Os anticorpos e os antígenos se sub-dividem em diversas categorias e tipos diferentes de moléculas. Conforme dito, não entraremos nestas sub-divisões por questão de simplificação. Para saber mais, veja em [6].

não são específicos para nenhum antígeno já cadastrado no repertório, mas que podem ser usados a qualquer momento que o sistema imunológico seja acionado, por exemplo, na luta contra um novo antígeno com que se tenha contato.

Além dos anticorpos produzidos por moléculas heterólogas, há também os chamados *anticorpos naturais* (*Natural antibodies* - NAbs), que estão presentes nos organismos independente de ter havido ou não contato com antígenos externos [7]. Acredita-se que estes resultem da estimulação de antígenos *self* [8], principalmente pela constatação de que os anticorpos naturais reagem preferencialmente com moléculas *self* [9, 10, 11]. Embora o papel fisiológico das NAbs autoreativas ainda não esteja totalmente elucidado, suas funções regulatórias na tolerância, autoimunidade e na defesa contra infecções foram evidenciadas em diferentes modelos experimentais [12, 13, 14, 15].

Além dos linfócitos B, existem também *linfócitos T*, que não produzem anticorpos. Os linfócitos T se originam na medula óssea, e migram para o timo. Na superfície dos linfócitos T existem estruturas chamadas de *receptores de células T* (TCRs), que possuem extrema capacidade de variar sua composição. Quando estão no timo, são testadas várias combinações de TCRs, e só os que estão aptos a atuar no organismo são liberados. Existem vários tipos de linfócitos T, e cada tipo atua de forma diferente no momento que o sistema imunológico é acionado. Por causa das TCRs, os linfócitos T também são capazes de mapear os antígenos apresentados ao organismo. Portanto, mais à frente no texto, quando houver referência à diversidade do repertório de linfócitos de um organismo, salvo indicação contrária, estamos nos referindo aos dois tipos de linfócitos, B e T.

2.2 Imunoblot

Estudos como em [16, 17, 18] procuraram compreender melhor a formação e seleção do repertório de NAbs, e quais seriam seus papéis no sistema imunológico. Nestes estudos, soros humanos e de camundongos contendo NAbs foram confrontados com centenas/milhares de moléculas autólogas e heterólogas, e demonstraram reatividades preferenciais com certos antígenos. Em especial, para os experimentos realizados em

[16] e [2], foi criada uma técnica chamada de Imunoblot. No trabalho realizado em [2], foi demonstrado que entre indivíduos normais há uma homogeneidade significativa nos repertórios de NAbs; entretanto, os perfis de reatividade são distintos entre espécies diferentes (comparando humanos e camundongos, e espécies diferentes de camundongos). Ou seja, há uma *assinatura* (perfil característico de reatividade do repertório) que possibilita diferenciar cada espécie.

A técnica utilizada em [16], inclusive o Imunoblot, consiste, resumidamente, em (veja figura 2.1):

1. Separação do extrato protéico (extratos totais autólogos ou heterólogos) por eletroforese em gel de poliacrilamida (SDS-PAGE) sem o uso de canaletas;
2. Transferência das proteínas para a membrana de nitrocelulose;
3. Incubação dos soros no sistema Cassete-MiniBlot;
4. Revelação das reatividades utilizando um segundo anticorpo ligado à fosfatase alcalina;
5. Digitalização dos perfis de reatividade de cada membrana, utilizando um scanner de alta definição, e quantificação por densitometria;
6. Incubação da membrana em um corante protéico (por exemplo, ouro coloidal);
7. Digitalização dos perfis de reatividade marcados com o corante protéico, e quantificação densitométrica dos perfis protéicos nos espaços entre as canaletas;
8. Tratamento computacional dos dados obtidos.

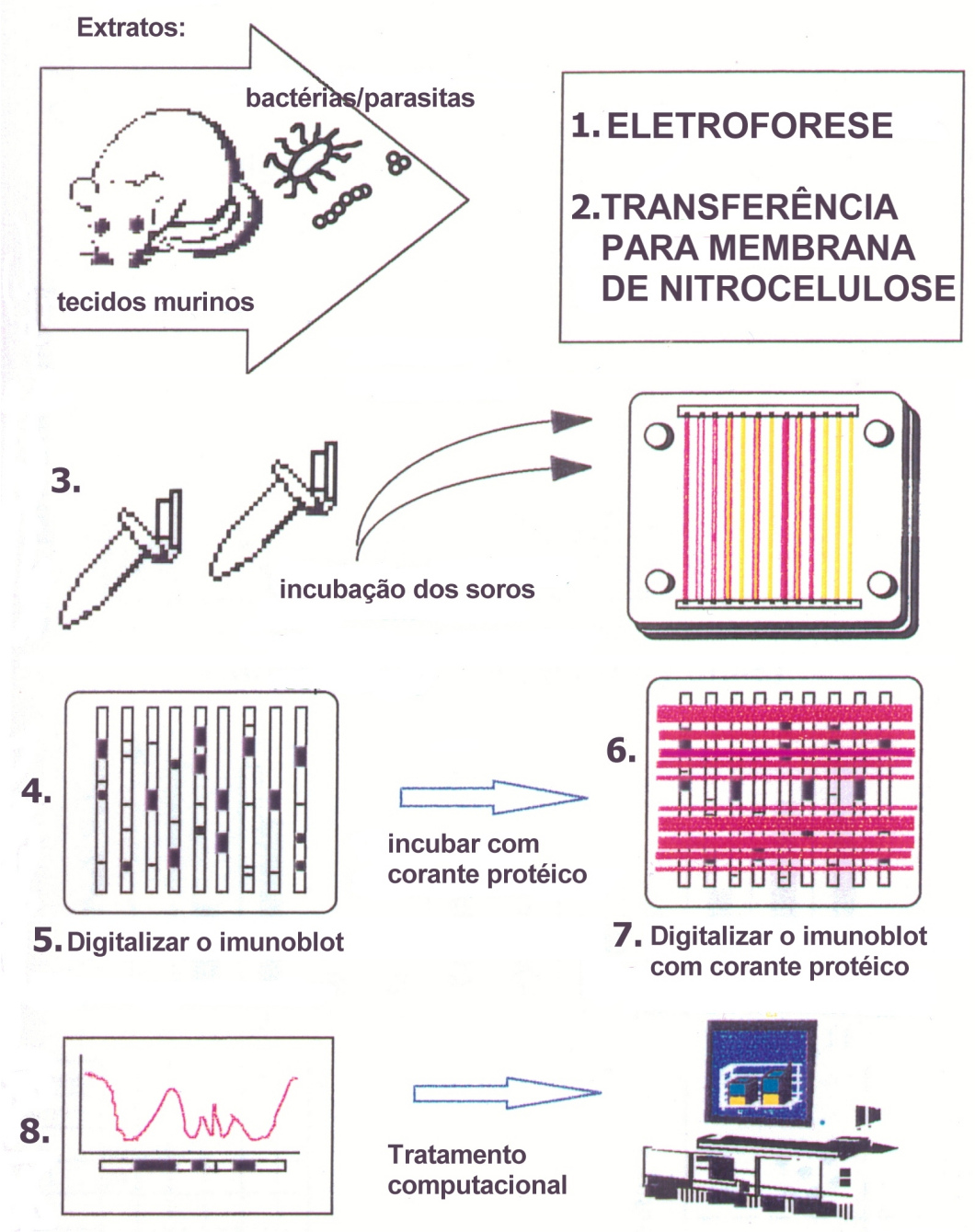


Figura 2.1: Técnica para utilização de Imunoblot.

Nos itens a seguir, serão descritos em mais detalhes os passos 5, 7 e 8 mencionados acima:

- Sobre a aquisição de imagens: as imagens dos Imunoblots foram digitalizadas, gerando os arquivos correspondentes aos perfis de imuno-reatividade e protéicos (antes e após coloração protéica, respectivamente). A quantificação densitométrica

dos perfis foi realizada com o programa NIH *Image* [19], na imagem original, não editada, dos Imunoblots e armazenada em arquivos para posterior análise e processamento dos dados. Esta quantificação densitométrica consiste em dividir o domínio em 1000 ou 1200 cortes horizontais e atribuir valores de 0 a 255 (correspondentes ao tom de cinza) para cada um dos cortes, para cada perfil (na vertical). O resultado deste procedimento é uma matriz cujas colunas são os perfis, e cujas linhas contêm os valores da *densidade ótica* para cada corte.

- Sobre o re-escalonamento dos Imunoblots: a fim de corrigir as eventuais distorções da migração eletroforética e permitir a comparação dos perfis de reatividade, foram desenvolvidos comandos especiais (macros) no programa IGOR [20]. Resumidamente, o ajuste foi feito como se segue:

    - É escolhido um perfil protéico na região central da membrana para servir como perfil de referência;
    - É escolhido neste perfil um número adequado de picos e vales de fácil reconhecimento;
    - Identifica-se os picos e vales homólogos no perfil protéico adjacente;
    - Este perfil em que foram identificados os picos/vales passa a servir como referência para o próximo, até que todos os perfis protéicos apresentem o mesmo número de marcações (picos/vales) estabelecidos no primeiro perfil de referência;
    - Passa-se a correção e superposição dos picos/vales identificados, usando-se um algoritmo de correção linear por partes, conforme descrito em [2].
    - Findo esse processamento, pode-se comparar uma determinada banda de reatividade entre amostras sabendo que estamos comparando áreas de imuno-reatividade com proteínas de mesmo padrão de migração. Essa comparação pode ser realizada internamente num mesmo Imunoblot, ou entre Imunoblots

distintos que tenham sido feitos com mesmo extrato e que usem o mesmo perfil de referência.

- Caso seja interessante para um determinado estudo, esta análise de re-escalonamento gera informações suficientes para realizar uma redução na dimensão dos dados, ou seja, diminuir a quantidade de cortes horizontais, sem perder informação. Nesta etapa da análise quantitativa das reatividades, é necessário em primeiro lugar separar os perfis densitométricos em seções de reatividade, de acordo com os picos (reatividades) obtidos para cada extrato antigênico. Observe que, desta forma, as faixas horizontais não precisam mais ter o mesmo tamanho. Em seguida pode-se calcular a área sob cada seção e desta forma, cada perfil de reatividade pode ser representado por uma série de valores, resultando em uma matriz de dimensão correspondente ao número de suas reatividades. Isso faz com que uma matriz de 1200 linhas possa ser representada por uma de 40 a 60 linhas.

Na figura 2.2, pode ser visto um exemplo de resultado obtido usando o método descrito acima. O experimento citado nesta figura será explicado na seção a seguir.

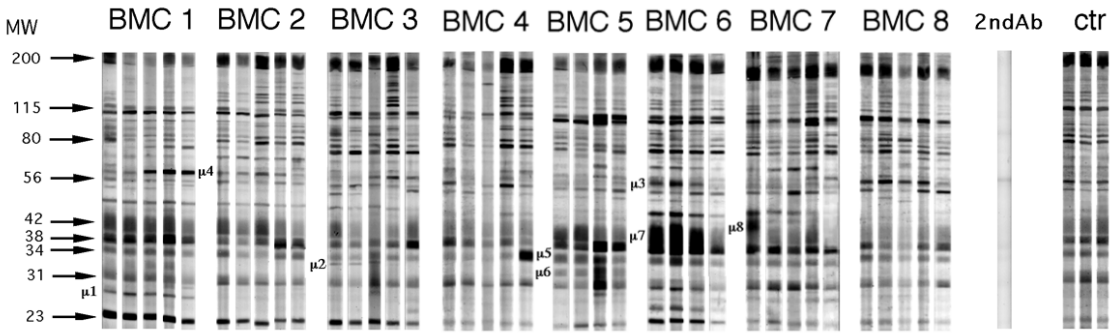


Figura 2.2: Experimento de regeneração. Retirado de [3].

2.2.1 Experimento de regeneração

O *experimento de regeneração* [3] foi criado para analisar a estabilidade do repertório

de NAbs e sua homeostase[2]. Para tal, camundongos foram expostos a uma radiação sub-letal que destrói as células que originam o seu repertório de anticorpos. Logo após, é inserido um extrato gerado a partir de medula óssea ou de fígado fetal de outros camundongos. Os camundongos que receberam esses extratos passam a ser chamados de quimera de medula óssea ou quimera de fígado fetal[3]. Destas quimeras, foram extraídas amostras de várias partes do seu organismo, cujos extratos serão usados para observar a reatividade dos anticorpos. Os resultados descritos em [3] são de extrato de fígado e de músculo, de ambas as quimeras. Note que todos os camundongos utilizados no experimento são isogênicos, ou seja, são geneticamente idênticos. O objetivo é testar se o sistema imunológico preserva sua assinatura mesmo após depleção[4]. Os resultados deste experimento mostraram que os repertórios dos camundongos irradiados regeneram-se de forma muito semelhante ao que era antes.

O experimento descrito acima foi feito sistematicamente para vários camundongos, sendo que cada um dos repertórios era ligeiramente diferente entre os indivíduos antes da irradiação[5]. Em suas recuperações, cada um teve seu repertório regenerado, conforme dito antes, de forma muito semelhante ao que era antes para cada um deles. Contudo, o processo os tornou um pouco mais semelhantes entre si. Isto indica que o comportamento do repertório, embora muito complexo, não é ao acaso: parecem existir regras em um nível mais alto que indicam como a "rede" de anticorpos deve ser montada.

A figura 2.2 mostra um exemplo de dados coletados desta experiência: **BMC1**, **BMC2,** até **BMC8** são as indicações de camundongo 1, 2, até 8 de quimera de medula óssea[6]; A faixa marcada como **2ndAb** seria o correspondente a um determinado ruído de fundo (*background*[7]); e o **ctr** é o camundongo controle. Cada coluna corresponde à reatividade dos anticorpos (quanto mais escuro, maior a reatividade registrada) para um

[2]Equilíbrio dinâmico em que algo se encontra.

[3]Quimeras são animais que reconstituem populações a partir de células provenientes de outro animal.

[4]Diminuição da quantidade dos humores do organismo.

[5]Conforme dito anteriormente, os repertórios eram suficientemente parecidos para indicar que são da mesma raça; no entanto, apesar de serem indivíduos isogênicos, os repertórios apresentavam algumas diferenças entre indivíduos.

[6]BMC vem de *Bone Marrow Chimera.*

[7]Neste caso, o *background* é um valor de fundo, associado à reatividade não específica. No cálculo da densidade ótica, esse valor do *background* pode ser subtraído das demais faixas, para obter valores mais significativos de reatividade.

camundongo em um tempo. Ao lado esquerdo, **MW** indica o provável peso molecular do grupo de peptídeos daquela "linha". Para cada um dos camundongos, na figura, são exibidos 5 tempos diferentes: o dia (-1) (um dia antes da irradiação), e os dias 7, 15, 30 e 60 após a irradiação. Em alguns indivíduos, como os indicados por **BCM5** e **BCM6**, não há os 5 tempos (para ambos não foi medido o dia 15). O controle mostra os dias (-1), 30 e 60.

Conforme dito anteriormente, para análise computacional as figuras digitalizadas são divididas em 1200 cortes horizontais, e em cada pequena faixa destas está representada uma proteína diferente (ou algumas poucas proteínas; mais especificamente, um grupo de peptídeos). Para indicar o quanto cada uma destas proteínas (ou conjunto de peptídeos) está reagindo, são dados valores correspondentes à escala de cinza (densidade ótica): de 0 (mais claro, menor atividade) à 255.

Se a intenção for trabalhar com estes dados brutos, é necessário fazer mais uma manipulação: destas 1200 faixas de valores, são desconsideradas as primeiras e as últimas, pois nestas as perturbações do método utilizado influenciam nas medidas. Portanto, os dados que serão utilizados são matrizes de 600 a 650 linhas, representando a parte mais estável do experimento. As colunas irão indicar os indivíduos com seus respectivos tempos: para os dados de quimera de medula óssea, foram testados 9 indivíduos, sendo que (conforme dito acima) em dois deles não foi medido o dia 15, portanto serão 43 colunas; e para a quimera de fígado fetal, foram testados 6 indivíduos, nos dias (-1), 15, 30 e 60, portanto são 24 colunas.

Para que seja possível comparar os perfis de reatividade representados pelas colunas, antes de selecionar as faixas de valores a serem usadas, foi aplicada a primeira etapa da análise de re-escalonamento, a fim de corrigir as eventuais distorções inerentes ao método.

Já se a intenção for trabalhar com os valores médios das seções de reatividade, a segunda etapa de análise de re-escalonamento, de redução da quantidade de dimensões, pode ser aplicada. Na referência [3], esta etapa do re-escalonamento foi feita somente para uma parte dos dados, das quimeras de medula óssea. Com isso, os dados passaram a ser representados por 40 linhas, representando as faixas de valores para o extrato de

fígado, e 38 linhas, para o extrato de músculo.

Resumidamente, pode ser visto na tabela 2.1 a relação de quimeras e extratos com os quais estaremos lidando no decorrer do trabalho, e o tamanho de cada matriz.

	Quimeras de Medula Óssea	Quimeras de Fígado Fetal
Extrato de Fígado - Dados Brutos	Matriz 624x43	Matriz 600x24
Extrato de Músculo - Dados Brutos	Matriz 650x43	Matriz 600x24
Extrato de Fígado - Médias das Reatividades	Matriz 40x43	—
Extrato de Músculo - Médias das Reatividades	Matriz 38x43	—

Tabela 2.1: Experimento de Regeneração - Tamanho das Matrizes

2.3 *Microarrays*

Como mencionado anteriormente, as pesquisas em biologia que buscam conhecimentos cada vez mais específicos se beneficiam com o aprimoramento da tecnologia. Hoje em dia é possível saber quais genes produzem quais proteínas, e como uma proteína interage com outra. Mas, sabe-se que nos organismos, a regulação desta produção dos genes e da interação das proteínas depende de um contexto amplo, que é muito difícil de ser analisado. Para este tipo de análise sistêmica, a tecnologia de *microarrays* é bastante útil, pois permite ver a interação de milhares de genes de uma só vez. Para entender o que está sendo testado no *microarray*, serão esclarecidos na próxima seção alguns conceitos da genética, inclusive o de hibridização, que é a base para esta tecnologia.

2.3.1 Processo de hibridização

O DNA (ácido desoxirribonucleico) é composto por *nucleotídeos* (também chamados de *bases*), que são denominados *Adenina*, *Timina*, *Citosina* e *Guanina*, e são representados respectivamente pelas letras **A**, **T**, **C** e **G**. Os nucleotídeos encontram-se ligados linearmente, formando uma cadeia sem ramificações (ou filamento). O DNA é composto por duas destas cadeias, formando uma estrutura descrita como uma dupla hélice. Uma *complementaridade de pares de base* é respeitada, ou seja, devido às características físico-

químicas dos nucleotídeos, a ligação entre as duas cadeias acontece entre um nucleotídeo A de uma cadeia com o T da outra, ou entre um nucleotídeo C de uma e um G da outra cadeia.

A seqüência de nucleotídeos do DNA guarda toda a informação genética dos organismos, e o total desta seqüência é chamado de genoma. As instruções contidas no DNA são "executadas" pelas *proteínas*, e as regiões do DNA que codificam as proteínas são chamadas de *genes*.

Para transformar genes em proteínas, é construída uma seqüência de RNA (ácido ribonucleico) correspondente àquela parte do DNA, repeitando a complementaridade de bases. Note que o RNA possui *Uracila* (U) em vez de Timina. Assim, os pares de base no RNA são C com G e A com U. Outra característica que difere o RNA do DNA é que o RNA possui apenas um filamento, ficando disponível para se *ligar* a outros filamentos de DNA (que não estejam previamente ligados) e RNA. Esta ligação entre filamentos chama-se *hibridização* ou *emparelhamento de bases.*

É possível quebrar um genoma e genes conhecidos em pedaços menores e colocá-los em um meio onde possam se "encontrar". Se um destes pedaços tiver a seqüência complementar de outra, ocorrerá a hibridização. Ou seja, seqüências complementares se atraem, como ímãs, e seqüências não complementares tendem a não se ligar.

2.3.2 O *microarray GeneChip*

O *microarray* é composto por centenas de milhares de pequenas seqüências de nucleotídeos presas a um meio sólido, por exemplo, vidro. Estas seqüências conhecidas são chamadas de *probes*, e juntas representam conjuntos de genes. As seqüências que se deseja testar, conhecidas ou não, chamam-se de *target*. As *targets* são marcadas e contrastadas com o *microarray*. Após um certo tempo o *microarray* é lavado, então somente onde ocorreu a hibridização é que as *targets* ficam. Como elas foram previamente marcadas, é possível procurar no *microarray* em quais partes houve hibridização. Esse método permite identificar de uma só vez vários genes que estariam ativos em uma determinada situação, por exemplo, genes associados a doenças.

a) 

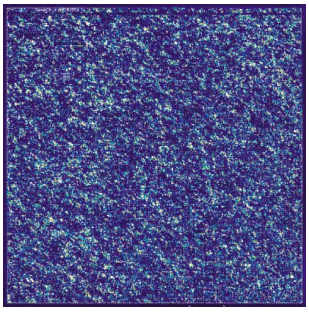
b) 

Figura 2.3: a) *GeneChip* da Affymetrics, modelo: *Human Genome* U133 *Plus* 2.0 *Array*. b) Resultados de um experimento mostrando a expressão de milhares de genes em um único *GeneChip*. A cor mais clara indica maior expressão. Imagens obtidas em Affymetrix [21].

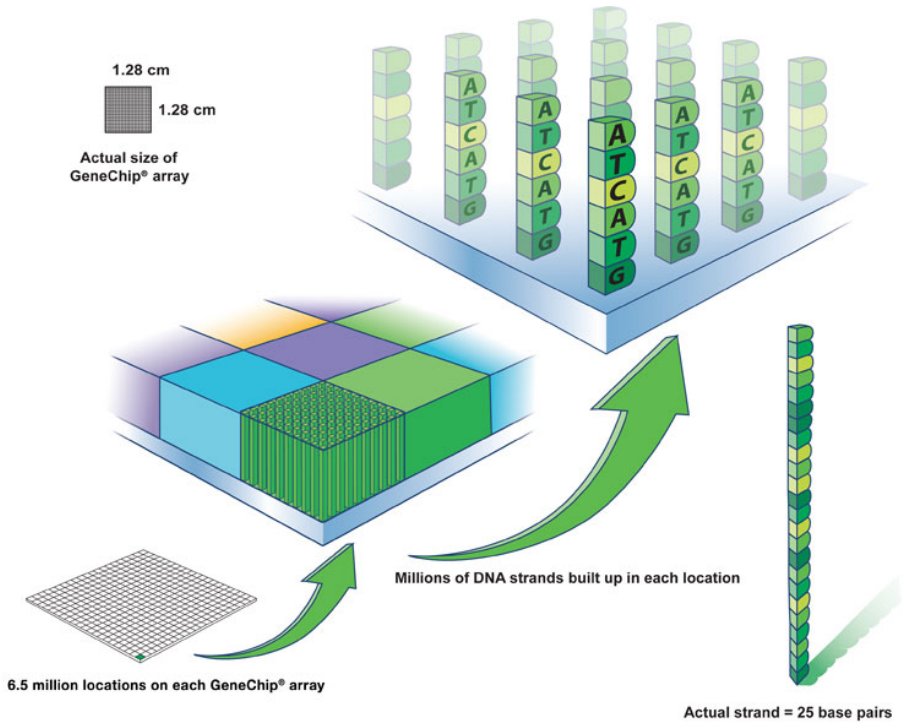


Figura 2.4: Representação das *probes* em um *GeneChip*. Imagem obtida em Affymetrix [21].

Existem vários tipos e marcas comerciais de *microarrays*. Neste trabalho, iremos nos concentrar somente no que foi utilizado pela equipe de imunólogos que fez o experimento a ser analisado: o *GeneChip* HGU95B, da empresa Affymetrics[21] (veja figuras 2.3 e 2.4). Os *microarrays* que esta empresa desenvolve usam seqüências de tamanho 25 para

as *probes*, e usam pares de *probes* representando o *perfect match* (PM) e o *mismatch* (MM). O MM é uma seqüência idêntica ao PM, mas a base do meio (a 13ª) é mudada. Com isso, é possível controlar a detecção de hibridização não específica, pois se uma *target* se liga tanto ao PM quanto ao MM, esta *target* não é específica para aquele gene. Este par de PM e MM é chamado de *probe set*.

Este *GeneChip* HGU95B foi projetado para consultar *expression sequence tags* (ESTs) que representam genes humanos. Cada EST é representado por 16 pares de *probe sets*. Neste *GeneChip*, 1,5% das *probes* são usadas para controle de qualidade da performance da hibridização, restando mais de 200.000 pares de *probes* (mais de 12.000 genes) para serem consultados.

2.3.3 Experimento de estimação da diversidade de repertórios

O grupo de Ogle *et al.* [4, 5] propôs o uso de *microarrays* para estimar a diversidade de um conjunto de linfócitos. A técnica descrita consiste na execução dos seguintes passos (veja figura 2.5):

1. **Preparação dos *Standards*:** os *Standards* são os conjuntos de amostras cujas diversidades são conhecidas. Para sua criação, foi criada uma seqüência aleatória de DNA de 18 nucleotídeos. Sobre esta seqüência, foram escolhidos pontos (`N`) em localizações específicas para serem trocados por qualquer um dos 4 nucleotídeos. Então, por exemplo, para gerar uma amostra com aproximadamente 10^6 seqüências diferentes, a seqüência original precisa ter 10 pontos de troca (4^{10} = 1,048,576). Veja a seqüência original e as 4 diversidades escolhidas para os *Standards* na tabela 2.2.

   As seqüências são biotiniladas[8] durante sua síntese, rotulando o último nucleotídeo. São separados 10μg de cada preparação (de cada diversidade) e estes são contrastados contra vários *GeneChips*, um para cada diversidade. Analisando cada *GeneChip*, o *número de hits*, ou seja, a quantidade de posições que mostrou

[8] Uma forma de marcar a seqüência para que esta possa ser detectada posteriormente.

Diversidade	Seqüência
$4^0 = 1$	`cagccaagtctgggacca`
$4^5 = 1024$	`caNcNaagtNtggNaNca`
$4^{10} \sim 10^6$	`NNgNcNaNNNtgggNcNN`
$4^{15} \sim 10^9$	`NNNNcaNNNNNgNNNNNN`

Tabela 2.2: As 4 diversidades escolhidas para os *Standards.* Seqüência original para criação dos *Standards* na diversidade 1, e posições dos pontos de troca (`N`) nas demais diversidades.

reação acima de um determinado ruído de fundo (*background*[9]) é somada. Este valor total de reações é usado para desenhar uma curva chamada de *Curva Standard*, que será então utilizada para estimar a diversidade de outras amostras.

2. **Preparação dos *Samples*:** os *Samples* são amostras fisiológicas cujas diversidades são desconhecidas. Para sua preparação, os primeiros passos são escolher o tipo de amostra a ser testado, isolar e selecionar os linfócitos, e separar o RNA de receptores específicos destes linfócitos. Ou seja, são separadas as ”partes” que realmente diferenciam os linfócitos. Estas seqüências também são biotiniladas, e são fragmentadas em pedaços que variam de tamanho entre 50 e 200 pares de bases. Da mesma forma que os *Standards*, 10μg são separados e contrastados contra o mesmo tipo de *GeneChip*, o *human U95B*. O número de *hits* deste resultado também é contado.

3. **Interpolação dos resultados:** a Curva *Standard* é criada relacionando o número de *hits* com cada diversidade criada para ser *Standard*, e interpolando linearmente estes resultados de diversidades conhecidas. Observe que a interpolação linear só é possível se os dados estiverem em escala logarítmica. Obtendo o número de *hits* do resultado de um *Sample*, a diversidade é estimada pelo ponto em que a Curva *Standard* é interceptada.

Para cada conjunto de *Samples* gerados, foram gerados também novos conjuntos de *Standards*, pois conforme Ogle *et al.* [4], ao criar uma série de diferentes *Standards*, a equipe de biólogos percebeu que, apesar da inclinação das Curvas *Standard* ser

[9]Neste caso, o *background* é um valor de fundo, comum a todos os *GeneChips* que são testados juntos em um determinado experimento. O *background* está associado à hibridização não específica.

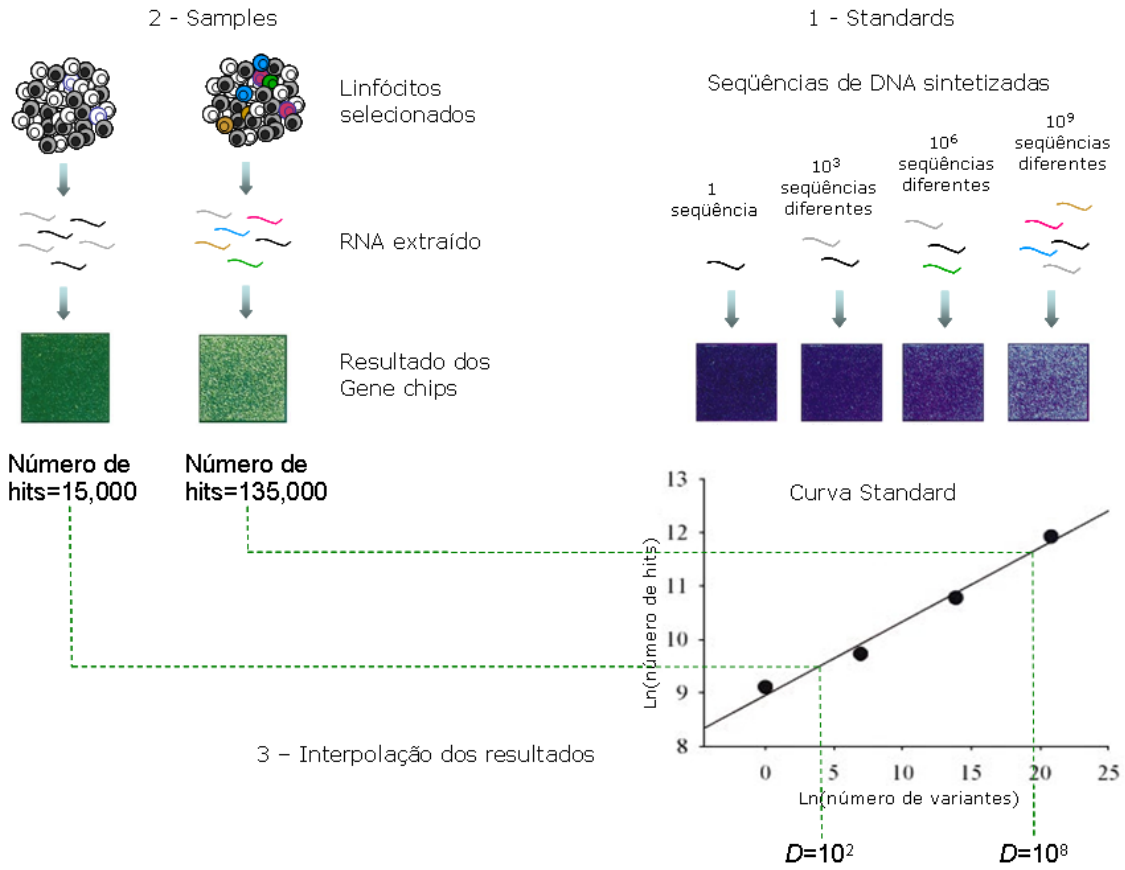


Figura 2.5: Técnica utilizada em Ogle *et al* [4] para criação e uso da Curva *Standard*.

(estatisticamente) a mesma, o ponto de interpolação, no eixo que representa a intensidade da hibridização, mudava de experimento para experimento. Portanto eles concluíram que para cada conjunto de *Samples*, era necessário fazer um novo conjunto de *Standards*. No Capítulo 5 comentaremos mais a respeito desta diferença.

2.4 Contexto desta Tese

Neste capítulo, foram expostos as técnicas e os objetivos de dois experimentos biológicos. Os dois grupos de pesquisa, tanto o que estava trabalhando com Imunoblots quanto o que utilizou *microarrays*, perceberam que seus dados poderiam sofrer uma análise mais aprofundada, pois esta análise computacional complementar poderia trazer novas informações sobre os resultados obtidos. Foi neste contexto que esta tese foi desenvolvida.

No próximo capítulo serão apresentados outros métodos computacionais diferentes dos citados neste capítulo, que serão utilizados para investigar em mais detalhes os

resultados obtidos nos experimentos descritos, procurando extrair informações relevantes dos dados biológicos. Estes métodos computacionais foram escolhidos principalmente visando a facilidade na interpretação dos resultados pelas equipes de biólogos. Portanto, alguns métodos que são extensamente utilizados em Bioinformática, e potencialmente muito eficazes, como por exemplo Redes Neuronais *Backpropagation*, não foram utilizados nesta tese pela dificuldade na interpretação dos seus resultados. Em contrapartida, foram utilizados outros métodos, alguns até menos elaborados, mas de interpretação mais direta, especialmente os que possibilitam uma visualização gráfica dos resultados. Durante a elaboração desta tese, as equipes de biólogos foram consultadas a respeito da facilidade de interpretação dos métodos selecionados.

Capítulo 3

Mineração de dados

O termo *mineração de dados* engloba uma série de métodos, vindos de áreas como a estatística, probabilidade, inteligência artificial, aprendizado de máquina, bancos de dados, etc. O principal objetivo da mineração é procurar informações que estão ocultas em bancos de dados, ou seja, que não são óbvias mesmo para o criador da base de dados, nem para um observador especialista na área dos dados gerados, porque na maioria das vezes a base é grande demais para ser analisada manualmente ou visualmente. O termo *KDD* também é utilizado para designar este tipo de procura. As informações adquiridas podem ser usadas para construir modelos, para prever próximas ocorrências, para dar suporte a decisões, ou simplesmente para entender melhor os dados. Para saber mais sobre mineração de dados e KDD há diversas fontes, por exemplo artigos como os das referências [22, 23], livros como o da referência [24], e endereços eletrônicos como o da referência [25].

A KDD tem sido utilizada em aplicações tão distintas quanto pesquisas sobre a Internet (por exemplo, em [26] e [27]), bases de clientes de grandes lojas e prestadoras de serviços (como lojas de departamentos, supermercados, e empresas de telefonia), dados sobre solo de certas regiões para empresas de extração de petróleo, e análise financeira de clientes de bancos. Seja qual for a base de dados a ser trabalhada, é muito importante que os especialistas que geram esses dados possam acompanhar o processo de mineração, principalmente em dois pontos: esclarecendo sobre o que são os dados, o que eles significam, e como eles foram gerados; e verificando se as informações descobertas fazem

sentido.

Para cada tipo de dado diferente, há métodos que dão melhores resultados. Portanto, seja de qual área for, se o método consegue extrair informações interessantes e úteis, este é bem aceito pelos especialistas. Na maioria dos casos, não é possível saber *a priori* se um método vai ter sucesso ou não, sendo necessário testá-lo. Algumas características dos dados a serem analisados podem indicar o uso de uma ferramenta ou de outra, e eliminar algumas possibilidades (isto será explorado mais adiante). Observe que, desta forma, minerar dados é uma tarefa muito ampla. Para tornar a KDD mais específica, há uma estrutura padrão de passos a serem seguidos, adaptada de [23], e exibida na figura 3.1. Esta estrutura é apenas uma sugestão, e pode-se navegar iterativamente pelos passos, voltando quando necessário. Cada um dos passos será explicado nas próximas seções.

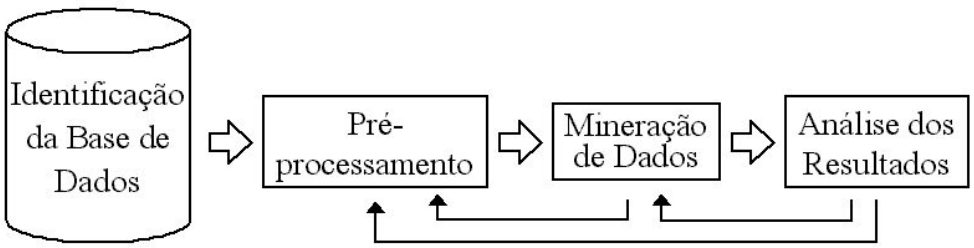


Figura 3.1: Passos do KDD

3.1 Identificação da base de dados e dos arquivos

Nesta etapa, após a aquisição dos arquivos contendo os dados a serem analisados, é essencial a parceria com os especialistas que os geraram, pois quanto mais for conhecido sobre os dados, mais facilitada será a busca das informações ocultas e relevantes. Devem ser selecionados quais dados serão úteis, e destes, quais *atributos* (características) serão analisados. Desta parte, é interessante adquirir, se possível, as seguintes informações: de que se tratam os dados; como foram gerados; e como estão organizados.

3.2 Pré-processamento

Esta é uma etapa essencial da KDD, e muitas das vezes uma das que mais consome tempo e recursos. Uma vez que os dados já foram identificados, pode-se realizar uma série de manipulações para adequá-los aos métodos e algoritmos que poderão ser escolhidos na próxima etapa. As informações das seções a seguir sobre limpeza e redução da base de dados foram retiradas de [24].

3.2.1 Limpeza dos dados

Uma das manipulações que pode ser feita na etapa de pré-processamento é a limpeza dos dados, pois há métodos de Mineração de Dados que não aceitam bases de dados incompletas. Neste caso, pode-se simplesmente apagar o *registro* (o caso) que está incompleto, ou completar o atributo faltante[1]. Para verificar se um registro pode ser apagado sem prejuízo para a análise, deve-se verificar:

- Se a base de dados é suficientemente grande, e os registros a se apagar são poucos em relação à quantidade total, de forma que ainda restem registros suficientes para as análises estatíticas;
- E se os dados estão aleatoriamente faltando, ou seja, se não houver uma causa específica que faça com que, sistematicamente, os dados de um ou alguns atributos não sejam coletados para vários registros. Se os dados faltantes não forem completamente aleatórios, apagar estes registros pode alterar alguma tendência (*bias*) dos dados.

[1] Normalmente, para as tarefas de mineração, as matrizes de dados a serem analisadas são organizados em registros nas linhas, e atributos nas colunas. Os registros são os casos, por exemplo, indivíduos diferentes, ou cada dia do experimento com cada animal, ou cada compra de cada pessoa. Os atributos são as características de cada registro que foram obtidas e/ou medidas, por exemplo, indicadores sócio-econômicos de um indivíduo, ou os valores de reatividade do perfil de um camundongo, ou quais produtos disponíveis na loja analisada. Portanto, apagar um registro é apagar um caso inteiro; e completar o atributo faltante é acrescentar na matriz apenas um valor correspondente ao atributo que os outros registros possuem, mas aquele não.

A alternativa então seria completar os dados faltantes. Há diversas técnicas para fazer esse preenchimento de dados, desde métodos estatísticos bastante simples e tradicionais como preencher com a média dos dados faltantes, ou usar regressão linear, até métodos mais modernos e elaborados como EM (*Expectation Maximization*), ou MCMC (*Markov Chain Monte Carlo*). Estes últimos são mais custosos computacionalmente, mas dão resultados bem melhores, no sentido de alterar o mínino possível a tendência dos dados. Mais informações sobre estas questões podem ser encontradas em [28]. Um exemplo de análise de dados com atributos faltantes em uma aplicação de Bioinformática é apresentado em [29]. Neste trabalho, utilizando uma base de dados do *International Breast Cancer Study Group* (<www.ibcsg.org>) contendo medidas de qualidade de vida de uma série de pacientes que tiveram câncer no seio passível de operação, os autores investigaram sobre como apagar registros com dados faltantes e como preencher estes dados faltantes pode alterar o resultado da análises estatísticas.

No caso do presente trabalho, conforme mencionado na seção 2.2.1, em dois camundongos não foi medido o dia 15 (terceiro tempo da medição). Para os métodos que exigiram dados completos, para completar a base de dados foi utilizada a média entre os valores dos tempos anterior e posterior (dias 7 e 30). Optamos por utilizar um método simples pois as informações que estão em foco são as do primeiro dia e do último dia do experimento, portanto um dado preenchido no terceiro tempo não irá alterar as análises feitas.

3.2.2 Seleção de registros e atributos

Se a base for grande demais para o poder computacional disponível, pode-se selecionar apenas alguns registros para serem analisados, com técnicas que variam de seleção aleatória até seleções mais elaboradas, como estudar estatisticamente os registros e selecioná-los de acordo com este estudo. A natureza dos dados dos dois experimentos biológicos estudados faz com que as bases de dados tenham uma quantidade grande de atributos, mas as metodologias experimentais adotadas resultaram em poucos registros para cada experimento. Portanto não houve a necessidade de aplicar técnicas de redução

de registros a esses resultados.

Quanto aos atributos, pode haver a necessidade de se reduzir significativamente a quantidade destes para que os métodos possam analisá-los. Nesta tese, quando necessário, foi realizada uma *análise de componentes principais* (PCA), que na maioria dos casos, reduz significativamente a quantidade de atributos, indicando em porcentagem o quanto da informação original é mantida quando utilizamos apenas as primeiras componentes. Mais detalhes sobre a PCA podem ser vistos na próxima seção.

3.2.2.1 Análise de componentes principais (PCA)

Em conjuntos de dados com muitos atributos, muitas vezes acontece de grupos de atributos conterem informação redundante. Este fato pode ter sua origem na técnica experimental ou nos instrumentos de medição utilizados, que podem estar medindo vários atributos que estão relacionados ao mesmo fator de comportamento de um determinado sistema. Nestes casos, é possível simplificar o problema em questão, substituindo um grupo de atributos por um novo atributo. Esta simplificação pode ser feita utilizando uma análise de componentes principais (PCA).

A partir do conjunto de dados original, a PCA gera um novo conjunto de dados, cujos novos atributos são chamados de *componentes principais*. Cada componente principal é uma combinação linear dos atributos originais. Todos os componentes principais são ortogonais entre si, formando uma base de um espaço vetorial, na qual a projeção dos atributos originais neste espaço não contém nenhuma informação redundante. O conjunto completo de componentes principais é do mesmo tamanho do conjunto de atributos original, mas as primeiras componentes concentram a maior parte da informação, portanto é possível utilizar menos atributos para representar o mesmo conjunto de dados.

A primeira componente principal é escolhida de forma que a variância deste atributo seja a máxima entre todas as outras escolhas possíveis. A interpretação geométrica da primeira componente principal é um eixo no espaço de atributos que está na direção da máxima variância. A segunda componente principal é outro eixo no espaço, perpendicular

ao primeiro.

Resumidamente, o problema que a PCA resolve é: a partir de um conjunto de atributos $\mathbf{X} = [X_1, ..., X_n]$, deve-se encontrar uma matriz **P**, de componentes principais, que faça com que o novo conjunto de atributos $\mathbf{Y} = [Y_1, ..., Y_n]$ seja não correlacionado, utilizando a fórmula $\mathbf{Y} = \mathbf{XP}$. A matriz **P** é chamada de *matriz de autovetor*es. A partir de **P**, é possível obter uma *matriz de autovalor*es, que representa os valores das variâncias de **Y**. Esta matriz de autovalores é ordenada de forma crescente, e cada autovalor é relacionado com um atributo Y_i. Ao utilizar os dados **Y** em uma análise, os atributos que estejam associados a valores muito pequenos de variância podem ser desconsiderados. Associando as variâncias à informação contida nos dados, considera-se que as primeiras componentes principais guardam quase toda a informação do conjunto original.

Mais informações sobre o cálculo da PCA podem ser vistas em [30]. Vários artigos que aplicam a técnica de Imunoblot, como em [3] e [31], usam PCA para analisar visualmente os agrupamentos de indivíduos, usando um gráfico bidimensional, ou seja, utilizam apenas as duas primeiras componentes principais.

3.2.3 Normalização e discretização

Outra transformação de dados bastante utilizada é a normalização, para que o intervalo de valores não tenha uma variação muito grande. Alguns algoritmos, como por exemplo o *k-means* para agrupamentos, funcionam muito melhor com dados normalizados. As normalizações mais comuns são as que deixam os dados com valores nos intervalos $[0..1]$ ou $[-1..1]$. A normalização $[-1..1]$ pode ser aplicada utilizando a seguinte fórmula:

$$\frac{(x-\bar{x})}{3\sigma}$$

onde x representa o valor, $\bar{x}$ representa o valor médio do vetor, e σ é o desvio padrão do vetor. No presente trabalho, esta foi a normalização escolhida, pois usando esta fórmula, 99% dos valores se encontrarão no intervalo $[-1..1]$, e os *valores aberrantes* (*outliers*) ficarão fora deste intervalo, facilitando assim sua localização e impedindo que

estes alterem muito os demais valores. Além disso, essa normalização faz com que a média tenda para zero, o que será favorável na aplicação de algoritmos como o *k-means.*

Por fim, mais uma manipulação foi utilizada neste trabalho, a *discretização* dos dados. Esta consiste em dividir os dados que sejam valores reais em várias partições. As partições, ou grupos, podem ter sempre o mesmo tamanho, ou seguir algum critério de separação definido, e cada um deles pode ser nomeado de modo a facilitar a leitura dos resultados. Por exemplo, se os dados são valores reais no intervalo 0..100, e este intervalo é quebrado em 10 grupos iguais, os dados passarão a ser identificados pelo grupo, não por seu valor original. Estes grupos poderiam ser nomeados, por exemplo, com os valores inicial e final do grupo: primeiro grupo: "0..10"; segundo grupo: "11..20"; e assim por diante; ou com nomes que indiquem alguma característica destes grupos. Vários trabalhos usam a discretização de dados em aplicações de Bioinformática, como a ferramenta de análise de *microarrays* da referência [32], e a aplicação de discretização como método de compressão de dados vindos de *microarrays* da referência [33].

No presente trabalho, esta manipulação foi aplicada pois um dos algoritmos utilizados, o de Regras de Associação, requer dados discretizados. Além disso, a discretização aproximou os dados numéricos do contexto biológico, pois os nomes escolhidos para os grupos foram termos já utilizados pelos especialistas em biologia. Por exemplo, nos dados de Imunoblots, o termo "Reatividade Alta" é usado pelos biólogos para denominar valores acima de um determinado limite. Este mesmo termo foi utilizado na discretização para rotular valores muito altos.

3.3 Mineração de dados

Nesta etapa é que os algoritmos e métodos (de diversas áreas) são adaptados para mineração de dados e utilizados sobre o banco de dados já pré-processado. Em especial, nas próximas seções, serão descritos métodos baseados em estatística e aprendizado de máquina, incluindo extração de regras de associação e os algoritmos de agrupamento *k-means, fuzzy c-means*, e SOM, que são utilizados na análise computacional feita neste trabalho. Os métodos estatísticos que não serão abordados, como análise de discriminante

e ANOVA, podem ser consultados em livros como em [30]; e os de aprendizado de máquina e inteligência artificial, como árvores de decisão, redes bayesianas, e redes neuronais *feed forward*, podem ser vistos em livros como em [34]. Outra ferramenta que, apesar de não ter sido utilizada neste trabalho, é interessante para mineração de dados, é o Weka [35]. Esta foi desenvolvida na Universidade de Waikato, Nova Zelândia, e está disponível gratuitamente em http://www.cs.waikato.ac.nz/ml/weka/ . O Weka simula diversos algoritmos, entre eles *k-means*, árvores de decisão, redes bayesianas, e redes neuronais, sendo útil para comparar o desempenho destes sobre uma base de dados.

O objetivo desta etapa de mineração pode ser um ou vários dos descritos a seguir:

- Encontrar estruturas repetitivas nos dados, indicando padrões. Por exemplo, indicar que um determinado atributo tem valores sempre muito próximos, ou que dois atributos com grande freqüência aumentam ou diminuem seus valores juntos, ou ocorrem juntos.
- Criar modelos a partir dos dados, e com isso, ser capaz de prever próximas ocorrências. Se não se sabe ao certo o que leva cada atributo a variar seus valores, ou o que leva cada registro a ser apontado para determinadas classes, é possível criar um modelo que "aprende" quais são os fatores que levam às mudanças entre diferentes classes, utilizando para tal os próprios exemplos observados em uma etapa de treinamento.
- Agrupar os dados por semelhança. Por exemplo, criar grupos de consumidores que compram produtos similares.

Quaisquer que sejam os objetivos escolhidos, levando-se em conta que os desafios podem envolver especialistas de áreas distantes da computação e da matemática, deve-se ter uma preocupação adicional com a visualização dos resultados. Se esta visualização não for clara, pode ser que o especialista que irá analisá-los não possa indicar se as informações adquiridas são interessantes, se fazem sentido, e se realmente podem auxiliar

no entendimento dos dados.

Para escolher os métodos que serão utilizados, primeiramente deve-se perguntar se o problema em questão é *supervisionado* ou *não supervisionado*. Se nos dados há indicação de classes para cada registro, então o problema é supervisionado. Por exemplo, sobre uma base de clientes de um banco, o gerente os classificou como "bons pagadores" ou "maus pagadores"; ou de quatro espécies diferentes de plantas foram medidas várias características (cada espécie é uma classe). Para este tipo de problema, há muitos tipos de métodos adequados: redes neuronais e bayesianas, classificação *fuzzy,* técnicas de estatística mais simples, etc.

Se nos dados não há classes explícitas definidas *a priori*, o problema é considerado *não supervisionado*. Para estes problemas, há menos estratégias propostas. Na maior parte das vezes, a solução é utilizar algoritmos agrupadores. Nestes, o objetivo é agrupar dados por similaridade, e ao mesmo tempo diferenciar ao máximo os grupos criados. Nas seções 3.3.2, 3.3.3, e 3.3.4 será falado mais sobre os algoritmos agrupadores utilizados neste trabalho. Outra solução para dados sem classe é conhecida como *extração de regras de associação*, que será descrita na seção 3.3.5.

Muitas vezes, o problema em Bioinformática é não supervisionado, e em vários trabalhos da área encontram-se aplicações de algoritmos de agrupamento. Por exemplo, a referência [32] é de uma ferramenta que utiliza várias técnicas de agrupamento (entre elas o *k-means*) para analisar dados vindos de *microarrays*. Outro exemplo é o artigo [36], que compara várias técnicas de agrupamento (entre elas, *k-means*, SOM, e discriminante linear), e conclui que para aqueles dados de proteômica[2], o melhor desempenho é o de um algoritmo de agrupamento baseado em um modelo Bayesiano e na transformada de Fourier.

Na próxima seção serão explicados os métodos de estatística utilizados neste trabalho, que podem ser aplicados tanto sobre dados supervisionados quanto não supervisionados.

[2]Proteômica é a pesquisa com foco nas proteínas, em vez de genes ou seqüências de nucelotídeos.

3.3.1 Análise estatística

Como o processo de KDD é iterativo, às vezes uma determinada técnica utilizada na etapa de mineração de dados pode servir como pré-processamento para uma outra técnica. Portanto, as análises estatísticas que foram utilizadas neste trabalho, e que estão descritas nesta seção como parte da etapa de mineração de dados, podem aparecer na etapa de pré-processamento dos dados em outras aplicações.

Informações estatísticas simples, como valores máximo, mínimo e médias de cada registro, de cada atributo, ou dos dados como um todo, auxiliam na utilização dos métodos e na criação dos gráficos para visualização dos resultados. Além disso, podem ser extraídas matrizes de correlação dos dados, onde cada valor é um índice no intervalo $[0..1]$ que indica o nível de similaridade entre dois registros: quanto mais próximo de 1, mais semelhante.

Ainda sobre valores máximo, mínimo e médias de cada registro, existe um gráfico que facilita a visualização destes indicadores, chamado de *box-plot*. Neste, no eixo y estão os valores máximo e mínimo da base inteira, e no eixo x, a quantidade de registros. Para cada registro, é mostrado, na vertical, os valores mínimo, máximo, média, e uma caixa que indica os valores que se encontram entre o primeiro e o terceiro quartil (veja figura 3.2).

Outro gráfico utilizado foi o histograma, que expõe as variações de valores, registro a registro ou atributo a atributo. Estes gráficos serão utilizados para auxiliar a visualização da variação de valores dos dados de Imunoblots. Já com os dados de *microarrays* foi utilizado o gráfico de função de distribuição cumulativa (CDF), que como o histograma, permite analisar visualmente a distribuição de freqüências dos dados, e que será usado como parâmetro de comparação nas simulações computacionais construídas.

Considerando que f é uma função de densidade de probabilidade para uma variável X, a CDF F associada a f é:

$$F(x) = P(X \leq x) = \int_{-\infty}^{x} f(t)dt$$

Desta forma, a CDF de um valor x, $F(x)$, é a probabilidade de que uma observação no conjunto de dados seja menor ou igual a x. A CDF empírica $F(x)$, calculada a partir dos

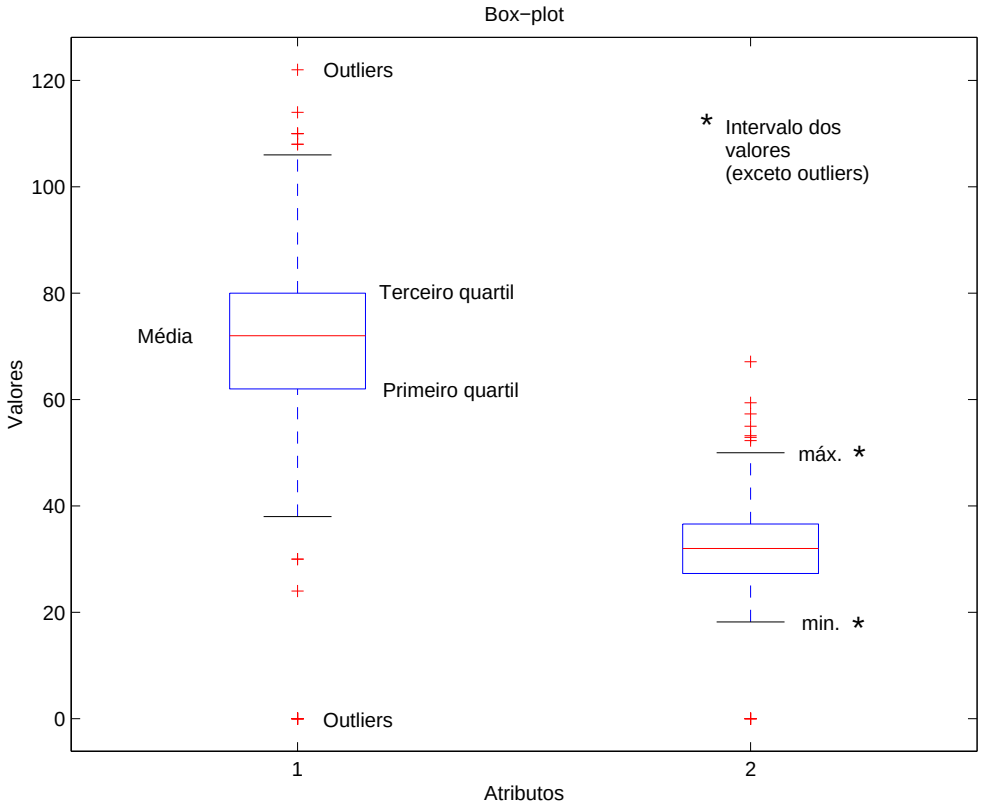


Figura 3.2: Exemplo de *Box-plot*

dados, representa a proporção de observações no conjunto de dados menor ou igual a x.

A outra parte da análise estatística que foi aplicada aos dados de *microarrays* consistiu em acompanhar a evolução de alguns medidores conforme as diversidades aumentavam. Estes medidores são: a média, a mediana, o desvio padrão e o coeficiente de variação.

A média e a mediana são medidas de tendência central. Enquanto a média é calculada somando todos os valores da amostra e dividindo pela quantidade de amostras, a mediana é obtida da seguinte forma: após ordenar os elementos da amostra, a mediana é o valor (pertencente ou não à amostra) que divide esta amostra ao meio, ou seja, 50% dos valores da amostra são menores ou iguais à mediana, e os outros 50% são maiores ou iguais.

Já o desvio padrão e o coeficiente de variação são medidas de dispersão dos dados. O primeiro é calculado como a raiz quadrada da variância dos dados, ou seja,

$$\sigma = \sqrt{var(X)}$$

Considerando que $\mu = E(X)$ é o valor esperado (média) de uma variável aleatória X, a variância pode ser calculada como:

$$var(X) = E((X - \mu)^2)$$

Enquanto o desvio padrão é da mesma ordem de grandeza dos dados, o coeficiente de variação (C_v) é um valor absoluto, que pode ser interpretado como a variabilidade dos dados em relação à média. Quanto menor for este C_v, mais homogêneo é o conjunto de dados. O C_v é calculado pela seguinte fórmula:

$$C_v = \frac{\sigma}{\mu}$$

Além desses medidores, uma outra análise estatística aplicada aos dados de *microarrays* foi a de Regressão Linear. Esta técnica procura a equação linear (da forma $y = ax + b$, no caso de uma única dimensão) que melhor se ajuste a um conjunto de pontos, de forma que novos valores consultados possam ser obtidos diretamente pela interpolação da reta traçada por esta equação linear. Uma das formas de se calcular os valores de a e b é usando o método dos mínimos quadrados, que consiste em minimizar a soma dos quadrados dos desvios verticais dos pontos para a reta. Dado um conjunto n de pares (x, y), as seguintes fórmulas calculam os coeficientes a e b:

$$a = \frac{n \sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \cdot \sum_{i=1}^{n} y_i}{n \sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i \right)^2} \qquad b = \frac{\sum_{i=1}^{n} y_i \cdot \sum_{i=1}^{n} x_i^2 - \sum_{i=1}^{n} x_i y_i \cdot \sum_{i=1}^{n} x_i}{n \sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i \right)^2}$$

Uma vez achada a equação da reta, é importante determinar a precisão do ajuste dessa reta aos dados reais, pela fórmula:

$$R = \frac{n \sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \cdot \sum_{i=1}^{n} y_i}{\sqrt{n \sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i \right)^2} \cdot \sqrt{n \sum_{i=1}^{n} y_i^2 - \left(\sum_{i=1}^{n} y_i \right)^2}}$$

O resultado de R fica no intervalo $[-1..1]$. Quanto mais próximo de 1 ou de -1, melhor terá sido o ajuste da reta.

3.3.2 *k-means*

Este algoritmo simples e clássico de agrupamento de dados utiliza a métrica Euclideana (representada abaixo por $d(x(t), w_i)$) para cálculo das distâncias entre pontos. Cada ponto é um registro da base de dados. Considere que:

$T = \{(x(t)), t = 1..N\}$ representa o conjunto de treinamento, onde cada $x(t)$ é um registro, sendo N o número total de registros;

e $x(t) = (x_1(t), ..., x_p(t)) \in X^p$, ou seja, cada $x(t)$ é um vetor de p atributos, onde X^p representa o universo de atributos.

Para utilizar o *k-means*, deve-se definir *a priori* a quantidade de grupos (K) que se deseja encontrar nos dados escolhidos. Ao iniciar a execução do algoritmo, K sementes deverão ser selecionadas. Há diversas formas de seleção destas sementes: aleatoriamente; usando os primeiros K registros; escolhendo os K registros mais distantes entre si; especificando K pontos espaçados regularmente em uma grade dentro do domínio; etc.

Prosseguindo a execução, cada um dos registros será ligado a uma e somente uma destas sementes, que serão substituídas por centróides, ou centros do agrupamento. Os valores dos centróides serão calculados pelo valor médio dos grupos, e se encontrarão armazenados na matriz $W = [w_1, ..., w_K]$, onde cada coluna $w_i \in X^p$ define as coordenadas do centro do agrupamento.

A partir da definição desta matriz W, o algoritmo busca minimizar um critério de erro $J(W)$, baseado na distância entre os registros da base de dados e os centros dos agrupamentos, procurando onde estes centros se encontrariam melhor localizados:

$$J(W) = \frac{1}{N} \sum_{t=1..N} \sum_{i=1..K} d(x(t), w_i)^2$$

O critério de parada utilizado para interrupção do algoritmo é quando a norma da diferença entre os valores da matriz W em duas iterações sucessivas é menor que uma tolerância especificada δ:

$$\|W_\eta - W_{\eta-1}\| \leq \delta, \text{ onde } \eta \text{ é a iteração atual.}$$

Como essas medidas de distância são diretamente influenciadas pela escala de valores dos dados, é indicado que os dados sejam normalizados antes da aplicação deste tipo de algoritmo agrupador.

O algoritmo do *k-means* segue os seguintes passos:

1- Definir o número K de grupos a serem procurados;

2- Estimar a posição inicial dos K centros de agrupamentos (neste trabalho, a inicialização foi aleatória);

3- Enquanto o critério de parada não for alcançado, repita:

3.1- (Re)agrupar cada registro $x(t)$ do conjunto de treinamento no agrupamento K, escolhendo a menor distância do registro a um dos centros.

3.2- Atualizar a matriz W, calculando os novos valores dos centros pela média das coordenadas dos pontos de cada agrupamento.

4- Fim.

3.3.3 *Fuzzy c-means*

Embora o *k-means* seja um algoritmo simples e de rápida execução, algumas de suas limitações têm sido alvo de críticas, e conseqüentemente, de propostas de modificação. Uma destas é a característica de que o *k-means* só permite que um registro esteja associado a um único grupo. BEZDEK [37] propôs que a distância dos registros aos centros de agrupamentos deveriam ser ponderadas por um valor de pertinência, modificando o critério de erro para:

$$J(m,W) = \sum_{t=1..N} \sum_{i=1..K} u_i(t)^m d\,(x(t), w_i)^2, \text{ onde}$$

m é o parâmetro que regula a forma das funções de pertinência;

$u_i(t) = \mu_{\omega_i}(x(t))$ é o valor de pertinência do registro $x(t)$ ao agrupamento $\omega_i, i = 1..K$.

Desta forma, utilizando noções de lógica *fuzzy* (difusa), cada registro pode pertencer a mais de um agrupamento, e este algoritmo passa a se chamar *fuzzy c-means* (ou FCM). A

matriz de partição U guarda os valores de pertinência de cada registro a cada agrupamento (valores no intervalo $[0..1]$), sendo que cada linha é relativa a um registro, e cada coluna aos agrupamentos. Esta matriz U deve seguir as seguintes propriedades:

- a soma de cada linha deve ser igual a 1, isto é, ou o registro pertence a um agrupamento com pertinência máxima 1, ou as pertinências aos agrupamentos devem somar 1.
- Todo registro deve pertencer a pelo menos um agrupamento.
- Nenhum agrupamento pode reter todos os registros, ou seja, o número mínimo de agrupamentos é 2.

O algoritmo do FCM segue quase os mesmos passos do *k-means*:

1- Escolher os parâmetros K (número de grupos) e m;

2- Inicializar aleatoriamente os K centros de agrupamentos;

3- Calcular a matriz de partição inicial usando a seguinte fórmula:

$$u_i(t) = \frac{\frac{1}{d\left(x(t),w_i\right)^{\frac{2}{(m-1)}}}}{\sum_{j=1..m} \frac{1}{d\left(x(t),w_i\right)^{\frac{2}{(m-1)}}}}$$

4- Enquanto o critério de parada não for alcançado, repita:

4.1- Calcular novos valores dos centros de agrupamentos a partir da matriz de partição atual, usando a fórmula:

$$W_i = \frac{\sum_{t=1..N} u_i(t)^m . x(t)}{\sum_{t=1..N} u_i(t)^m}$$

```
    4.2- Atualizar a matriz de partição, utilizando a fórmula
    do passo 3;

5- Fim.
```

Neste algoritmo, além de calcular o critério de parada como no *k-means*, pode-se também levar em consideração a variação da matriz de partição entre uma iteração e outra:

$\|U_\eta - U_{\eta-1}\| \leq \delta_2$, onde

η é a iteração atual, e

δ_2 um outro (ou o mesmo) valor de tolerância.

Observe que, tanto para o FCM como para o *k-means*, a tendência é que estes caiam em mínimos locais a cada execução, resultando assim em diferentes configurações de agrupamentos para cada um destes mínimos locais. Quando a inicialização é aleatória, pode-se executá-los diversas vezes até que a configuração mais estável seja detectada, ou seja, até que seja possível identificar uma determinada configuração que apareça na maioria das execuções. Espera-se que esta configuração seja também o mínimo global do problema. Neste trabalho, ambos foram executados 10 vezes, para cada base de dados, escolhendo a configuração que apareceu em mais do que 50% das vezes. Em caso de dúvida, executamos mais 10 vezes.

3.3.4 Mapas Auto-organizáveis (SOM)

A inspiração para os algoritmos de *Redes Neuronais Artificiais* vem do modelo que temos do funcionamento do sistema nervoso, mais especificamente sobre o modelo do cérebro humano e dos neurônios que o compõe. Um *Mapa Auto-organizável* (*Self-organizing Map - SOM*) [38] é um tipo de rede neuronal artificial que simula a noção

de que o conhecimento é armazenado no cérebro de acordo com uma certa topologia, isto é, conceitos iguais ficam juntos, outros não tão similares ficam próximos, e conceitos completamente diferentes se encontram bem distantes. A idéia do SOM não põe os conceitos em agrupamentos com fronteiras rígidas, completamente separados, mas sim, cria regiões que mudam aos poucos de um conceito para o outro. Isto implica que não é necessário especificar a priori a quantidade de agrupamentos que se quer buscar. Ao invés disso, a resposta do SOM deve ser analisada para entender como os dados estão distribuídos no domínio, e se for o caso, a partir desta análise, pode-se determinar a quantidade de agrupamentos (veja exemplo na figura 3.3). Para criar esta resposta, o SOM usa um método de *aprendizado competitivo*, onde os neurônios presentes na camada de saída competem entre si para responder por cada *estímulo* (registro) apresentado. Note que o SOM é uma rede neural artificial de apenas duas camadas completamente conectadas entre si: *camada de entrada*, para receber os registros; e *camada de saída*, onde é possível visualizar a resposta.

Ao ser escolhido como neurônio vencedor, este tem seus *pesos* (ligações com a camada de entrada) aumentados. Os neurônios próximos ao vencedor também têm seus pesos aumentados, mas com menor intensidade, e esta intensidade diminui conforme a distância do vencedor cresce. O quanto o peso será aumentado é chamado de taxa de aprendizado, e é um parâmetro a ser definido para utilizar o SOM. Conforme cada entrada é apresentada, e os pesos de certas regiões na camada de saída são aumentados, estas regiões vão sendo treinadas para reconhecer aquele tipo de estímulo.

Além disso, para se utilizar um SOM, deve-se especificar o que é mais adequado para o problema proposto, dentre os seguintes pontos:

- A resposta do SOM é uma grade n-dimensional de neurônios. Se a quantidade de neurônios for igual ou próxima da quantidade de registros, a tendência é ter uma resposta como a da figura 3.3. Se for aumentada a quantidade de neurônios em relação à quantidade de registros, a tendência é que as regiões associadas com os registros se encontrem mais separadas no espaço, facilitando assim a separação de agrupamentos.

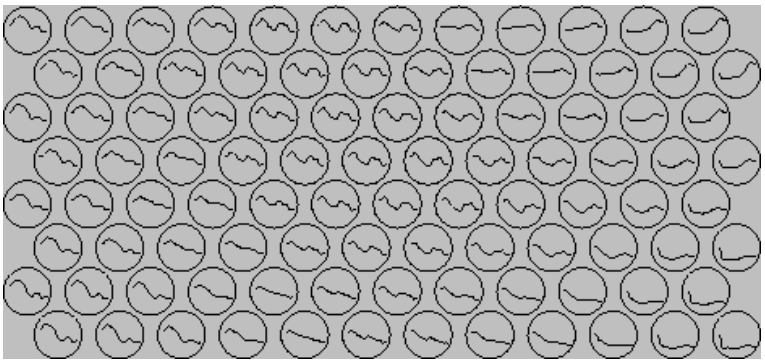


Figura 3.3: Exemplo de resposta de um SOM. Os círculos representam as posições dos neurônios, e a curva dentro deles, o modelo que se quis representar. Note que modelos vizinhos são mutuamente similares. Figura retirada de [39].

- A maneira como os neurônios da camada de saída estão conectados é chamada de topologia, e os tipos mais comuns de topologias são: em grade, hexagonal e aleatoriamente dispostos. Veja na figura 3.4 as topologias em grade e hexagonal. A topologia influi diretamente no cálculo de quem são os vizinhos do neurônio vencedor.

- É necessário especificar a *vizinhança*, ou seja, como será calculada a distância entre neurônios, e até qual distância os vizinhos de um neurônio vencedor terão os pesos alterados. Por exemplo, podem ser utilizados neste cálculo somente os vizinhos mais próximos, que estão diretamente conectados ao neurônio vencedor.

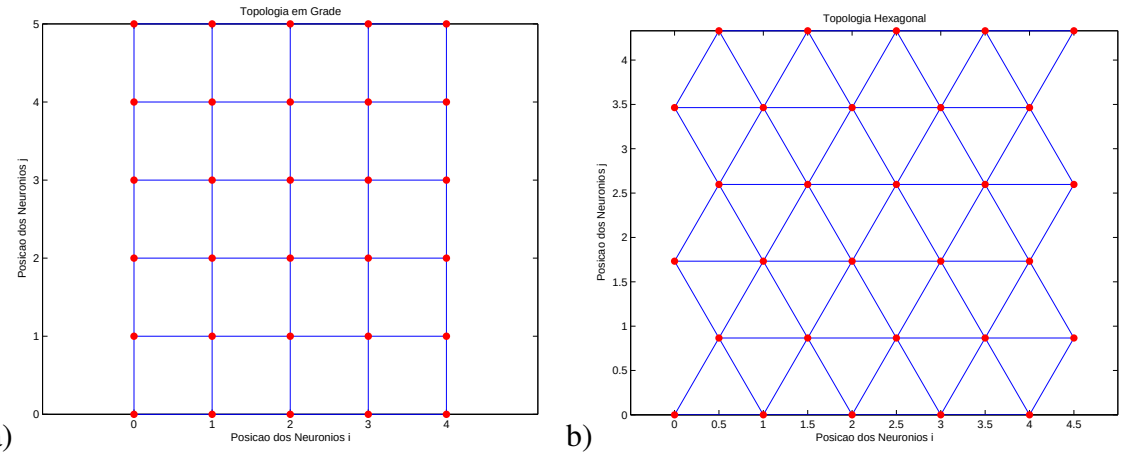


Figura 3.4: Exemplo de posições dos neurônios em topologias bidimensionais i, j.
a) Topologia em Grade. b) Topologia Hexagonal.

Após serem definidas as características da arquitetura do SOM, o algoritmo segue os seguintes passos:

1- Inicializar os pesos θ_{ij} entre a camada de entrada i e a camada de saída j.

2- Enquanto o critério de parada não for alcançado, repita:

2.1- Ler um registro x_i.

2.2- Calcular, para cada neurônio j, qual é seu valor de resposta, usando a fórmula: $\sum_i \theta_{ij} x_i$

2.3- Avaliar qual foi o neurônio vencedor $j*$, procurando por aquele que teve o maior valor de resposta, e atualizar seus pesos usando a fórmula:

$$\theta_{ij}(t) = \theta_{ij}(t-1) + \alpha\left(x_i - \theta_{ij}(t-1)\right),$$

onde t é a iteração atual,
e α é a taxa de aprendizagem.

2.4- De acordo com a função de vizinhança $N_{j*}(d)$ escolhida, onde d é o raio de alcance da atualização dos pesos, atualizar os neurônios na vizinhança do vencedor, diminuindo a taxa de aprendizado α de acordo com um parâmetro escolhido.

3- Fim.

3.3.5 Extração de regras de associação

A vantagem de se utilizar um método baseado em extração de regras em relação a métodos estatísticos e/ou puramente numéricos ou de aglomeração, é que as regras são mais expressivas e mais fáceis de interpretar. Uma regra descreve um modelo de dados através de seus atributos, o que permite revelar os atributos mais importantes ou significativos que definem um determinado padrão para classificação. No entanto, esta vantagem pode se tornar uma desvantagem, se o número de regras obtido pelo algoritmo de extração de regras for muito grande, tornando quase impossível a interpretação do modelo por parte do especialista.

Métodos poderosos de extração de regras, por exemplo, a ILP (*Inductive Logic Programming*), baseado em lógica de primeira ordem, podem ser aplicados em problemas de Bioinformática, como na referência [40], onde os autores aplicam ILP sobre uma base de dados de imagens de mamografias e resultados de biópsias, e encontraram regras consideradas interessantes pelos especialistas em mamografias, e que foram validadas pelo próprio conjunto de dados.

No presente trabalho, como as bases de dados têm poucos registros, e os dados são todos numéricos, não é muito vantajoso usar uma técnica mais elaborada como a ILP, pois é complicado adequar os dados para o algoritmo. Apesar disso, tentamos aplicar ILP aos dados de Imunoblots, mas o método não obteve respostas significativas e regras relevantes para os especialistas biólogos.

Portanto, o algoritmo escolhido para o trabalho desta tese foi o de *extração de regras de associação Apriori*. A primeira proposta de *extração de regras de associação* de grandes bases de dados foi feita em 1993, por AGRAWAL *et al* [41]. Eles criaram o algoritmo *Apriori*, que gera regras proposicionais do tipo "condição $\rightarrow$ conclusão", onde tanto a condição quanto a conclusão podem ter mais de um termo. Este algoritmo recebe como entrada dados nominais (não dados numéricos). Além disso, os registros não precisam ser todos do mesmo tamanho, e os dados de cada registro não precisam seguir uma determinada ordem de atributos. Por estas características, um dos usos mais comuns desta técnica é sobre bases de dados de transações comerciais. Por exemplo,

selecionando compras de vários clientes de um supermercado como entrada para o Apriori, o algoritmo responderia regras do tipo “80% das pessoas que compraram pão e leite também compraram queijo” (pão e leite estão na condição da regra, e queijo na conclusão). Os indicadores principais destas regras são:

- Suporte: porcentagem dos registros que contêm todos os itens da condição e da conclusão de uma regra.
- Confiança: considere os registros que contêm todos os itens da condição da regra; confiança é a porcentagem destas transações que contêm também todos os termos da conclusão. No exemplo citado acima, o valor 80% corresponde à confiança.

O algoritmo *Apriori* segue os seguintes passos:

1- Escolher valores de suporte e confiança mínimos.

2- Sobre toda a base de dados, contar a quantidade de ocorrências de cada item.

3- Criar uma tabela com todos os itens cuja quantidade de ocorrências seja maior que o suporte mínimo.

4- Iterativamente, contar a quantidade de ocorrências de cada combinação de dois itens juntos; contar a quantidade de ocorrências de cada combinação de três itens juntos; etc.

5- Para cada quantidade k de itens a partir de $k = 2$, criar tabelas com todas as combinações de itens cuja quantidade de ocorrências seja maior que o suporte mínimo.

6- Sobre o resultado final da iteração, voltando até $k = 2$, fazer todas as combinações possíveis de itens gerando as regras, e calculando o valor de confiança de cada regra gerada.

7- Guardar somente as regras geradas que tiverem confiança acima do mínimo determinado.

8- Fim.

Para que o algoritmo *Apriori* seja capaz de ler dados numéricos, deve-se dividir os valores dos dados em intervalos discretos (discretizá-los, como sugerido em [42]), ou mesmo em intervalos cujas fronteiras não sejam muito bem definidas, utilizando uma discretização *fuzzy*. Se não for feita alguma discretização dos dados numéricos, dois valores que na escala numérica são bastante próximos, pelas regras de associação serão tratados como dados completamente diferentes. Uma discretização simples resolve esta questão até certo ponto, pois agrupa valores próximos em partições. Mas o último número de uma partição e o primeiro número da partição seguinte também são números bastante próximos, mas que serão tratados como sendo completamente diferentes. Para resolver este problema, no trabalho de KUOK *et al* [43] foi sugerido que estas discretizações podem ser feitas usando *conjuntos fuzzy*. Estes conjuntos *fuzzy* permitem que as bordas dos conjuntos se sobreponham, mas nos valores com sobreposição os graus de pertinência não são máximos. No trabalho [43], é definido que:

- a base de dados é representada por $T = \{t_1, t_2, ..., t_n\}$, sendo t_i o i-ésimo registro;
- os atributos são representados por $I = \{i_1, i_2, ..., i_n\}$, sendo i_j o j-ésimo atributo;
- a cada atributo i_j , podem ser associados vários conjuntos *fuzzy*;
- $F_{ik} = \{f_{ik}^1, f_{ik}^2, ..., f_{ik}^n\}$ representa o conjunto de conjuntos *fuzzy* associados a i_k, e f_{ik}^jrepresenta o j-ésimo conjunto *fuzzy* em F_{ik}.

O *software* utilizado para gerar as regras de associação foi o CBA [44], desenvolvido na Universidade Nacional de Singapura, que é capaz de fazer também a classificação dos dados baseada nas regras encontradas. Este programa gera regras de associação com o seguinte formato:

X1 = *valor1*

...

X2 = *valor2*

-> X3 = *valor3*

...

X4 = *valor4*

(Cover% Conf% CoverCount SupCount Sup%)

onde **X*N*** é um item, e ***valorN*** é o valor deste item; entre parênteses, o quarto valor, **SupCount** representa quantos registros têm todos os itens tanto da condição quanto da conclusão, e o quinto valor, **Sup%**, é o suporte descrito anteriormente, na página 43 (**Sup%**=**SupCount**/quantidade total de registros). **CoverCount** indica quantos registros têm todos os itens da condição, e **Cover%** é a porcentagem (**CoverCount**/quantidade total de registros). **Conf%** é a confiança mencionada anteriormente, na página 43, e a fórmula para seu cálculo é (**SupCount/CoverCount**) * 100.

Um dos grandes problemas da mineração de regras de associação é descobrir regras que realmente sejam úteis e possuam padrões interessantes. Regras com padrões óbvios não são interessantes e devem ser descartadas, mas nem sempre isto pode ser garantido ao se estipular um nível mínimo para o suporte e para a confiança. Muitas vezes, regras com 100% de confiança não são interessantes, mesmo estando corretas. Em grande parte das vezes, este problema só pode ser resolvido levando as regras para o especialista. Caso este aponte algumas regras óbvias, o *software* CBA possui um módulo chamado de IAS (*Interestingness Analysis System*) que se propõe a procurar regras mais interessantes, que seriam as mais ortogonais às óbvias indicadas.

3.4 Análise

Nesta etapa, todos os resultados obtidos devem ser exibidos aos especialistas, para

que estes analisem se as informações apontadas são interessantes. Como mencionado no início da seção 3.3, a visualização destes resultados é fundamental. Muitas vezes, o próprio especialista poderá indicar a forma como quer ver os resultados, e quais são as partes mais importantes do resultado para serem analisadas. Esta visualização pode ser feita simplesmente com gráficos**,** ou utilizando técnicas mais elaboradas. Há, inclusive, uma área de pesquisa chamada de *Data Visualization* (visualização de dados), especializada em buscar soluções para este problema. Mais informações sobre técnicas de visualização podem ser obtidas da referência [45].

Esta etapa da KDD pode encerrar o processo, chegando a resultados interessantes, ou indicar a volta para alguma etapa anterior, para buscar mais resultados. Ao indicar o retorno a alguma etapa anterior da KDD, pode ser que os resultados obtidos na análise possam ser usados como entrada de dados para uma etapa anterior. Por exemplo, na etapa de mineração de dados, um algoritmo agrupador separou os dados originais em grupos, e na etapa de análise os especialistas criaram classes correspondendo a estes grupos. Seria possível então voltar para a etapa de mineração de dados e escolher um algoritmo de classificação supervisionada para criar um modelo para estes dados.

No caso da análise dos resultados indicar o encerramento do processo de KDD, as informações obtidas podem auxiliar na tomada de decisões. Por exemplo, no caso de uma empresa, o resultado pode ser a sugestão de criar promoções conjuntas de itens que foram destacados pelas regras de associação; ou para pesquisadores que tiverem gerado uma primeira bateria de dados, as informações obtidas podem sugerir novas direções para os próximos experimentos, alterações no design experimental, ou indicar uma parte do problema que poderia ser submetida a uma pesquisa mais aprofundada.

Capítulo 4

Resultados e discussão com dados de Imunoblots

Neste capítulo, serão mostrados os resultados da aplicação de alguns métodos descritos no capítulo 3, sobre os dados do experimento descrito na seção 2.2.1. Neste capítulo também estão incluídas a análise destes resultados, e a discussão sobre sua relevância e significado biológico.

O estudo feito no capítulo 2, mais especificamente, as informações adquiridas na seção 2.2.1 sobre o experimento biológico, correspondem ao primeiro passo do KDD, a identificação da base de dados e dos arquivos, descrito na seção 3.1. Sobre o pré-processamento, tudo que foi utilizado foi descrito na seção 3.2. A mineração dos dados e a análise dos resultados estão nas próximas seções. É importante ressaltar que a linha de pensamento que foi utilizada, definida através de consultas aos especialistas que geraram os dados, é a de comparar o dia (-1) do experimento com o último dia medido, procurando comprovar ou refutar a teoria de que o repertório se regenera, mesmo quando submetido a perturbações. Em outras palavras, o objetivo principal do estudo é sobre a cinética de recuperação do repertório de anticorpos. A partir deste ponto, os dias dos experimentos são citados como tempo 1, tempo 2, tempo 3, tempo 4 e tempo 5. Observe que as quimeras de medula óssea terão o tempo 5 como último, enquanto as quimeras de fígado fetal terão como último o tempo 4, ambos representando o *dia 60* do experimento.

No primeiro experimento, detalhado na próxima seção, foi utilizado somente uma matriz, a das médias das reatividades do extrato de fígado, de quimera de medula óssea. O objetivo foi testar com um arquivo menor quais métodos poderiam ser aplicados sobre os arquivos maiores, de dados brutos. Desta forma, no segundo experimento, na seção 4.2, serão utilizados os quatro arquivos de dados brutos, e somente os métodos que adicionaram informação relevante no primeiro experimento.

4.1 Primeiro experimento

Neste, foi utilizada a matriz de quimera de medula óssea, com as médias das reatividades do extrato de fígado. O objetivo deste experimento foi testar alguns métodos de mineração de dados e avaliar quais teriam os resultados mais relevantes, e portanto, quais deveriam ser utilizados sobre os demais dados. Os resultados obtidos neste primeiro experimento foram publicados na referência [46].

4.1.1 Análise estatística

O primeiro passo foi fazer uma análise de componentes principais (PCA) sobre toda a matriz escolhida, a fim de diminuir a quantidade de atributos. Nesta, foi indicado (pela matriz de autovalores) que utilizando apenas as duas primeiras componentes principais, cerca de 75% da informação é mantida. Já se a matriz for normalizada antes, a quantidade de informação mantida cai para cerca de 65%.

Considerando todos os indivíduos, *box-plots* dos tempos 1 e 5 foram traçados (veja figura 4.1). As médias dos valores destes *box-plots* são, respectivamente, $2002,2$ e $1994,0$. O fato destes valores serem bastante próximos sugere que a regeneração do repertório foi bem sucedida, considerando todo o grupo. Ainda nesta linha, foram geradas duas matrizes de correlação entre os indivíduos, uma para o tempo 1, outra para o tempo 5. Os valores médios dessas matrizes são, respectivamente, $0,6590$ e $0,6642$. Isso indica que os repertórios eram um pouco parecidos no tempo 1, e mantiveram quase a mesma similaridade mesmo depois de submetidos às perturbações.

Os histogramas traçados comparando os tempos 1 e 5 de cada indivíduo foram bastante informativos visualmente, e selecionamos alguns para serem comentados.

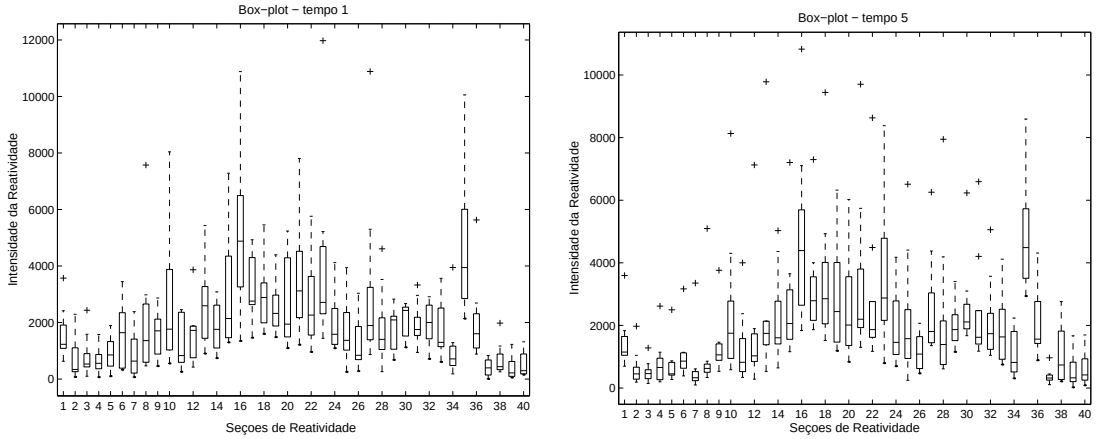


Figura 4.1: *Box-plots*, tempos 1 e 5 no experimento 1

Nas figuras 4.2 e 4.3 estão os histogramas dos camundongos 6 e 9, respectivamente, selecionados para mostrar camundongos cujo repertório original se regenerou de maneira muito forte. Note que a linha representando os valores de reatividade no tempo 5 está quase sobreposta à linha do tempo 1. As matrizes de correlação para comparar os tempos 1 e 5 de cada indivíduo indicaram, de forma geral, uma grande correlação entre estes tempos. Mais especificamente, para o camundongo 6, a média dos valores da matriz de correlação é $0,9036$, e para o camundongo 9, $0,7872$.

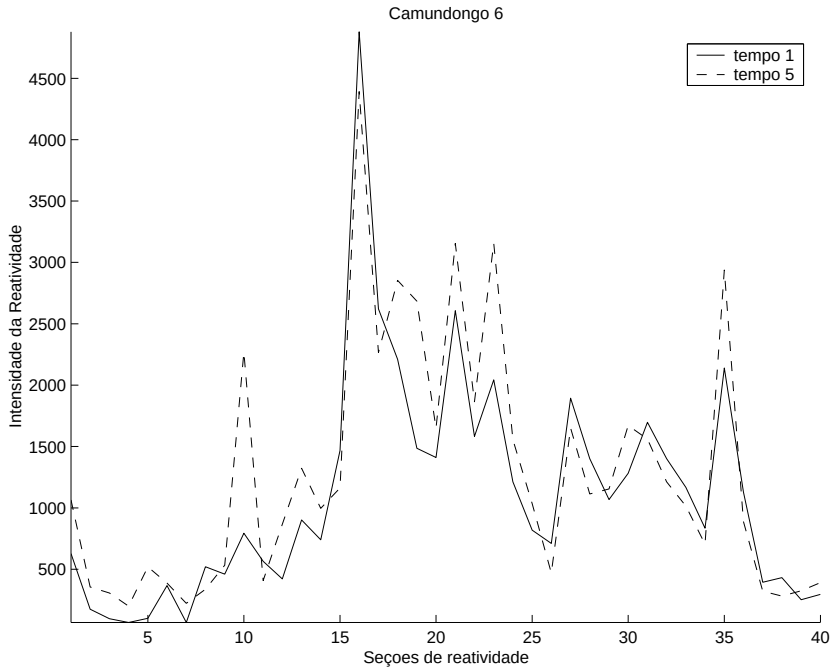


Figura 4.2: Histograma do camundongo 6, experimento 1

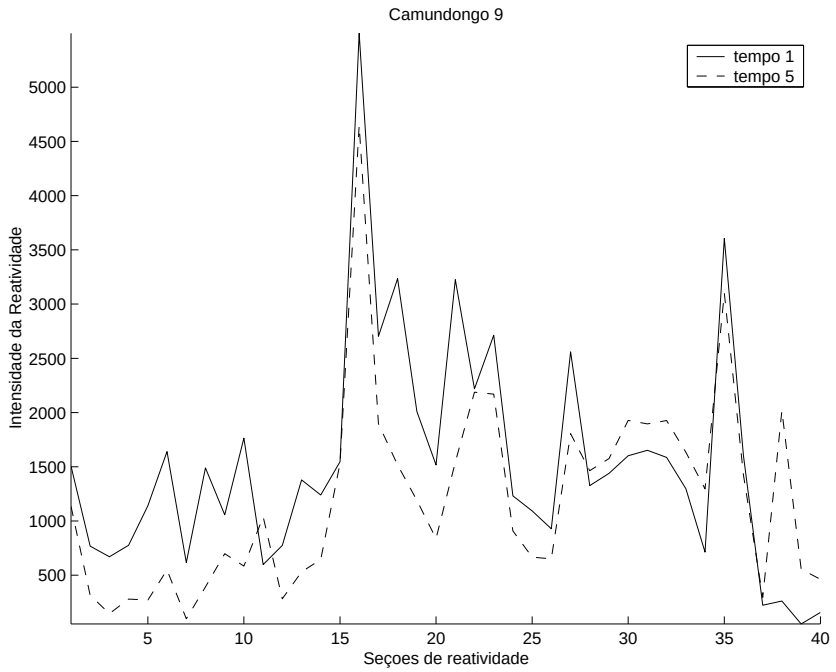


Figura 4.3: Histograma do camundongo 9, experimento 1

Nas figuras 4.4, 4.5 e 4.6, estão exemplos de camundongos cujos repertórios mudaram após a exposição às perturbações. É interessante notar que, embora o intervalo de valores tenha mudado consideravelmente para os camundongos 2 e 3 no tempo 5 (decrescendo e crescendo, respectivamente), aparentemente eles foram capazes de manter a assinatura de seus repertórios, ou seja, os picos e vales dos tempos 1 e 5 são coincidentes. Já para o camundongo 4, o histograma na figura 4.6 indica que não houve regeneração de seu repertório.

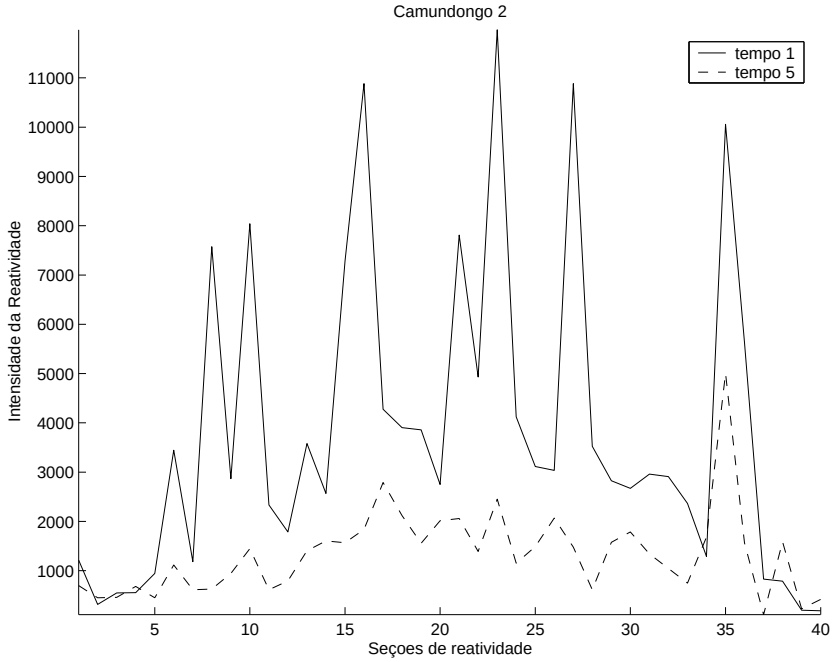


Figura 4.4: Histograma do camundongo 2, experimento 1

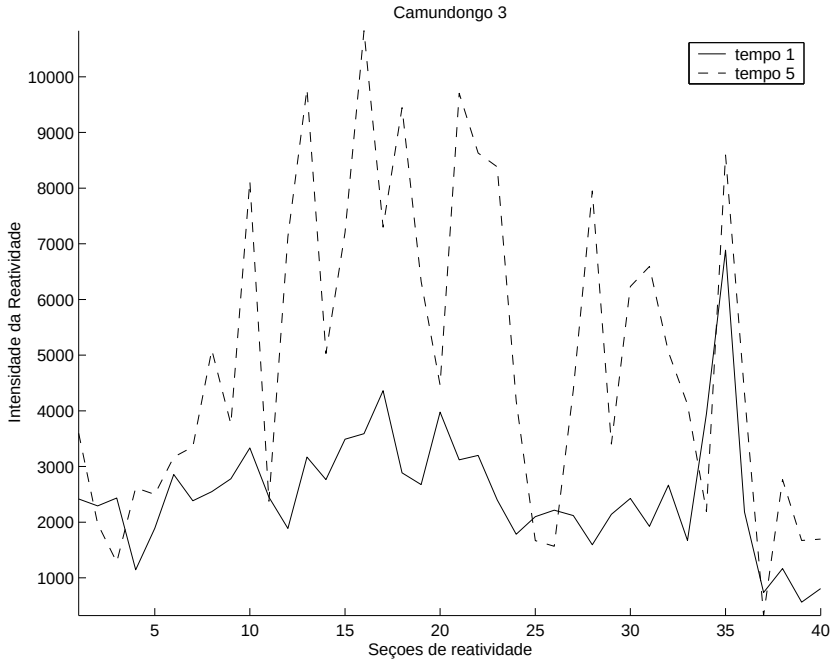


Figura 4.5: Histograma do camundongo 3, experimento 1

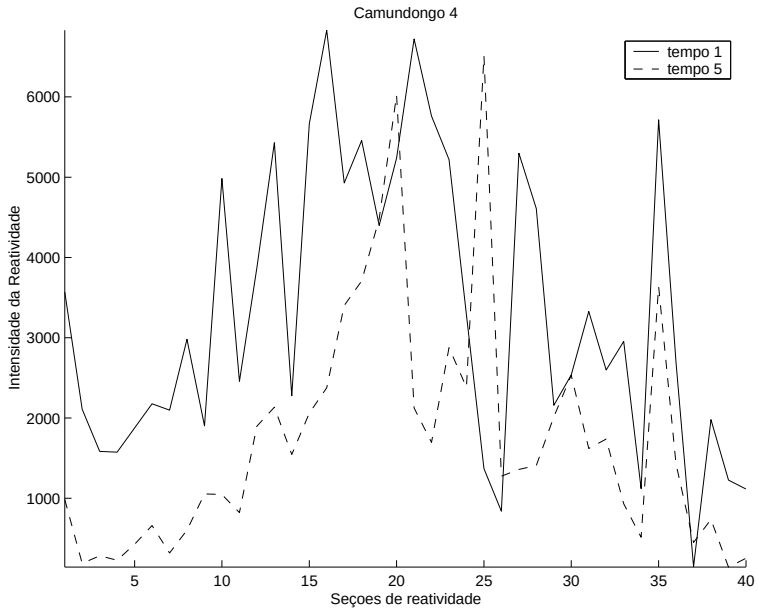


Figura 4.6: Histograma do camundongo 4, experimento 1

4.1.2 Algoritmos de agrupamento

4.1.2.1 *k-means* e FCM

Os algoritmos *k-means* e FCM foram implementados no *software* MatLab [47]. Ambos requerem que o número de agrupamentos (k) seja definido *a priori*, e este parâmetro foi fixado em $k = 3$. Os dados utilizados foram apenas os dois primeiros componentes principais de cada registro, dos dados normalizados. Os dados foram separados de forma que cada gráfico corresponde a um tempo, e os indivíduos são os pontos a serem agrupados. Os símbolos utilizados para representar cada camundongo estão exibidos na tabela 4.1, e os centros dos grupos estão indicados com um asterisco (*). Os agrupamentos indicados foram os mesmos para o *k-means* e para o FCM, em todos os tempos, portanto só serão mostradas as figuras de um dos métodos; neste caso, escolhemos as figuras do FCM.

Como o foco deste trabalho é comparar os tempos 1 e 5, as figuras 4.7 e 4.8 mostram, respectivamente, os gráficos somente destes tempos. Note que, em ambas figuras, a maior parte dos indivíduos se concentra em um grupo. Isso indica que a maioria dos camundongos já tinha um repertório bastante próximo antes da perturbação, e mantiveram suas proximidades mesmo após todo o processo. Isso acontece, por exemplo, com os

camundongos 6 e 9 (▽ e ✡), já mostrados nos histogramas nas figuras 4.2 e 4.3.

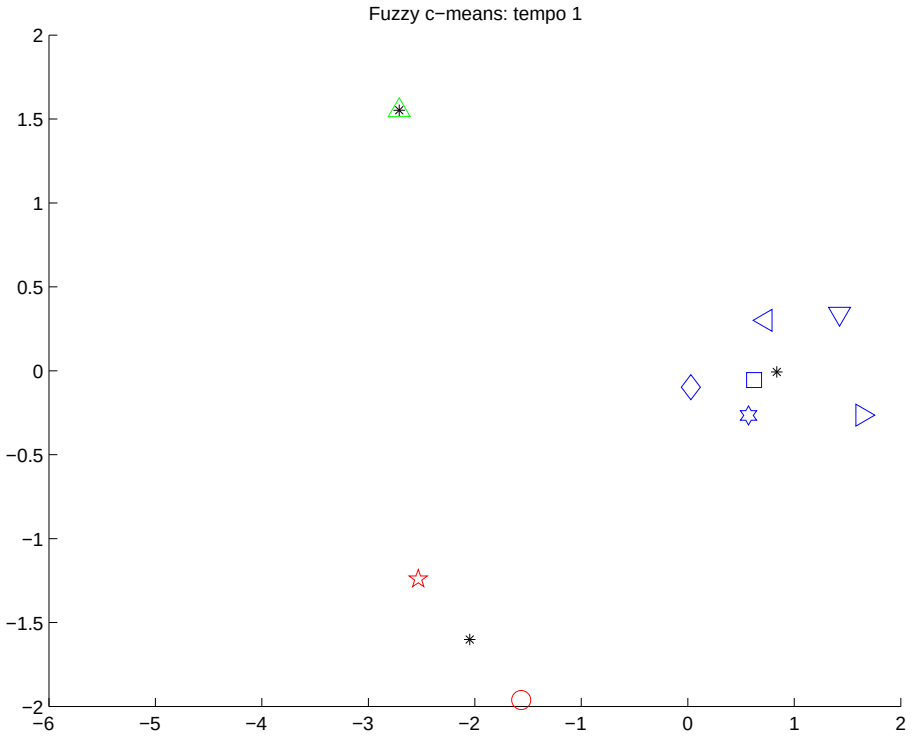


Figura 4.7: FCM tempo 1, experimento 1

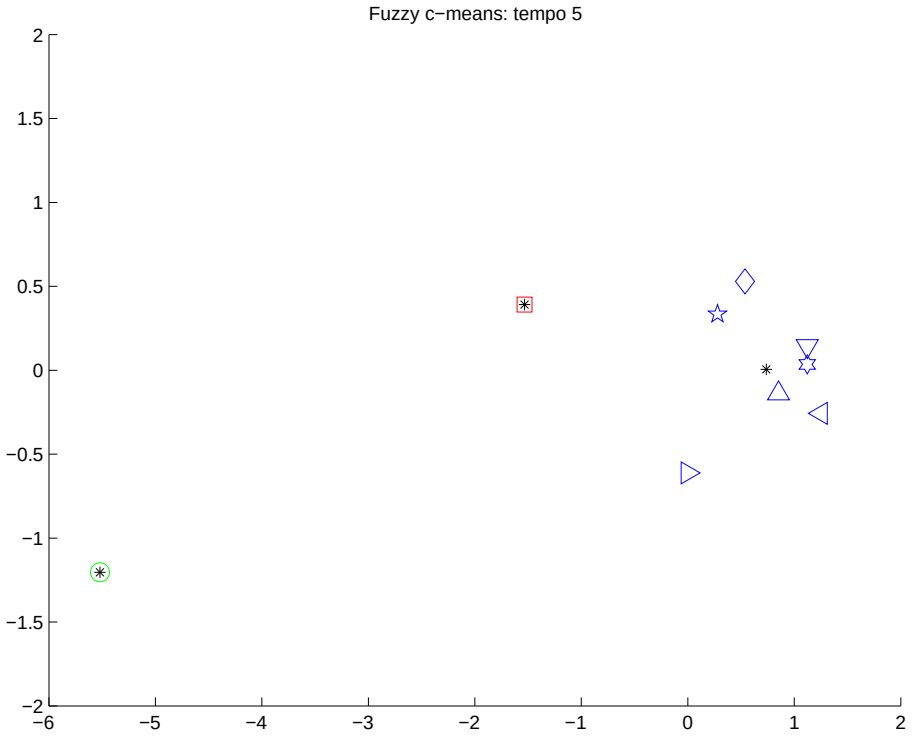


Figura 4.8: FCM tempo 5, experimento 1

Camundongo	1	2	3	4	5	6	7	8	9
Símbolo	◇	△	○	☆	□	▽	◁	▷	✡

Tabela 4.1: Símbolos usados nos gráficos de agrupamento

Note que os camundongos 2 e 4 (△ e ☆), no tempo 1 estão em agrupamentos distantes, e no tempo 5 migram para o agrupamento maior. Mesmo que o camundongo 4 não tenha sido capaz de regenerar a assinatura de seu repertório até o tempo 5 (conforme mostrado na figura 4.6), ele o fez de forma a ficar mais parecido com os demais. Já o camundongo 3 (representado por ○), que no histograma (figura 4.5) mostrou regeneração de sua assinatura, nos gráficos do FCM não aparece no grupo maior em nenhum dos tempos, demonstrando que seu padrão de comportamento é bastante distinto dos demais animais.

4.1.2.2 SOM

Para implementar o SOM, foi utilizada a *Neural Network Toolbox* do *software* MatLab [47]. Vários parâmetros e definições padrão desta *toolbox* foram mantidos, foi modificado apenas o mínimo necessário para adequar o SOM aos dados, para controlar a quantidade e a variabilidade dos experimentos. A figura 4.9 mostra o melhor resultado obtido para estes dados, usando os seguintes parâmetros e definições:

- A camada de entrada possui 2 neurônios, que lêem as 2 primeiras componentes principais geradas a partir dos dados.
- A camada de saída possui 30 neurônios, distribuídos numa grade bidimensional de 5 x 6. Como temos 18 registros (9 camundongos nos tempos 1 e 5), espera-se que com 30 neurônios os grupos criados fiquem bem separados na grade; e que para 2 registos terem o mesmo neurônio vencedor associado a ele, estes 2 têm que ser realmente muito semelhantes.
- A topologia da camada de saída é hexagonal, e a distância entre neurônios é calculada pela quantidade de conexões entre os neurônios.
- A função de vizinhança faz com que a cada passo na distância entre neurônios, o

incremento no valor dos vizinhos seja a metade do passo anterior. Ou seja, para o vencedor o incremento é a própria taxa de aprendizado; para distância do vencedor igual a 1, o incremento será metade da taxa de aprendizado; para distância do vencedor igual a 2, o incremento será metade da taxa de aprendizado aplicada aos vizinhos com distância 1; e assim por diante.

- O SOM foi treinado em duas fases: uma fase de ordenação, e outra de calibração, chamadas respectivamente de *ordering phase* e *tuning phase*. A fase de ordenação corresponde às 1000 primeiras épocas de treinamento, e a taxa de aprendizado inicial é $0,9$. Nestas 1000 épocas, a taxa de aprendizado decresce de $0,9$ a $0,02$, que é a taxa de aprendizado inicial da fase de calibração. Durante a fase de calibração, a taxa de aprendizado continua diminuindo, mas mais devagar do que na fase de ordenação. Como a fase de calibração geralmente leva muito mais tempo, o número de épocas deve ser bem maior, portanto a quantidade total de épocas de treinamento foi definido para 10000.

Na figura 4.9, os pontos são neurônios, as figuras vazadas são os camundongos no tempo 1, e as figuras fechadas, são os camundongos no tempo 5. Os símbolos dos camundongos são os mesmos usados nos gráficos anteriores. As figuras relativas a cada camundongo, representadas sobre um neurônio, indica que este neurônio está associado àquele estímulo (camundongo). Cada neurônio pode estar associado a nenhum ou a mais de um estímulo, mas cada camundongo só estará associado a um neurônio.

Note que nesta figura utilizamos os tempos 1 e 5 no mesmo gráfico, para facilitar a identificação de repertórios que se regeneram. Note ainda que na figura é possível identificar 3 grupos, e que o único evento que mais de um estímulo é associado a um único neurônio foram os do tempo 1 e 5 do camundongo 6 ($\bigtriangledown$), já indicado anteriormente por sua boa regeneração. O camundongo 9 ($\stackrel{\bigtriangleup}{\bigtriangledown}$), também já identificado anteriormente por sua boa regeneração, é o único que têm seus estímulos associados a neurônios vizinhos. Os camundongos 2, 3 e 4 ($\bigtriangleup$, $\bigcirc$ e $\bigstar$), identificados anteriormente por não regenerarem seu repertório, aqui aparecem também com seu estímulo do tempo 1 bastante distante do tempo 5.

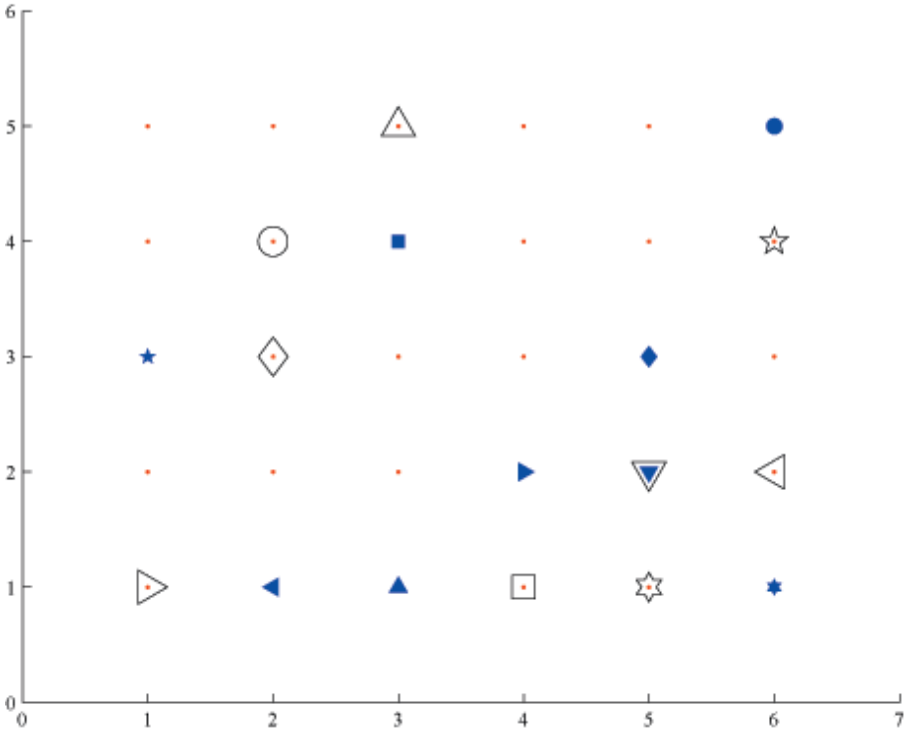


Figura 4.9: SOM para os tempos 1 e 5, experimento 1

4.1.3 Regras de associação

Para aplicação desta técnica, foi escolhida uma discretização *fuzzy* do espaço de valores. De acordo com a definição desta discretização feita na seção 3.3.5, os 43 registros da base de dados são representados por $T = \{t_1, t_2, ..., t_{43}\}$. Para as seções de reatividade (os atributos), foram criadas três diferentes representações: $I_1 = \{R1, R2, ..., R40\}$, sem os atributos indicando o tempo e o indivíduo; $I_2 = \{R1, R2, ..., R40, camundongo\}$, sem o atributo indicando os tempos; $I_3 = \{R1, R2, ..., R40, tempo\}$, sem o atributo indicando os indivíduos. Considerando que não há necessidade de discretizar os atributos *tempo* e *camundongo*, a modelagem da discretização *fuzzy* descrita daqui em diante considera somente os valores de $R1$ a $R40$.

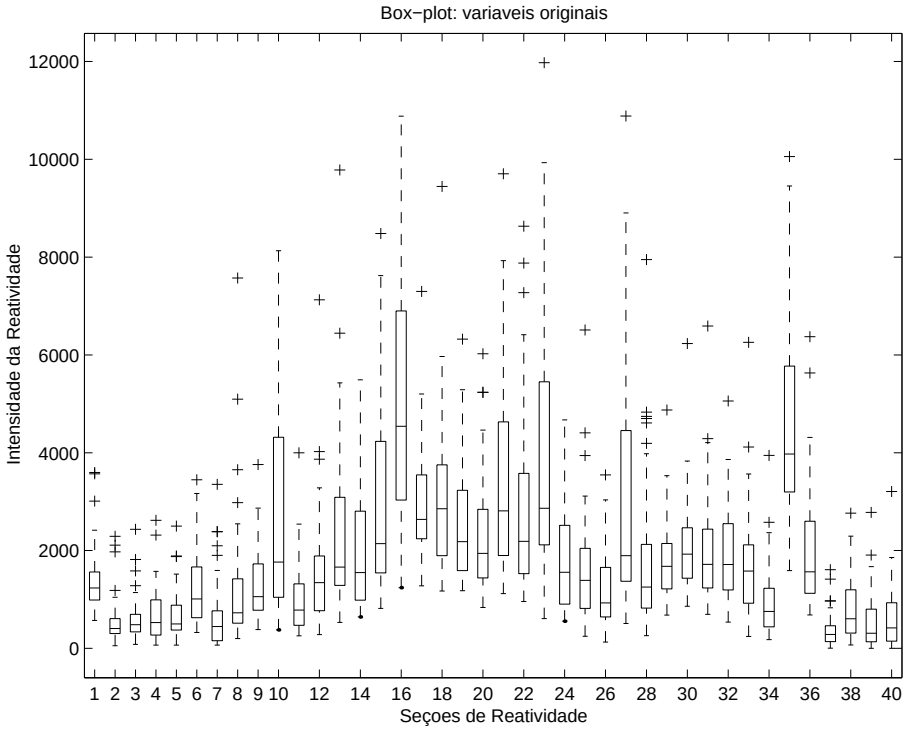


Figura 4.10: *Box-plot* de P1 a P40

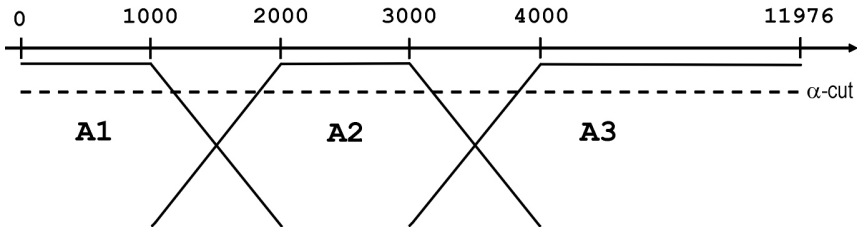


Figura 4.11: Discretização *fuzzy* escolhida

Utilizando o *box-plot* dos dados de $R1$ a $R40$ (figura 4.10), juntamente com a consulta aos especialistas, chegou-se à discretização *fuzzy* indicada na figura 4.11. Nesta, as reatividades foram separadas em 3 grupos: A1 = baixa; A2 = média; e A3 = alta. Na notação indicada na seção 3.3.5, $F = \{baixa, media, alta\}$, ou $F = \{A1, A2, A3\}$. As fronteiras destes grupos foram indicadas pelos especialistas, que identificaram quais faixas de valores eles utilizariam caso tivessem que dividir o domínio em 3. Note que, se o domínio fosse simplesmente dividido em 3 partes iguais, considerando os valores mínimo e máximo da base de dados (0 e 11976, respectivamente), estaríamos superestimando os grupos A1 e A2, e subestimando A3, devido aos altos valores que foram atingidos em

R3 = baixa
-> R2 = baixa
(90.698% 100.00% 39 39 90.698%)

R5 = baixa
R3 = baixa
-> R2 = baixa
(90.698% 100.00% 39 39 90.698%)

R2 = baixa
-> R5 = baixa
  R3 = baixa
(93.023% 97.50% 40 39 90.70%)

Tabela 4.2: Regras para o conjunto I_1

poucos casos (*outliers* e algumas seções que tiveram reatividade muito alta).

No trabalho de KUOK *et al* [43], na parte dos testes ele indica que apenas dados que tivessem valor de pertinência acima de um *threshold* seriam considerados como pertencendo a um conjunto. Em nosso caso, o *threshold* escolhido foi de pertinência $0,8$, ou seja, apenas os valores acima deste α*-cut* foram considerados na discretização (como na linha tracejada marcada na figura 4.11). Os valores entre intervalos foram indicados na base de dados discretizada com o símbolo de interrogação "?".

Numa primeira execução do CBA para o conjunto I_1, fixando o suporte mínimo em 1% e a confiança mínima em 50%, foram geradas 117.657 regras. Aos poucos, estes valores mínimos foram sendo elevados, de forma a filtrar os resultados, diminuindo a quantidade de regras geradas, tornando o resultado legível. Por fim, com valores mínimos de suporte e confiança fixados em 80% e 100%, respectivamente, foram geradas 88 regras; e com 90% de suporte e confiança mínimos, foram geradas apenas 12 regras. Destas, três foram selecionadas como exemplos, e podem ser vistas na tabela 4.2.

Para os conjuntos I_2 e I_3, a princípio usando as valores 1% e 50% de suporte e confiança mínimos, são geradas 108.723 regras (para I_2) e 108.544 (para I_3). Ao utilizar o mesmo critério de filtragem citado para o conjunto I_1, consistindo em subir os valores mínimos de suporte e confiança, ao atingir 90% em ambos, são selecionadas as mesmas

12 regras de I_1. Caso um dos atributos possa ser usado como uma indicação de classe, o CBA filtra as regras geradas, de forma a selecionar algumas que levem à classificação dos registros. Por exemplo, no I_3, o último atributo é o tempo. O CBA vai selecionar as regras que indiquem na conclusão qual é o tempo, dadas condições dos demais atributos. Nesta classificação, também devem ser indicados os valores mínimos de suporte e confiança, que foram fixados em 1% e 50%, respectivamente. Como são poucos registros, no CBA foi selecionada a opção que testa as regras selecionadas sobre todos os registros (poderia-se selecionar, por exemplo, apenas 30% da base de dados). Este teste gera uma matriz de confusão[1], que indica que as regras selecionadas foram capazes de classificar corretamente 100% dos registros. Algumas destas regras classificam apenas um registro, ou seja, são muito específicas (identificadas por valores de *CoverCount* e *SupCount* iguais a 1). Portanto, se estas regras realmente fossem usadas para classificação, elas teriam que ser descartadas, pois estariam contrariando a condição de generalização que um classificador deve seguir. Como neste caso este recurso foi utilizado só para filtrar a grande quantidade de regras, o resultado final ficou com todas as regras selecionadas pelo classificador: são 25 para o I_2 , e 30 regras para o I_3. Destas, foram selecionados alguns exemplos de cada, que podem ser vistas nas tabelas 4.3 e 4.4. Note que nas duas tabelas, o último exemplo selecionado classifica somente um registro (*CoverCount* e *SupCount* = 1). Note ainda que na tabela 4.3, o valor de suporte para as duas primeiras regras é baixo, mas pelos valores de *CoverCount* e *SupCount* (ambos iguais a 4) pode-se dizer que a regra é forte, pois são no máximo 5 registros para cada "classe" (são somente 4 ou 5 tempos para cada camundongo). Já para a tabela 4.4, a primeira regra também tem valores de *CoverCount* e *SupCount* iguais a 4, porém isso não indica tanta robustez, pois neste caso, para cada "classe" (tempo) há 7 ou 9 registros (correspondendo aos camundongos).

[1]Matriz que indica a quantidade de registros que foram atribuídos a cada classe, e a qual classe os registros pertencem. Ou seja, indica quantos registros foram corretamente e incorretamente classificados.

R38 = baixa
R26 = média
R2 = baixa
-> camundongo = 2
(9.302% 100.00% 4 4 9.302%)

R40 = baixa
R26 = média
R20 = média
-> camundongo = 2
(9.302% 100.00% 4 4 9.302%)

R13 = alta
R8 = média
-> camundongo = 4
(6.977% 100.00% 3 3 6.977%)

R11 = alta
-> camundongo = 8
(2.326% 100.00% 1 1 2.326%)

Tabela 4.3: Regras para o conjunto I_2

R30 = média
R9 = baixa
R6 = baixa
R1 = baixa
-> tempo = 5
(9.302% 100.00% 4 4 9.302%)

R28 = média
R4 = baixa
-> tempo = 3
(6.977% 100.00% 3 3 6.977%)

R30 = baixa
R16 = alta
-> tempo = 2
(2.326% 100.00% 1 1 2.326%)

Tabela 4.4: Regras para o conjunto I_3

Do ponto de vista computacional, foi interessante a extração de regras de associação. Porém, do ponto de vista biológico, a interpretação das regras geradas se mostrou bastante difícil. Primeiro, pela grande quantidade de regras; depois, mesmo selecionando apenas algumas com alto suporte e confiança, as regras não indicam causalidade, apenas indicam fatos que ocorreram simultaneamente. Além disso, não se tem a informação exata de quais proteínas correspondem a cada uma das seções de reatividade, representadas por R_1 a R_{40}, há somente a estimativa de seus pesos moleculares. Outro fator que levou a decisão de não utilizar este método com as outras bases de dados foi que neste teste, haviam apenas 40 atributos, e os resultados já são de difícil interpretação; se fosse executado para a base com 600 atributos, a quantidade de regras aumentaria absurdamente, a legibilidade ficaria ainda mais prejudicada, e a interpretação, praticamente inviabilizada.

4.2 Segundo experimento

Neste experimento, foram utilizados os dados brutos dos dois extratos das quimeras de medula óssea e das quimeras de fígado fetal, sem usar as médias de reatividade. Na seção anterior foram testados alguns métodos; nesta seção, serão utilizados os métodos que demonstraram bons resultados. Mais especificamente, serão mostrados nesta seção histogramas e resultados de agrupamentos feitos pelo *k-means*.

A primeira etapa consistiu em observar as regenerações dos repertórios tanto para o extrato de fígado quanto para o extrato de músculo, comparando os dois extratos para cada camundongo. Nas figuras 4.12, 4.13, 4.14, e 4.15 foram selecionados alguns exemplos de camundongos que tiveram boa regeneração de seus repertórios, respectivamente: camundongos 6 e 9, de quimera de medula óssea (que já tinham sido selecionadas na seção anterior); e camundongos 3 e 4, de quimera de fígado fetal. Observe que na figura 4.15, as reatividades do camundongo 4 nos tempos 1 e 4 não estão sobrepostas, porém a assinatura foi mantida (vide os pico e vales coincidindo nas seções de reatividade).

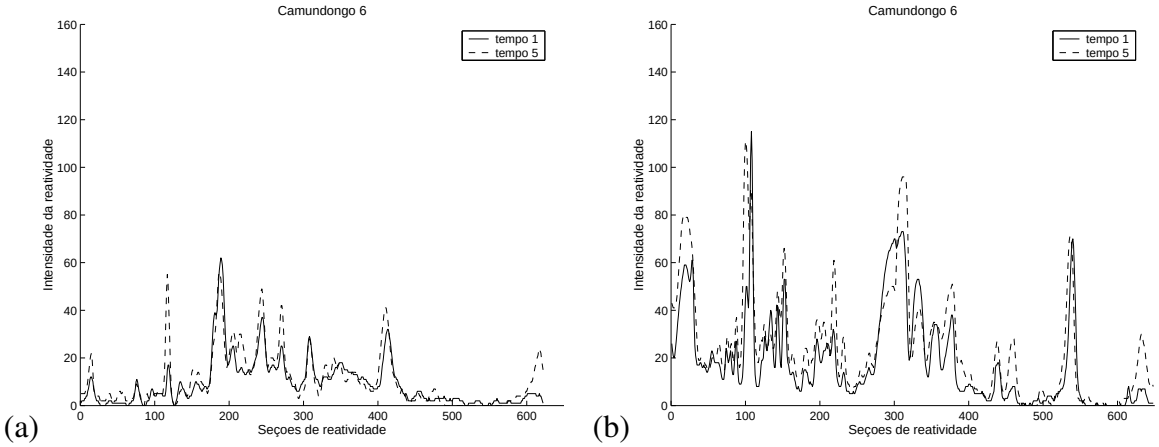


Figura 4.12: Histogramas de extratos de fígado (a) e músculo (b), camundongo 6, quimera de medula óssea.

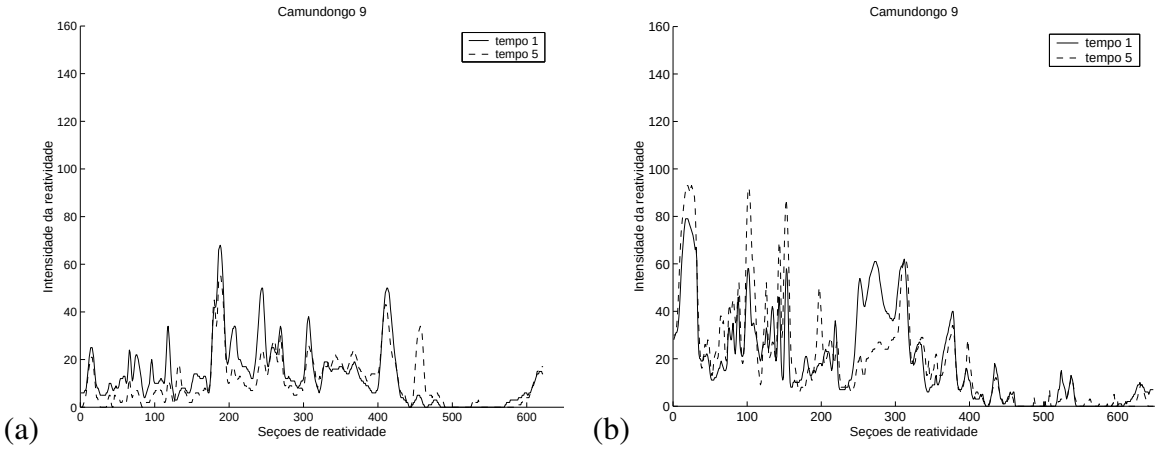


Figura 4.13: Histogramas de extratos de fígado (a) e músculo (b), camundongo 9, quimera de medula óssea.

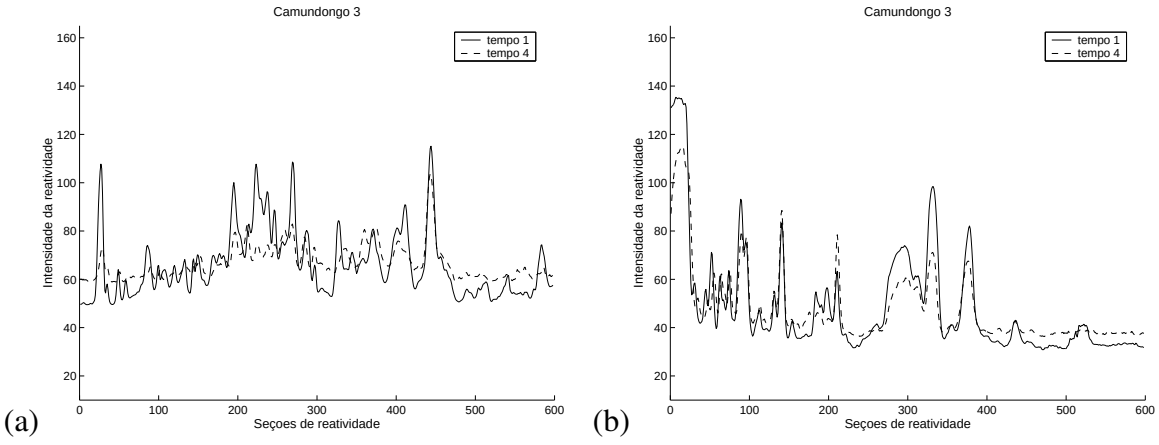


Figura 4.14: Histogramas de extratos de fígado (a) e músculo (b), camundongo 3, quimera de fígado fetal.

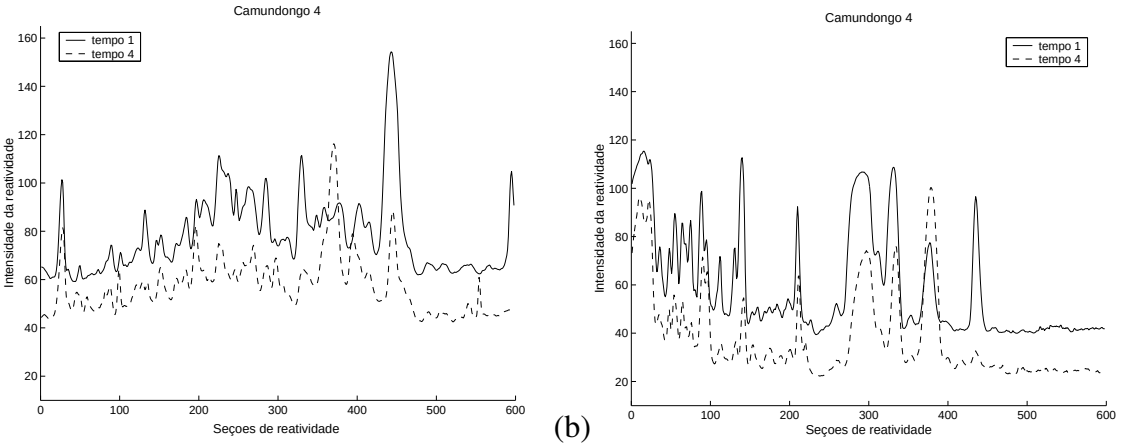


Figura 4.15: Histogramas de extratos de fígado (a) e músculo (b), camundongo 4, quimera de fígado fetal.

A próxima etapa foi fazer a PCA sobre as matrizes, a fim de diminuir a quantidade de atributos para aplicação posterior do *k-means*. Na PCA, foi indicado que utilizando apenas as duas primeiras componentes principais, para os dados quimera de medula óssea, cerca de 60% a 70% da informação é mantida. Para os dados de quimera de fígado fetal, cerca de 80% da informação é mantida. Já se a matriz for normalizada antes, a quantidade de informação mantida para os dados quimera de medula óssea cai cerca de 20%, e para os dados de quimera de fígado fetal, sobe cerca de 5%. Estes valores se encontram na tabela 4.5.

	Quimeras de Medula Óssea	Quimeras de Fígado Fetal
Extrato de Fígado	70%	80%
Extrato de Músculo	60%	80%
Extrato de Fígado - Dados Normalizados	50%	85%
Extrato de Músculo - Dados Normalizados	40%	85%

Tabela 4.5: Quantidade aproximada de informação mantida na utilização das duas primeiras componentes principais. Valores obtidos pela variância medida nas matrizes de autovalores.

Os gráficos gerados pelo *k-means* continuam com objetivo de comparar o primeiro e último dias, porém de forma um pouco diferente. Agora, em um único gráfico estão os tempos 1 e 5, para quimera de medula óssea (figuras 4.16 e 4.17), e os tempos 1 e 4 para quimera de fígado fetal (figuras 4.18 e 4.19). O tempo 1 está representado como figuras

abertas, e o último dia (tempo 5 para quimera de medula óssea, e tempo 4 para quimera de fígado fetal), como figuras fechadas. O objetivo é indicar se um determinado indivíduo permaneceu no mesmo grupo, ou mudou de grupo após o experimento. Observe nas figuras 4.16 e 4.17 que os camundongos 6 e 9 de quimera de medula óssea (representados por $\bigtriangledown$ e $\hexagram$, respectivamente, e cujos histogramas foram selecionados por demonstrarem boa regeneração), em ambos os tempos se encontram no grupo maior. Observe também que no tempo 5 eles estão mais próximos, isso fica claro principalmente na figura 4.17, onde todos os camundongos, exceto um, no tempo 5 se encontram em um único grupo.

Já as figuras 4.18 e 4.19 mostram os resultados do *k-means* para as quimeras de fígado fetal. Observe que no histograma da figura 4.14 as reatividades dos tempos 1 e 4 do camundongo 3 (representado por $\bigcirc$) estão bastante superpostas, e nos resultados do *k-means* para os dois extratos, ele permanece no mesmo grupo, o grupo central. Já o camundongo 4 ($\star$) tem o comportamento um pouco diferente, e isso pode ser visto também pelo fato de que ele troca de grupo nos resultados do *k-means* dos dois extratos.

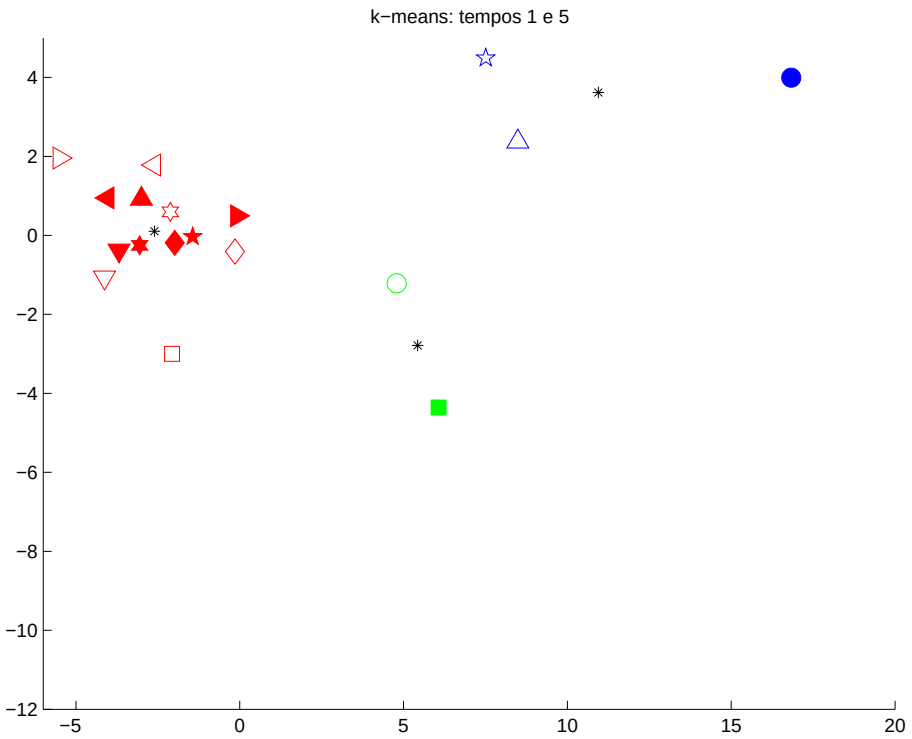


Figura 4.16: *k-means* para extrato de fígado, quimera de medula óssea.

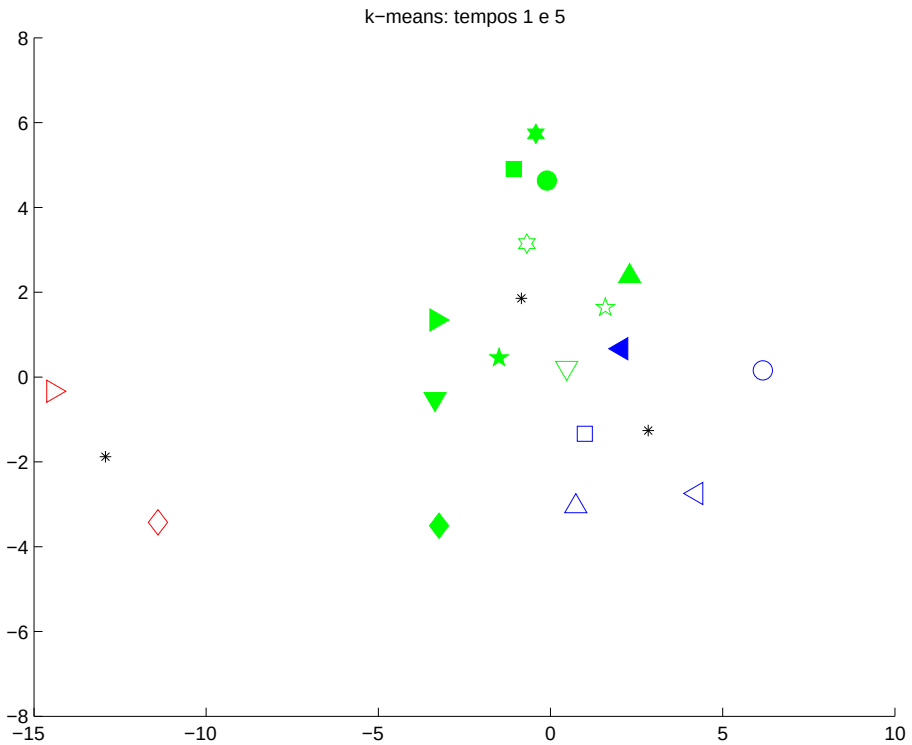


Figura 4.17: *k-means* para extrato de músculo, quimera de medula óssea.

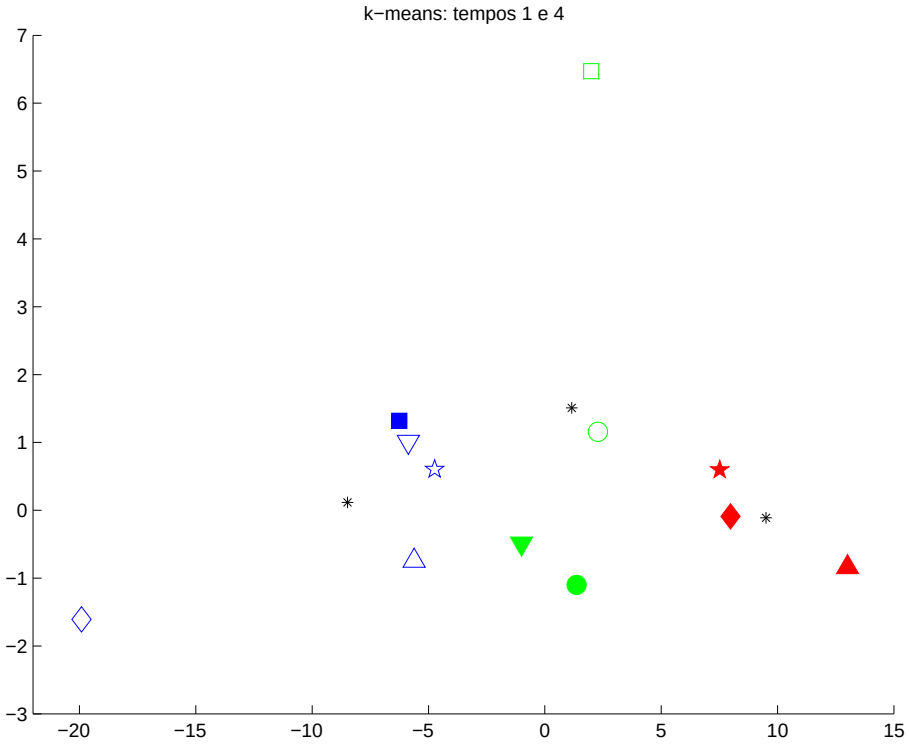


Figura 4.18: *k-means* para extrato de fígado, quimera de fígado fetal.

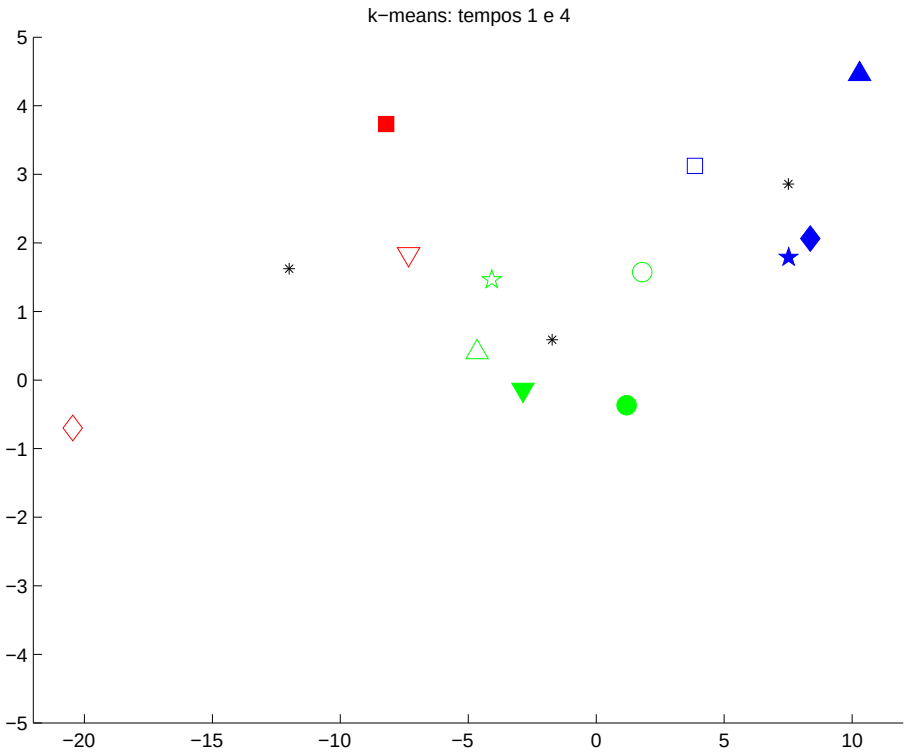


Figura 4.19: *k-means* para extrato de músculo, quimera de fígado fetal.

Capítulo 5

Resultados e discussão com dados de *Microarrays*

Neste capítulo, serão mostrados os resultados da aplicação de alguns métodos descritos no capítulo 3, sobre os dados do experimento descrito na seção 2.3.3. Conforme dito na seção 2.4, o grupo de especialistas que criou a técnica de estimação de diversidade utilizando *microarrays* [4] percebeu que precisava fazer uma análise mais detalhada sobre o método sugerido. A princípio, esta análise, descrita nesta tese, procurou estimar os erros associados ao método sugerido de contagem do número de *hits*. Em seguida, investigamos qual outra abordagem poderia ser utilizada no lugar desta contagem, e quais seriam os fatores que estariam influenciando nestas estimativas de diversidade. Toda esta informação foi utilizada, então, para criação de um modelo computacional capaz de simular os dados experimentais. Grande parte dos resultados e da discussão presentes neste capítulo foram publicados em dois pôsteres [48, 49], e fazem parte de um artigo sendo preparado para submissão.

Como descrito na seção 2.3.3 do Capítulo 2, a equipe de Biólogos que realizou os experimentos de *microarrays* gerou dezenas de baterias de testes, algumas contendo somente *Standards*, outras contendo *Standards* e *Samples* em diferentes quantidades. No presente trabalho, escolhemos três conjuntos de dados para trabalhar: dois contendo 4 *Standards*, sendo eles de diversidades: 4^0, 4^5, 4^{10}, e 4^{15}, que serão chamados de *Conjunto*

de Dados 1 e *2*; e o outro contendo 2 *Standards*, sendo eles de diversidades: 4^0 e 4^{15}, e 3 *Samples*, de camundongos[1] WT, QM, e $J_H^{-/-}$, que será chamado de *Conjunto de Dados* 3. Estes conjuntos foram selecionados pois são uma parte dos que foram usados no artigo [4]. Além disso, o Conjunto de Dados 1, com mais diversidades conhecidas, foi utilizado para estudar as propriedades das distribuições e para calibrar o modelo criado. O Conjunto de Dados 2 foi usado para comparar *Standards* de diferentes experimentos, quando confrontado com o Conjunto de Dados 1. Já o Conjunto de Dados 3, além de participar da comparação entre *Standards* e na discussão sobre número de *hits*, foi usado também para testar o modelo, pois possui amostras reais, cujas diversidades do repertório de linfócitos já foram estimadas. Ainda segundo o artigo [4], estima-se que um QM tenha cerca de 20% a 40% da diversidade de um WT, enquanto que a diversidade em um $J_H^{-/-}$ estima-se que seja quase nenhuma.

5.1 Número de *Hits*

Como descrito na seção 2.3.3, a técnica original usa o número de *hits* de cada *Standard* para criar a Curva *Standard*. Esta análise do número de *hits* requer que o ruído de fundo seja estimado, para que o ponto de corte seja escolhido. Se o gráfico da distribuição cumulativa de freqüências (CDF) da intensidade do sinal por *probe* for observado (veja figura 5.1), pode-se notar duas questões importantes:

1. O formato das curvas da CDF são diferentes dependendo da diversidade do *Standard* (em cinza), e diferem também entre *Standards* e *Samples* (vermelha e azul).

2. Dependendo do ponto de corte escolhido, o resultado da estimação da diversidade (D) é completamente diferente.

Nota-se então que estas diferenças entre as curvas fazem com que as estimativas de número de *hits* não sejam robustas. Para justificar este argumento, considere dois pontos

[1] WT significa *Wild Type*, que é o camundongo normal. Os outros dois tipos citados (QM: *Quasi-monoclonal* e $J_H^{-/-}$) são mutações específicas de camundongos, feitas para estudar o sistema imunológico, já que interferem diretamente na diversidade dos repertórios de linfócitos.

de corte, 50 e 100, como na figura 5.1. Para cada um dos pontos de corte, é construída sua respectiva Curva *Standard*, e os valores estimados de Diversidade (D) são obtidos através da interpolação linear da Curva *Standard,* conforme a técnica descrita na seção *2.3.3*. Para o ponto de corte 50, os *Samples* parecem ter diversidade muito maior, por exemplo o WT (vermelho) tem quase tanta diversidade (3×10^8) quanto o *Standard* de maior intensidade ($4^{15} \sim 10^9$). Se o ponto de corte fosse 100, o valor para o WT já não seria tão próximo (5×10^6). Portanto, as estimativas de diversidade não são únicas para cada conjunto de dados.

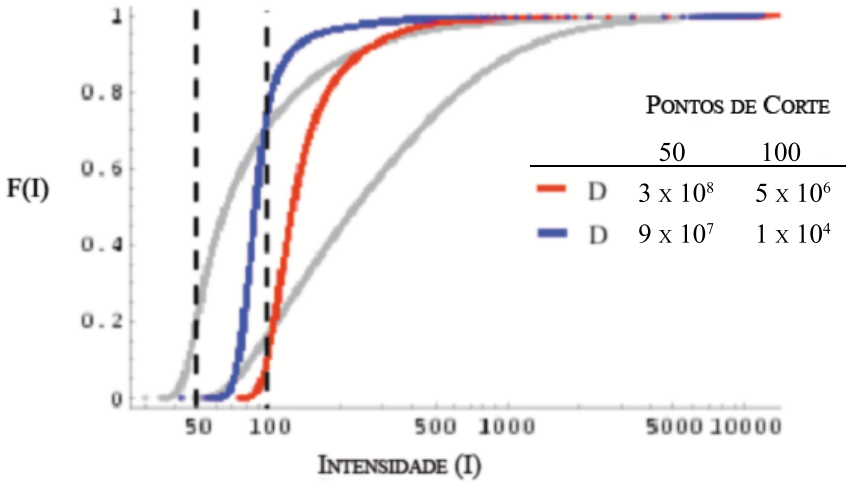


Figura 5.1: CDF da intensidade do sinal por *probe*, para os *Standards* (em cinza) e *Samples* (vermelha é o WT e azul, $J_H^{-/-}$) do Conjunto de Dados 3. Diferentes pontos de corte resultam em diferentes estimativas de diversidade (D).

Isso resulta em duas perguntas: Qual ou quais outras propriedades dos dados poderiam ser usadas em vez do número de hits? E qual é a origem destas diferenças entre as curvas de CDF? Estas duas questões serão estudadas nas próximas seções.

5.2 Análise estatítica de *Standards*

Para responder a primeira pergunta, qual outra propriedade dos dados poderia ser usada para criar a Curva *Standard*, foram investigadas outras propriedades estatísticas destes dados. No artigo [4], os autores afirmam que compararam uma série de *Standards* feitos em diferentes experimentos, e concluíram que era necessário criar um novo *Standard* para

cada experimento. Portanto, serão usados na presente análise os Conjuntos de Dados 1 e 2, para avaliar se algumas de suas propriedades têm comportamento semelhante em diferentes conjuntos de dados.

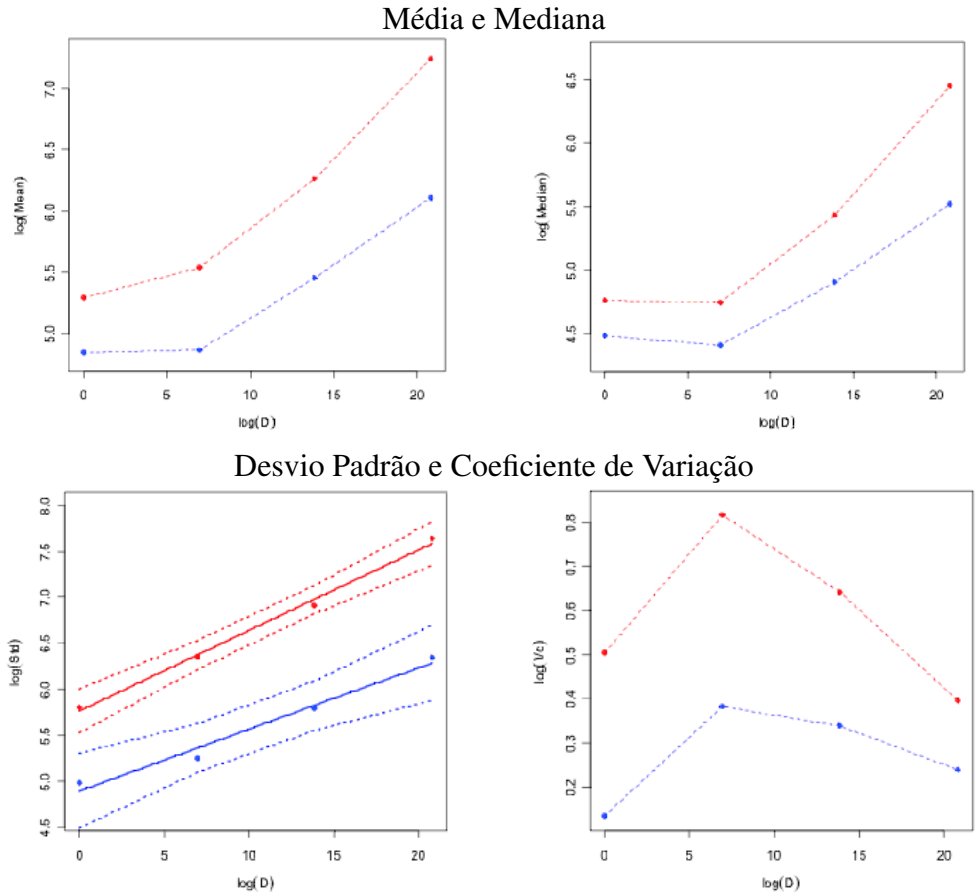


Figura 5.2: Agumas propriedades que acompanham a diversidade. Todos os gráficos estão em escala logarítmica. Para o desvio padrão, foi feita regressão linear (linha contínua) com intervalos de confiança de 95% (linhas pontilhadas). Conjunto de Dados 1 em vermelho e 2 em azul.

Na figura 5.2 estão algumas propriedades dos Conjuntos de Dados 1 e 2, em vermelho e azul, respectivamente. Nota-se que conforme a diversidade aumenta, a média, a mediana e o desvio padrão também aumentam (em escala logarítmica). Em especial, o desvio padrão aumenta linearmente (em escala logarítmica), o que indica que este poderia ser usado como medida para criação da Curva *Standard*. Na figura 5.2, no desvio padrão a linha contínua indica a regressão linear em escala logarítmica, e as linhas pontilhadas são o intervalo de confiança de 95%. O valor de R para os Conjuntos de Dados 1 e 2 são, respectivamente, $0,9929$ e $0,9651$.

O problema de usar o desvio padrão como medida é que voltamos à questão de que

o formato da distribuição da CDF da intensidade das Samples é muito diferente da dos *Standards*. Além disso, compactar a informação de muitos dados em um só número (400.000 pontos de reatividade de um *GeneChip* representados por um único valor), leva à perda informação. Pode-se usar, então, o próprio formato das curvas de CDF como medida para analisar os dados. Por elas, têm-se medidas visuais claras da diferença entre amostras. Portanto, em vez de usar uma Curva *Standard* baseada em um valor de cada *GeneChip*, vamos criar um modelo computacional que simule estas curvas de CDF. Assim, podemos usar toda a informação disponível.

Na figura 5.3, é possível inspecionar visualmente as curvas de CDF de *Standards* e *Samples*, e notar a diferença no formato das curvas. Nesta figura, encontram-se as curvas de *Standard* dos Conjuntos de Dados 1 e 3, e as diferenças destes *Standards* para as *Samples* do Conjunto de Dados 3. O valor da diversidade da primeira curva de ambos *Standards* é 4^0 , e da última diversidade é 4^{15}. Note que estas duas curvas têm formato bem diferente, em ambos conjuntos de dados. Porém, note que o formato da curva de diversidade 4^0 de um conjunto de dados é bem parecido com o do outro, o que acontece também para a curva da diversidade 4^{15}. Isso nos indica que há uma propriedade que faz com que as curvas mudem conforme a diversidade cresce, e que adicionar um valor de ruído de fundo ao Conjunto de Dados 3 poderia fazer com que suas curvas *Standard* ficassem sobrepostas às curvas *Standard* equivalentes do Conjunto de Dados 1. Estas informações serão utilizadas no modelo computacional descrito na seção 5.4.

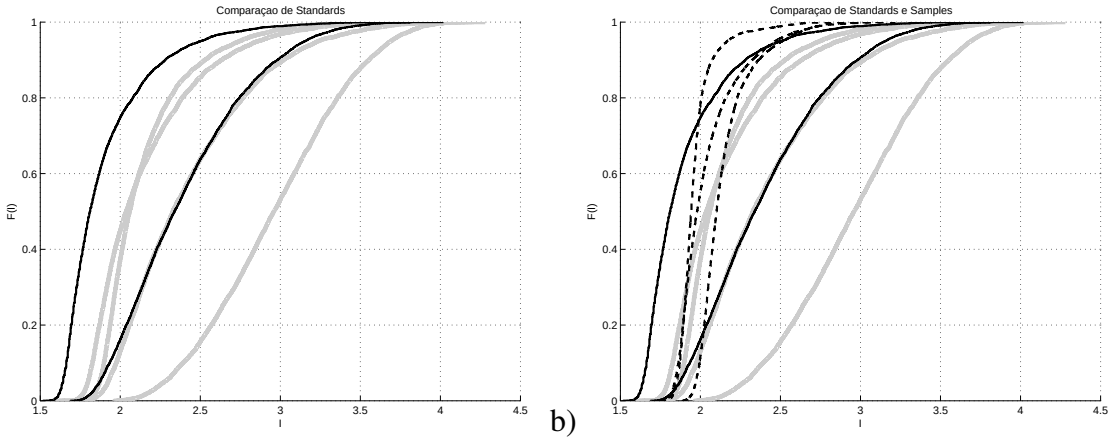


Figura 5.3: Comparação dos *Standards* Conjunto de Dados 1 (em cinza) com o Conjunto de Dados 3 (em preto; linhas contínuas são *Standards*, e linhas tracejadas são *Samples*).

Antes de falar sobre o modelo, vamos responder àquela outra questão sobre qual é a origem das diferenças entre as amostras, pois estas respostas também ajudarão na construção do modelo.

5.3 Natureza das seqüências

Observando como as curvas de CDF têm formatos bem diferentes comparando *Samples* e *Standards*, nos voltamos para estudar em mais detalhes como a metodologia experimental indica a criação e utilização das seqüências das amostras. Pode-se observar na tabela 5.1 que há diferenças significativas entre as seqüências.

	Samples	*Standards*
Tamanho da Seqüência	de 50 a 200	18
Distribuição de Tamanhos	Sim	Não
Níveis de Biotina	$\sim$10/seqüência	1/seqüência
Natureza química	RNA	DNA
Independência	?	Alguma dependência

Tabela 5.1: Diferenças entre a preparação das seqüências dos *Samples* e dos *Standards*.

As seqüências dos *Standards*, como são construídas sinteticamente nucleotídeo por nucleotídeo, têm sempre tamanho 18. As seqüências dos *Samples* são obtidas de amostras reais e ”cortadas” em pedaços, e a técnica utilizada para tal não tem tanto controle sobre os pontos de corte, portanto, sobre o tamanho destes pedaços. Isso faz com que as *Samples* tenham uma distribuição do tamanho das seqüências bem diferente das *Standards*: as seqüências cortadas têm entre 50 e 200 nucleotídeos. Para demostrar que o tamanho das seqüências faz diferença na energia de ligação, e, consequentemente, na medição da intensidade da hibridização de um *microarray*, foi utilizado um pacote de *softwares* chamado de UNAFold (UNAFold v3.2, by Nick Markham & Michael Zuker, © 2006 Rensselaer Polytechnic Institute, Troy, NY, EUA [50]). Este pacote, entre outras funcionalidades, faz o cálculo da energia de hibridização entre duas seqüências. Foram geradas várias baterias de seqüências de nucleotídeos aleatoriamente, onde determinamos apenas o tamanho das sequências, para calcular a energia de hibridização entre elas. Na figura 5.4, cada ponto corresponde à média dos valores de energia calculados para 10

baterias de testes. Em cada bateria de teste foram geradas 100 seqüências aleatorias (T - *targets*) para cada *probe* (P). Na diversidade 4^0 há apenas uma probe, e na diversidade 4^5 há 1024 probes. Os tamanhos escolhidos foram:

- *probe* de tamanho 10 e *target* de tamanho 10 (na legenda, P10 T10): para representar um tamanho bastante pequeno, como um limite inferior.
- *probe* de tamanho 25 e *targets* de tamanho 18 e 50 (na legenda, P25 T18 e P25 T50, respectivamente): para representar o tamanho das *probes* de um *GeneChip* (25) e o tamanho das seqüências de *Standards* (18) e *Samples* (de 50 a 200).
- *probe* de tamanho 50 e *target* de tamanho 50 (na legenda, P50 T50): para representar um tamanho grande, como um limite superior. Este é o tamanho máximo suportado pelo *software* UNAFold.

Note que, na figura, a média calculada da energia de ligação está em escala logarítmica, e que a diferença destes valores é bastante significativa.

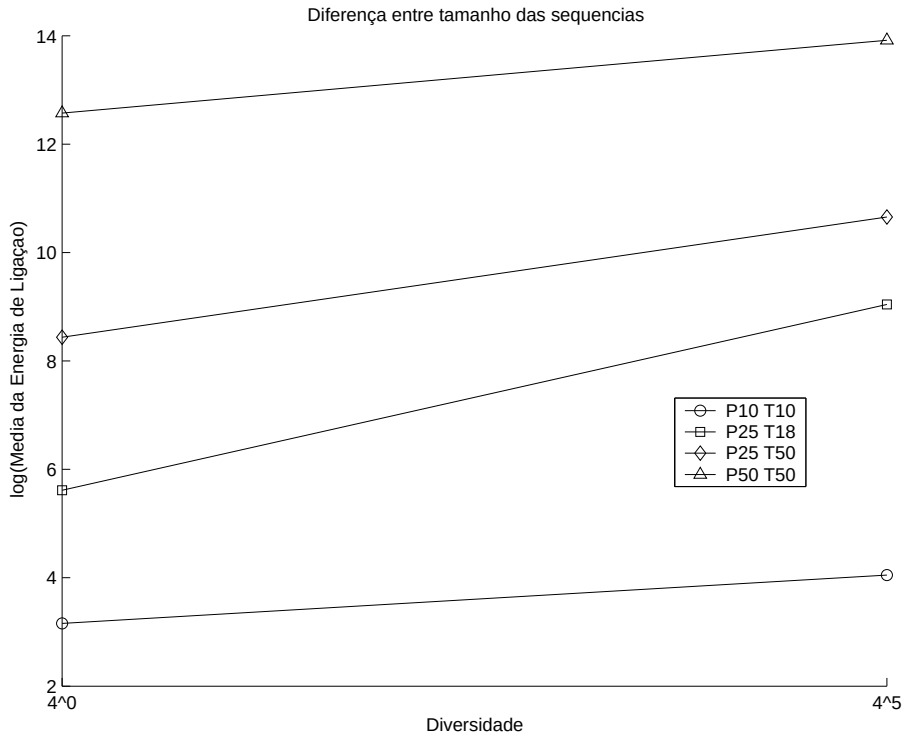


Figura 5.4: Resultados do UNAFold. Na legenda, P10 é *probe* de tamanho 10, e T10 é *target* de tamanho 10, e assim sucessivamente para os demais tamanhos.

Quanto aos níveis de biotina mencionados na tabela 5.1, essa quantidade maior nas *Samples* pode influenciar na leitura da intensidade das reações do *GeneChip*, indicando reatividade maior para as *Samples*, dificultando sua comparação com os *Standards*. E a natureza química das seqüências - DNA ou RNA - também influencia no cálculo da energia de ligação entre seqüências. Inclusive, para o uso do *software* UNAFold é necessário indicar se as seqüências testadas são de DNA ou de RNA [50], pois os parâmetros utilizados para o cálculo das energias de hibridização são diferentes. No caso da figura 5.4, utilizamos seqüências de DNA.

Sobre a indepedência das seqüências mencionada também na tabela 5.1, a princípio supõe-se que as *Samples* são independentes, mas isso pode não se confirmar, pois depende da amostra coletada. E as *Standards* são de certa forma dependentes, pois derivam de uma única seqüência. Esta informação afeta também a construção do modelo, pois para tal, é preciso assumir distribuições de probabilidades para vários parâmetros.

Portanto, considerando tudo o que está mencionado na tabela 5.1, conclui-se que a diferença na preparação das seqüências afeta a intensidade do sinal lido de um GeneChip de forma sistemática. Uma forma de lidar com isso é alterar o *design* do modelo experimental. Outra forma é trabalhar sobre o modelo computacional de forma que ele tenha parâmetros livres a serem alterados para cada conjunto experimental. Falaremos sobre isso na próxima seção.

5.4 Modelo Computacional de Simulação dos Dados Experimentais

O modelo computacional deve levar em conta os aspectos citados nas seções anteriores, e gerar dados que simulem os dados reais. Se o modelo for capaz de descrever como a forma da curva de distribuição de intensidade do sinal depende da diversidade, e se for possível interpretar os parâmetros biologicamente, o ajuste dos parâmetros do modelo pode ajudar na identificação de como estas diferenças mencionadas no capítulo anterior estão afetando os resultados.

Tentamos várias abordagens para criar este modelo, por exemplo, usando o *software* UNAFold para cálculo das energias de ligação, assumindo que o ruído de fundo é distribuído, aceitando um fator de saturação, assumindo que a probabilidade de uma *probe* se ligar a uma *target* é distribuída, ou assumindo que a afinidade média de uma *probe* por uma *target* também é distribuída. A única combinação que se mostrou compatível com os dados experimentais foi assumindo que o ruído de fundo, a probabilidade de cada *probe* se ligar a uma *target*, e a afinidade de ligação são distribuídas. Tivemos que supor, ainda, uma relação entre a probabilidade de uma *probe* se ligar a uma *target* e a média de afinidade (energia) de ligação de uma *probe*. Esta relação implica que *probes* que se ligam a muitas *targets* têm, em média, uma afinidade de ligação menor do que as *probes* que se ligam a poucas *targets*, ou seja, quanto mais degenerada for uma probe, menor será sua energia de ligação média.

O modelo computacional pode ser descrito pela seguinte equação:

$$I_i = b_i + \sum_{j=1}^{D} K_i(p_i)\,\frac{1}{D}\,X_{i,j}$$

onde b_i ~Lognormal(1, 1.15)*φ+ρ ,

$X_{i,j}$ ~Bernoulli(p_i) , e

$K_i = \dfrac{K_0}{p_i^n}$

I_i representa a intensidade do sinal lido do *GeneChip* para uma probe i, e D são as diversidades. b_i é o ruído de fundo (*background*), que consideramos como uma distribuição Lognornal que representa a ligação não-específica, independente da seqüência.

K_i é a afinidade de ligação, e tem influência direta de p_i, que está representando a degenerecência da *probe*, a probabilidade de ligação. O formato de p_i é de uma distribuição Beta(1.3, 1.3), no qual seus valores máximo e mínino (p_u e p_l) foram deixados como parâmetros a serem escolhidos dependendo do conjunto de dados (mas que variam de 10^{-4} a 10^{-9}).

Após estimar os valores da intensidade I_i, um fator de saturação tem que ser aplicado, representando a diferença entre a intensidade real do sinal emitido pela probe e o sinal que os equipamentos são capazes de ler. I_{max} é o valor máximo da intensidade obtida para

a maior diversidade, e será usado para limitar os demais sinais. O valor da intensidade com saturação para cada probe Is_i é calculado pela fórmula:

$$Is_i = \frac{(I_{max} * I_i)}{(s + I_i)}$$

onde $s = I_{max} - (I_{max} * 0,1)$

O modelo foi aplicado sobre os dados dos Conjuntos 1 e 3. Em especial, o Conjunto de dados 1 foi usado para calibrar o modelo, e só após a calibragem o modelo foi aplicado ao Conjunto de dados 3. Note que na figura 5.5, o modelo se ajusta adequadamente à forma das curvas de CDF dos dados originais, e prevê a dependência já observada da mediana e do desvio padrão em relação à diversidade. Em alguns casos testados, o modelo é capaz também de prever a média. Para ajustar o modelo (já calibrado usando o Conjunto de dados 1) às *Standards* do Conjunto de dados 3 (veja figura 5.6), fixamos os parâmetros que deveriam ser os mesmos (p_u, p_l e n, que medem a afinidade de ligação), e recalibramos ajustando os parâmetros que dependem do experimento (φ e ρ de b_i, e K_0). Para ajustar o modelo para as *Samples* do Conjunto de dados 3, alteramos p_u, p_l e n, bem como o ruído de fundo (φ e ρ), e fixamos o valor de K_0, que consideramos ser um fator que muda de experimento para experimento, mas que não muda dentro de um mesmo conjunto de dados. Veja os valores dos parâmetros na tabela 5.2.

	Conjunto de Dados 1	Conjunto de Dados 2	Conjunto de Dados 2
Parâmetros	*Standards*	*Standards*	*Samples*
p_u	10^{-4}	10^{-4}	10^{-4}
p_l	$10^{-8,6}$	$10^{-8,6}$	10^{-8}
n	1,45	1,45	1,41
φ	20	10	3
ρ	50	33	75
K_0	1	0,17	0,17

Tabela 5.2: Parâmetros das simulações exibidas nas figuras 5.5 e 5.6.

Neste ponto, nos deparamos com a questão de associar propriamente os parâmetros do modelo às propriedades biofísicas do experimento, o que só poderá ser feito por especialistas biólogos. Ao fazer esta associação dos parâmetros, e ao observar quais deles são alterados para simular o comportamento de cada conjunto de dados, será possível

entender melhor quais são os fatores podem ser corrigidos no *design* experimental para que novos resultados tenham o mínimo de erro sistemático.

Mesmo sem este feedback dos especialistas biólogos na análise dos resultados das simulações, podemos indicar, baseando-se nos resultados obtidos nesta tese, as seguintes alterações no *design* experimental que poderiam minimizar o erro das estimativas de diversidade:

- Criar seqüências de *Standards* e *Samples* de mesmo tamanho, ou se não for possível controlar com precisão o tamanho das *Samples*, estudar qual é distribuição estatística se adequa às *Samples* e criar *Standards* com mesma distribuição estatística;
- Usar seqüências de mesma natureza química;
- Fazer mais *Standards* para cada conjunto de dados, pelo menos 4 para cada conjunto;
- Usar uma informação mais completa, como as curvas de CDF, para tirar conclusões sobre os dados experimentais, em vez de medidas que condensem o resultado de um *GeneChip* em um único número.

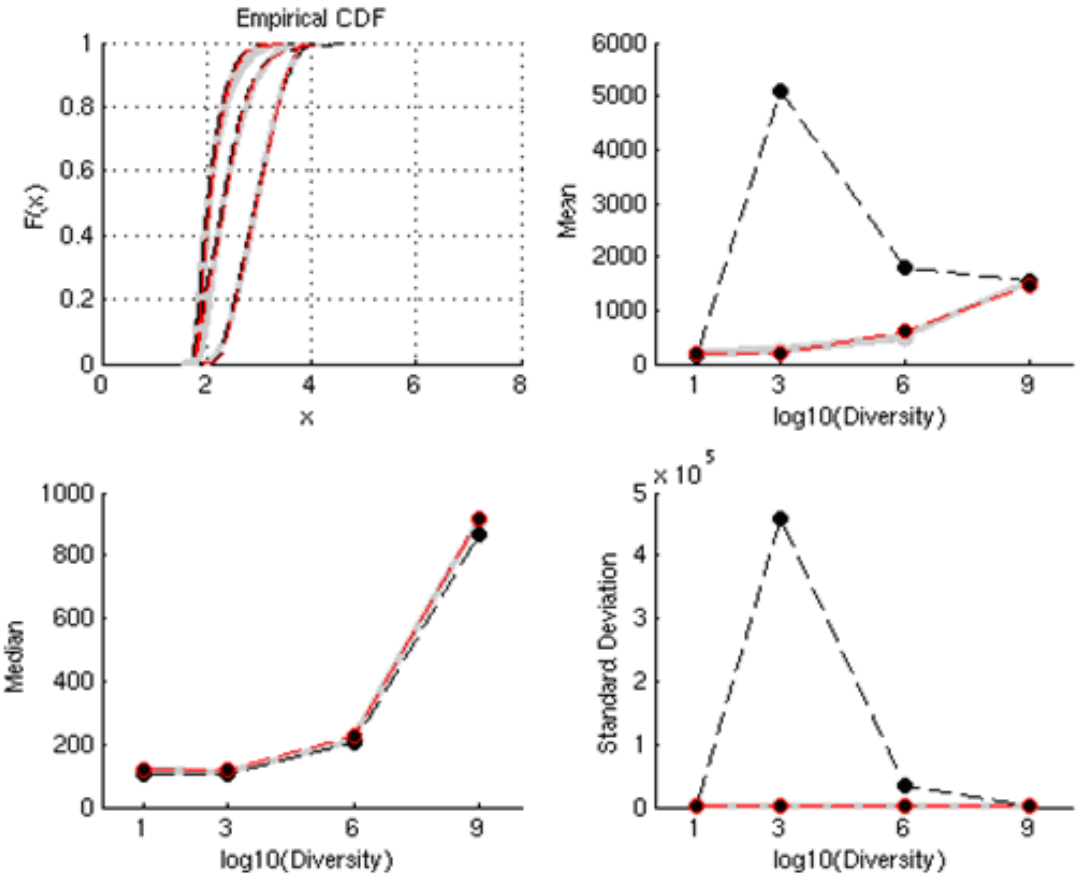
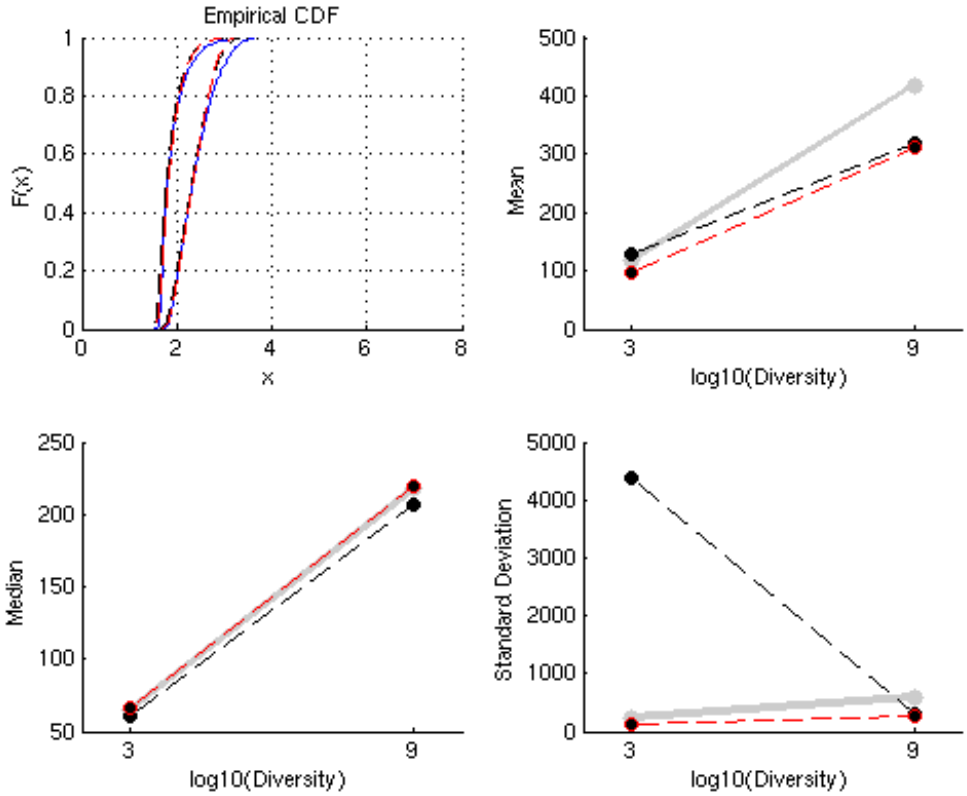


Figura 5.5: Resultados da simulação para o Conjunto de Dados 1. Em cinza estão os dados experimentais; em preto, o modelo sem o fator de saturação; em vermelho, o modelo com o fator de saturação.

Standards



Samples

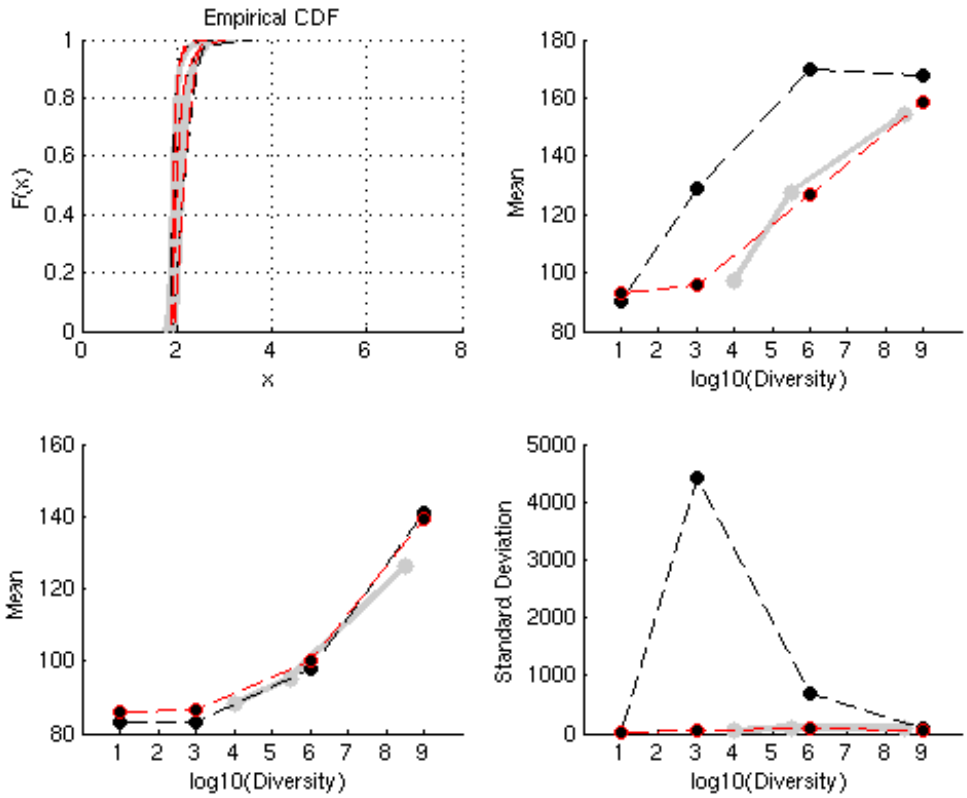


Figura 5.6: Resultados da simulação para o Conjunto de Dados 3. Em cinza estão os dados experimentais; em preto, o modelo sem o fator de saturação; em vermelho, o modelo com o fator de saturação.

Capítulo 6

Conclusão e trabalhos futuros

A bioinformática é uma parceria bastante interessante, tanto para os biólogos quanto para os especialistas em computação. Ambas as áreas estão constantemente alimentando uma e outra com novas idéias, inspirações, modelos, e aplicações. No presente trabalho, o foco da biologia é a imunologia, mais especificamente, sobre experimentos que analisam o sistema imunológico de maneira global, utilizando técnicas inovadoras como Imunoblots e *microarrays*. Já pelo lado da computação, o foco é criar metodologias de análise que tragam informações novas e relevantes sobre os resultados obtidos anteriormente, e sugestões de novos experimentos que possam aproveitar melhor as técnicas descritas. Para tal, a KDD e a mineração de dados serviram como base para estipular passos a seguir e escolher métodos adequados para analisar os dados experimentais.

Esta tese teve como objeto de aplicação dois experimentos. Um destes trata da regeneração de repertórios de anticorpos de camundongos, utilizando a técnica de Imunoblot. Este experimento consistiu em submeter camundongos a radiações que destroem células que originam o seu repertório de anticorpos, e, após tratá-los, observar se o repertório é capaz de se regenerar ou não [3].

Os histogramas traçados, e resultados dos algoritmos de agrupamento ajudaram a demonstrar que, na maior parte dos casos, esta recuperação é bem sucedida, e estas técnicas destacaram-se por uma boa aceitação pelos especialistas biólogos.

Os resultados dos algoritmos de agrupamento *k-means*, FCM e SOM são muito semelhantes, o que implica em uma alta probabilidade de que os resultados estejam

corretos. Além disso, podemos indicar o uso de apenas um deles: se houver a necessidade de utilizar o mais simples por consumir menor poder computacional, sugerimos o *k-means*; se o objetivo é ter a sugestão dos agrupamentos mais isenta de tendências, sugerimos o SOM, pois não é preciso indicar *a priori* a quantidade de agrupamentos.

Já a extração de regras de associação fuzzy esbarrou no problema de gerar regras demais, o que dificultou a interpretação destas pelos especialistas, e portanto não houve uma boa aceitação desta técnica.

Para esta primeira aplicação, a metodologia para análise de dados vindos de Imunoblots facilitou o entendimento e a visualização dos dados. E uma sugestão que podemos fazer sobre alteração do *design* experimental é de analisar, de uma só vez, mais indivíduos, para que as análises estatísticas sejam mais robustas.

Encontram-se abaixo sugestões de trabalhos futuros que gostaríamos de realizar sobre esta metodologia:

- É interessante aplicar esta metodologia sobre outras bases de dados de Imunoblots, que tenham sido geradas através de diferentes experimentos. Há pelo menos dois trabalhos que estão disponíveis para tal: na referência [53], foram realizados experimentos utilizando a doença de chagas como fator de perturbação do repertório de camundongos; e na referência [52] foram analisados repertórios de anticorpos de cães naturalmente infectados com *L. chagasi*.

- Mais uma técnica poderia ser acrescentada a esta metodologia: a análise de fatores do PCA, como a que foi feita na referência [31]. O objetivo desta análise é descobrir, através da observação dos pesos da PCA, quais foram as faixas de reatividade que mais influenciaram na construção das componentes da PCA. Se algumas faixas de reatividade tiverem sempre um peso significativo na construção, por exemplo, da primeira e da segunda componentes principais da PCA, seria possível sugerir uma volta aos experimentos de bancada para descobrir qual ou quais anticorpos fazem parte desta faixa de reatividade.

O segundo objeto de aplicação consiste em dados vindos de um experimento que procura estimar a diversidade do repertório de linfócitos de um organismo, utilizando a técnica de *microarrays* [4]. A metodologia computacional aplicada contou com uma análise dos erros associados a esta técnica, e com a modelagem computacional para simulação dos dados experimentais.

Sobre a análise dos erros, concluímos que estes vêm da natureza muito diferente dos dados, e que estas diferenças fazem com que as estimativas baseadas em contagem de número de *hits* não sejam robustas.

No lugar destas, sugerimos o uso de forma da função de distribuição cumulativa (CDF) para estimação da diversidade, pois assim podemos utilizar informações preciosas que seriam perdidas se fossem condensadas em um só número. Para tal, criamos um modelo computacional que simula os dados experimentais, criando curvas de CDF compatíveis com os dados originais.

Utilizando os resultados da análise dos erros e uma avaliação preliminar da associação de parâmetros do modelo à características biofísicas do experimento, sugerimos ainda algumas alterações no *design* experimental, como utilizar Standards e Samples de mesmo tamanho e mesma natureza química; e fazer pelo menos quatro Standards para cada conjunto de dados.

Sobre esta metodologia, gostaríamos de realizar os seguintes trabalhos futuros:

- Utilizar algoritmos genéticos ou alguma outra técnica estatística para ajustar com maior precisão os parâmetros do modelo computacional. Para tal, seria preciso estipular uma medida para avaliação do quanto uma simulação se encaixou com os dados experimentais. A medida poderia ser por exemplo, um teste estatístico como Kolmogorov-Smirnov ou Critério de Informação de Akaike [54].

- Em conjunto com o grupo de especialistas biólogos, será possível associar propriamente os parâmetros do modelo com as propriedades biofísicas dos experimentos de *microarrays*, de forma a permitir uma análise mais detalhada dos fatores que influenciam no erro das estimativas de diversidade.

- O grupo de pesquisa que realizou os experimentos citados em [4] fez muitas outras baterias de testes, a maioria ainda não faz parte de nenhuma publicação. Pretendemos usar o modelo computacional sobre estes outros conjuntos de dados, para avaliar quais parâmetros mudarão, e se será possível fazer as estimativas de diversidades nestas novas amostras.

Resumindo, as contribuições deste trabalho são: as metodologias criadas para análise de dados de imunologia, que permitem uma análise mais quantitativa e precisa dos resultados obtidos anteriormente; as ferramentas implementadas em MatLab para realização das análises previstas pelas metodologias, que estão disponíveis para a comunidade científica; a avaliação da contribuição dos métodos de mineração de dados nestas aplicações; e a descoberta de informações relevantes sobre repertórios de linfócitos/anticorpos de organismos.

Por fim, podemos dizer que a maior contribuição deste trabalho foi o estabelecimento de interfaces entre grupos de pesquisa experimental de biologia e o grupo de ciência da computação. A troca de experiências e a apresentação para os especialistas biólogos de métodos relativamente comuns em computação, mas inovadores na imunologia, foi extremamente recompensadora e fértil. A atual convergência entre áreas de pesquisa distintas tende a aumentar, e espera-se que as contribuições apresentadas nesta tese ajudem a fomentar o crescimento da Bioinformática.

Referências Bibliográficas

[1] CASTRO, L. N.; VON ZUBEN, F. J. **Artificial Immune Systems: Part II - A Survey of Applications.** Technical Report, DCA-RT 02/00. Unicamp, SP, Brazil. 65 p., 2000. Disponível em http://www.dca.fee.unicamp.br/~vonzuben/research/rt_dca.html . Acesso em 2005.

[2] HAURY, M.; GRANDIEN, A.; SUNBLAD, A.; COUTINHO, A; NOBREGA, A. **Global Analysis of Antibody Repertoires. 1. An Immunoblot Method for the Quantitative Screening of a Large Number of Reactivities.** Scand J Immunol, 39, p 79-87, 1994.

[3] NOBREGA, A.; STRANSKY, B.; NICOLAS, N.; COUTINHO, A. **Regeneration of Natural Antibody Repertoire After Massive Ablation of Lymphoid System: Robust Selection Mechanisms Preserve Antigen Binding Specificities.** The Journal of Immunology, vol 169, p 2971-2978, 2002.

[4] OGLE, B.M.; CASCALHO, M.; JOAO, C.M.; TAYLOR, W.; WEST, L.J.; PLATT; J.L. **Direct measurement of lymphocyte receptor diversity.** Nucleic Acids Research 31(22):e139. 2003.

[5] JOÃO, C.M.; OGLE, B.M.; GAY-RUBENSTEIN, C.; PLATT, J.L.; CASCALHO, M. **B cell-dependent TCR diversification.** Journal of Immunology 172(8):4709-4716. 2004.

[6] ABBAS, A. K.; LICHMAN, A. H.; POBER, J. S. **Imunologia celular e molecular.** Segunda edição, Revinter, 1998.

[7] AVRAMEAS, S. **Natural autoantibodies: from 'horror autotoxicus' to 'gnothi seauton'.** Immunol Today 12, no. 5, p 154-9, 1991.

[8] HOOIJKAAS, H.; BENNER, R.; PLEASANTS, J. R.; WOSTMANN, B. S. **Isotypes and specificities of immunoglobulins produced by germ-free mice fed chemically defined ultrafiltered "antigen-free" diet.** Eur J Immunol 14, no. 12, p 1127-30, 1984.

[9] DIGHIERO, G.; GUILBERT, B.; AVRAMEAS, S. **Naturally occurring antibodies against nine common antigens in humans sera. II. High incidence of monoclonal Ig exhibiting antibody activity against actin and tubulin and sharing antibody specificities with natural antibodies.** J Immunol 128, no. 6, p 2788-92, 1982.

[10] SOUROUJON, M.; WHITE-SHARF, M. E.; ANDRESCHWARTZ, J.; GEFTER, M. L.; SCHWARTZ, R. S. **Preferential autoantibody reactivity of the preimmune B cell repertoire in normal mice.** J Immunol 140, no. 12, p 4173-9, 1988.

[11] GAUDIN, E.; ROSADO, M.; FREITAS, A. **Antigen dose a key factor for B cell selection.** 11th congress of Immunology. Scand. J. Immunol., 54 (suppl. 1): C14, 2001.

[12] ANDERSSON, A.; FORSGREN, S.; SODERSTROM, A.; HOLMBERG, D. **Monoclonal, natural antibodies prevent development of diabetes in the non-obese diabetic (NOD) mouse.** J Autoimmun 4, no. 5, p 733-42, 1991.

[13] BOES, M.; PRODEUS, A. P.; SCHIMIDT, T.; CARROLL, M. C.; CHEN, J. **A critical role of natural immunoglobulin M in immediate defense against systemic bacterial infection.** J Exp Med 188, no. 12, p 2381-6, 1998.

[14] KAZATCHKINE, M. D.; KAVERI, S. V. **Immunomodulation of autoimmune and inflammatory diseases with intravenous immune globulin.** N Engl J Med 345, no. 10, p 747-55, 2001.

[15] OCHSENBEIN, A. F.; ZINKERNAGEL, R. M. **Natural antibodies and complement link innate and acquired immunity.** Immunol Today 21, no. 12, p 624-30, 2000.

[16] NOBREGA, A.; HAURY M.; GRANDIEN A.; MALENCHERE E.; SUNBLAD A.; COUTINHO A. **Global analysis of antibody repertoires. II. Evidence for specificity, self-selection and the immunological homunculus of antibodies in normal serum.** Eur. J. Immunol. 11, p 2851, 1993.

[17] MOUTHON, L.; NOBREGA, A.; NICOLAS, N.; KAVERI, S.; BARREAU, C.; COUTINHO, A.; KAZATCHKINE, M. **Invariance and restriction toward a limited set of self antigens characterize neonatal IgM antibody repertoires and prevail in autoreactive repertoires of healthy adults.** PNAS, 92, p 3839, 1995.

[18] BERNEMAN, A.; TERNYNCK, T.; AVRAMEAS, S. **Natural mouse IgG reacts with self antigens including molecules involved in the immune response.** Eur J Immunol 22, no. 3, p 625-33, 1992.

[19] **NIH Image software.** Disponível em http://rsb.info.nih.gov/nih-image/Default.html

[20] **Igor Pro Software.** Wavemetrics. Mais informações em www.wavemetrics.com/products/igorpro/igorpro.htm . Acesso em 2005.

[21] **Affymetrix Inc.**, Santa Clara, CA, USA. http://www.affymetrix.com . Acesso em 2008.

[22] FAYYAD, U.; PIATETSKY-SHAPIRO, G.; Smyth, P. **From Data Mining to Knowledge Discovery in Databases**. AI Magazine 17(3): 37-54, Fall 1996.

[23] FAYYAD, U. **Mining Databases: Towards Algorithms for Knowledge Discovery.** Bulletin of the IEEE Computer Society Technical Comittee on Data Engeneering, 21, vol 1, p 39 - 48, 1998.

[24] HAN, J.; KAMBER, M. **Data Mining : Concepts and Techniques.** Morgan Kaufmann, 2000.

[25] **KD Nuggets**. http://www.kdnuggets.com/ . Acesso em 2005.

[26] CHAKRABARTI S. **Data mining for hypertext: A tutorial survey.** SIGKDD Explorations, ACM SIGKDD, vol 1, issue 2, p 1, 2000.

[27] ZAÏANE O. R.; XIN M.; HAN J. **Discovering Web Access Patterns and Trends by Applying OLAP and Data Mining Technology on Web Logs.** Proc. ADL'98 (Advances in Digital Libraries), 1998.

[28] LITTLE R.J.A.; RUBIN, D.B. **Statistical Analysis with Missing Data.** Segunda edição, J. Wiley & Sons, New York, 2002.

[29] HERRING, A.H.; IBRAHIM, J.G.; LIPSITZ, S.R. **Non-ignorable missing covariate data in survival analysis a case-study of an int. breast cancer study group trial.** Journal of the Royal Statistical Society, 53(2):293–310, 2004.

[30] RENCHER, A. C. **Methods of Multivariate Analysis - Wiley Series in Probability and Statistics.** Segunda Edição, Wiley-Interscience, 2002

[31] GUIYEDI, V.; CHANSEAUD, Y.; FESEL, C.; SNOUNOU, G.; ROUSSELLE, J.C.; LIM, P.; KOKO, J.; NAMANE, A.; CAZENAVE, P.A.; KOMBILA, M.; PIED, S. **Self-Reactivities to the Non-Erythroid Alpha Spectrin Correlate with Cerebral Malaria in Gabonese Children.** PLoS ONE, 2(4): e389. 2007.

[32] BARKOW, S.; BLEULER, S.; PRELIC, A.; ZIMMERMANN, P.; ZITZLER, E. **BicAT: a biclustering analysis toolbox**. Bioinformatics 22: 1282-1283. 2006.

[33] RAWAT, A.; DENG, Y. **Novel implementation of conditional co-regulation by graph theory to derive co-expressed genes from microarray data.** BMC Bioinformatics, v.9(Suppl 9):S7. 2008.

[34] MITCHEL, T. M. **Machine Learning.** McGraw-Hill, 1997.

[35] WITTEN, I. H.; FRANK, E. **Data Mining: Practical machine learning tools and techniques**. Segunda Edição, Morgan Kaufmann, 2005.

[36] BENSMAIL, H.; GOLEK, J.; MOODY, M.M.; SEMMES, J.O.; HAOUDI, A. **A novel approach for clustering proteomics data using Bayesian fast Fourier transform.** Bioinformatics, Vol. 21, No. 10, pp. 2210-2224. 2005.

[37] BEZDEK, J. C. **Pattern Recognition with Fuzzy Objective Function Algorithms.** New York, Plenum, 1981.

[38] KOHONEN, T. **Self-Organizing Maps.** Series in Information Sciences, Vol. 30. Segunda edição, Springer, Heidelberg, 1997.

[39] KOHONEN, T. **Intro to SOM by Teuvo Kohonen.** Disponível em http://www.cis.hut.fi/projects/somtoolbox/theory/somalgorithm.shtml . Acesso em 2008.

[40] BURNSIDE, E.; DAVIS, J.; SANTOS COSTA, V.; DUTRA, I. C.; KAHN, C. E. ; FINE, J.; PAGE, D. **Knowledge Discovery from Structured Mammography Reports using Inductive Logic Programming**, AMIA Annu Symp Proc. 2005, p. 96–100. 2005.

[41] AGRAWAL, R.; IMIELINSKI, T.; SWAMI, A. **Mining association rules between sets of items in large databases.** SIGMOD, pgs 207-216, 1993.

[42] SRIKANT, R.; AGRAWAL, R. **Mining quantitative association rules in large relational tables.** ACM SIGMOD Int'l Conference on Management of Data, p. 1-12, 1996.

[43] KUOK, C.; FU, A.; WONG, M. **Mining fuzzy association rules in databases.** ACM SIGMOD Record, v. 27 n. 1, p. 41-46, 1998.

[44] LIU, B; HSU, W.; MA, Y. **Integrating Classification and Association Rule Mining.** Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98), pp. 80-86, 1998. *Software* **CBA - Classification Based Algorithm** disponível para *download* em http://www.comp.nus.edu.sg/~dm2/index.html. Acesso em 2008.

[45] FAYYAD, U.; GRINSTEIN, G.; WIERSE, A. **Information Visualization in Data Mining and Knowledge Discovery.** Morgan Kaufmann, 2001.

[46] FERRARI, L. I.; STRANSKY, B.; NICOLE, N.; COUTINHO, A.; NOBREGA, A.; CARVALHO, L. A. V. **Mining Relevant Infomation from Natural Antibody Repertoire Regeneration Experiments.** Proceedings of the IV Brazilian Symposium on Mathematical and Computational Biology / I International Symposium on Mathematical and Computational Biology. Vol 1. E-papers Serviços Editoriais Ltda., 2004.

[47] **MatLab - The Language of Tecnical Computing.** MathWorks. Mais informações em www.mathworks.com . Acesso em 2004.

[48] FERRARI, L. I.; GARDNER, R.; SOUSA, A.E.; OGLE, B.; PLATT, J.L.; CASCALHO, M.; CARNEIRO J. **Precision and accuracy of antigen-receptor diversity estimates using microarray technology.** 13^{th} International Congress of Immunology, Rio de Janeiro, Brasil. 2007

[49] GARDNER, R.; FERRARI, L. I.; CARNEIRO, J. **Quantitative Determination of Lymphocyte Repertoire: Analysis of antigen-receptor diversity estimation based on gene-chip hybridization technology.** 1^{st} Mediterranean Workshop on Clinical Immunology, Évora, Portugal. 2006

[50] MARKHAM, N.R.; ZUKER, M. **DINAMelt web server for nucleic acid melting prediction.** Nucleic Acids Research, Vol. 33, Web Server issue, W577-W581, 2005. Disponível em <http://www.bioinfo.rpi.edu/applications/hybrid/>, Acessado em 2007.

[51] WU, Z.; IRIZARRY, R. A. **Stochastic Models Inspired by Hybridization Theory for Short Oligonucleotide Arrays.** Journal of Computational Biology. Vol 12(6), p.882-893, 2005.

[52] VALE, A. M. **Identificação de antígenos em diferentes espécies do gênero *Leishmania* do Novo Mundo utilizando anticorpos presentes no soro de cães infectados com *Leishmania (leishmania) chagasi*.** Dissertação (Mestrado). Universidade Federal de Minas Gerais, Belo Horizonte, 2004.

[53] BESSA, M. C. **Estudo dos Anticorpos Naturais Auto-reativos na Doença de Chagas Experimental.** Dissertação (Mestrado). Instituto Oswaldo Cruz, Rio de Janeiro, 2004.

[54] AKAIKE, H. **A new look at the statistical model identification**. IEEE Trans. Automat. Contr. v. 19, n. 6, p. 716-23. 1974.