Embrapa

As ações de pesquisa em genética e genômica, que antes se encontravam limitadas às "bancadas", já não existem mais sem que sejam complementadas com procedimentos computacionais, pois os dados obtidos nos laboratórios devem ser identificados, organizados, armazenados e interpretados para que possam ser recuperados, apresentados e, ainda, utilizados como um novo atributo de informação.

Se antes parecia estranho a junção de áreas com conceitos tão distantes, tais como, computação e genômica, atualmente, pode não ser totalmente compreendido, mas pela proximidade e "mistura" das aplicações desses conceitos.

O livro, **TACG - Talking About Computing and Genomics - Vol. I: Modelos e Métodos Computacionais em Bioinformática**, apresenta, em seus dois primeiros capítulos, discussões e aplicações de conceitos que tratam da mistura da computação com a genômica e, nos três capítulos finais, traz aplicações específicas em problemas técnico-científicos que estão em estudo por grupos de pesquisa em diversas partes do mundo.

Os autores e editores desta obra entregam para comunidade científica resultados de pesquisas que deixam claro como a genômica "clássica" pode beneficiar-se de métodos computacionais e matemáticos para avançar em direção à fronteira do conhecimento e, da mesma forma, apontam para onde a computação deve voltar suas pesquisas no sentido da investigação científica.

Ministério da
Agricultura, Pecuária
e Abastecimento

GOVERNO FEDERAL
BRASIL
PAÍS RICO É PAÍS SEM POBREZA

TACG - Talking About Computing and Genomics - Vol. I

Embrapa

Talking About Computing and Genomics TACG

Vol. I

Modelos e Métodos Computacionais em Bioinformática

Editores Técnicos
Wagner Arbex
Natália Florêncio Martins
Marta Fonseca Martins

Embrapa

Empresa Brasileira de Pesquisa Agropecuária
Embrapa Gado de Leite
Embrapa Recursos Genéticos e Biotecnologia
Ministério da Agricultura, Pecuária e Abastecimento

Talking About Computing and Genomic
TACG

Vol. I

Modelos e Métodos Computacionais em Bioinformática

Editores Técnicos
Wagner Arbex
Natália Florêncio Martins
Marta Fonseca Martins

Embrapa
Brasília, DF
2014

Exemplares desta publicação podem ser adquiridos na:

Embrapa Gado de Leite
Rua Eugênio do Nascimento, 610 – Dom Bosco
CEP 36038-330 – Juiz de Fora, MG
Fone (32) 3311-7459
Fax (32) 3311-7424
www.embrapa.br
www.embrapa.br/fale-conosco/sac

Embrapa Recursos Genéticos e Biotecnologia
Parque Estação Biológica, PqEB, Av. W5 Norte (final)
CEP 70770-917 – Brasília, DF
Caixa Postal 02372
Fone: (61) 3448-4700
Fax: (61) 3340-3624

Unidade responsável pelo conteúdo
Embrapa Gado de Leite
Embrapa Recursos Genéticos e Biotecnologia

Unidade responsável pela edição
Embrapa Gado de Leite

Supervisão editorial
Wagner Arbex

Revisão de texto
Wagner Arbex

Normalização bibliográfica
Inês Maria Rodrigues

Projeto gráfico e editoração eletrônica
Wagner Arbex e Pedro Antonio de Castro Bittencourt

Capa
Adriana Barros Guimarães

1a. edição
1a. impressão (2014): 1.000 exemplares

Dados Internacionais de Catalogação na Publicação (CIP)

Talking About Computing and Genomics (TACG): vol. I: Modelos e Métodos Computacionais em Bioinformática / editores técnicos, Wagner Arbex, Natália Florêncio Martins e Marta Fonseca Martins. Brasília, DF : Embrapa, 2014.
201 p. : il. color. ; 14,8 cm x 21 cm.

ISBN 978-85-7035-382-5

1. Bioinformática. 2. Modelagem computacional. 3. Método computacional. 4. Genômica. I. Arbex, Wagner. II. Embrapa Gado de Leite.

CDD 570.285

Autores

André dos Santos Gonzaga
Acadêmico de Ciência da Computação da Universidade de São Paulo, São Paulo, SP.

Benilton de Sá Carvalho
Estatístico, Doutor em Bioestatística, pesquisador associado da Universidade Estadual de Campinas, Campinas, SP.

Carlos Cristiano Hasenclever Borges
Engenheiro Civil, Doutor em Engenharia Civil, professor da Universidade Federal de Juiz de Fora, Juiz de Fora, MG.

Eduardo Raul Hruschka
Engenheiro Civil, Doutor em Computação de Alto Desempenho, cientista-chefe da Big Data, São Paulo, SP.

Fabiana Barichello Mokry
Médica-veterinária, Doutora em Genética e Melhoramento Animal, pesquisadora recém-doutora da Universidade Federal de São Carlos, São Carlos, SP.

Fabrízzio Condé de Oliveira
Matemático, doutorando em Modelagem Computacional da Universidade Federal de Juiz de Fora, Juiz de Fora, MG.

Fabyano Fonseca e Silva
Zootecnista, Doutor em Estatística e Experimentação Agropecuária, professor da Universidade Federal de Viçosa, Viçosa, MG.

Fernanda Nascimento Almeida
Biomédica, Doutora em Bioinformática pela Universidade de São Paulo, São Paulo, SP.

Francisco Pereira Lobo
Biólogo, Doutor em Bioinformática, pesquisador da Embrapa Informática Agropecuária, Campinas, SP.

Luciana Correia de Almeida Regitano
Médica-veterinária, Doutora em Genética e Melhoramento, pesquisadora da Embrapa Pecuária Sudeste, São Carlos, SP.

Marcos Vinícius Gualberto Barbosa da Silva
Zoootecnista, Doutor em Genética e Melhoramento, pesquisador da Embrapa Gado de Leite, Juiz de Fora, MG.

Maurício de Alvarenga Mudadu
Biólogo, Doutor em Bioinformática, pesquisador da Embrapa Pecuária Sudeste, São Carlos, SP.

Raul Fonseca Neto
Engenheiro Civil, Doutor em Engenharia de Sistemas e Computação, professor da Universidade Federal de Juiz de Fora, Juiz de Fora, MG.

Roberto Hiroshi Higa,
Engenheiro Elétrico, Doutor em Engenharia Elétrica,

pesquisador da Embrapa Informática Agropecuária, Campinas, SP.

Saul de Castro Leite
Cientista da Computação, Doutor em Modelagem Computacional, professor da Universidade Federal de Juiz de Fora, Juiz de Fora, MG.

Saulo Moraes Villela
Cientista da Computação, Doutor em Engenharia de Sistemas e Computação, professor da Universidade Federal de Juiz de Fora, Juiz de Fora, MG.

Wagner Arbex
Matemático, Doutor em Engenharia de Sistemas e Computação, analista da Embrapa Gado de Leite, Juiz de Fora, MG.

Lista de ilustrações

Lista de tabelas

Sumário

Apresentação

PRODUZIR ALIMENTOS, fibras e energia a custos e preços decrescentes, expandir continuamente as aquisições de produtos e serviços urbano-industriais e, ainda, gerar excedentes que permitam exportações e que se traduzam em moeda forte para aquisição de insumos importados, todas essas são tarefas que precisam ser cumpridas pelo setor agrícola para que uma nação encontre o desenvolvimento econômico. A imensa maioria dos países sucumbem nessa trajetória, pois executar essas tarefas não é algo trivial.

O Brasil é um raro exemplo, em todo o mundo, do cumprimento dessas tarefas, a partir dos anos oitenta do século passado, quando as transformações começaramm a se materializar. A existência de produtores empreendedores se somou à oferta de soluções tecnológicas geradas e adaptadas por pesquisadores brasileiros, traduzindo em inovação e em dinamismo velozes. O resultado é que o Brasil deixou de ser somente litorâneo, em que praticamente toda a vida da nação acontecia a até quinhentos km de distância do mar. Portanto, interiorizou a produção, ao tempo em que passou a ser formador de preços para diferentes commodities agrícolas.

Atualmente, vivemos uma rápida e definitiva mudança da base tecnológica produtiva. Uma nova onda de inovação. Isso é re-

sultado das transformações recentes na geração de conhecimento, que iniciam uma nova revolução produtiva. Visando registrar uma parte do conhecimento gerado em seus projetos de pesquisa, dois grupos se reuniram para produzir esta obra. Os projetos *"Modelos computacionais para estabelecimento de meios e procedimentos metodológicos para análise de dados em bioinformática – MCBio"* e *"Rede nacional para o desenvolvimento e adaptação de estratégicas genômicas inovadoras aplicadas ao melhoramento, conservação e produção animal – Rede Genômica Animal"* são desenvolvidos por profissionais comprometidos na disseminação do conhecimento, cujo esforço está materializado nesta obra, que preenche um vazio existente na organização sistêmica de aspectos relacionados a modelos computacionais e genômica. Por tudo isso, esta obra já nasce como marco referencial.

Paulo Martins
Chefe-Geral
Embrapa Gado de Leite

Prefácio

NAS ÚLTIMAS décadas, grandes avanços foram alcançados no uso de ferramentas e métodos da computação e da informática para a análise de dados moleculares, permitindo a solução de problemas específicos e criando e estimulando a geração de novas oportunidades de avanços científicos e tecnológicos com base nos dados gerados e nos resultados obtidos.

Inovações recentes nos métodos de sequenciamento e genotipagem de ácidos nucleicos ampliaram vertiginosamente a capacidade de geração de dados genômicos e proporcionaram, nos últimos anos, reduções colossais nos custos de sequenciamento. Tais mudanças levaram a capacidade de geração de dados a ultrapassar a capacidade de análise, interpretação e inferência de resultados e ações subsequentes. Este cenário lança novos desafios para as "Ciências da Vida", os quais, quando superados, poderão prover soluções importantes para, por exemplo, avanços na saúde e também na produção sustentável de alimentos mais nutritivos, seguros, e em quantidade adequada para acomodar as demandas geradas pela crescente população mundial.

Em sua essência, a bioinformática trata do manuseio de dados genômicos, seja para tratamento da estrutura e do armazenamento dos dados, seja para a análise e geração de conhecimento

em processos automáticos ou semi-automatizados. Conjuntos de dados genômicos disponíveis não podem ser considerados “tradicionais” e caracterizam-se por bases de dados da ordem de terabytes, muitas vezes fisicamente dispersas, geradas com tecnologias distintas e dados não categorizados, entre outros fatores.

O contexto atual da pesquisa científica moldou um novo paradigma, chamado por alguns de “ciência intensiva”, e por outros, “computação científica” e “e-Science”, o qual combina a disponibilidade de grandes massas de dados com recursos de computação de altíssimos porte e capacidade. Por serem cada vez mais acessíveis, tais recursos criam oportunidades para busca de soluções e respostas para questões complexas, multidisciplinares, que poderão revolucionar a maneira como produzimos alimentos, geramos energia e materiais com base em organismos vivos, e tratamos as doenças que acometem o ser humano.

Na pesquisa com o uso intensivo de dados, não basta o simples estabelecimento de hipóteses a serem testadas por meio da coleta/análise dirigida de dados. O aprendizado e a descoberta, gerados pela combinação de dados disponíveis e variados conhecimentos, são essenciais para formular e testar novas hipóteses. Experimentos concebidos nesse contexto devem tratar de problemas cada vez mais complexos com abordagens que estejam para além da tradicional, que envolvam colaboração e visão multidisciplinar. É essencial o compartilhamento e distribuição de dados heterogêneos de alta dimensionalidade, baseados em novos protocolos e processos de comunicação, em agendas de pesquisa dinâmicas que estejam à frente do que hoje se convencionou chamar “dilúvio de dados”.

A presente obra lança um olhar sobre esses desafios e oportunidades, apresentando e discutindo os meios para a aplicação da ciência intensiva em bioinformática e biologia computacional para produção de impactos em diversos campos da pesquisa e

da inovação. Os autores priorizam e discutem modelos computacionais e matemáticos que enfatizem desafios científicos sob um novo prisma de investigação e uma nova estratégia metodológica de observação. Como resultado, são apresentadas propostas de novas formas de análise que aumentem a nossa capacidade de geração de inovação a partir de tecnologias, produtos e serviços relevantes para os distintos setores da sociedade.

O conteúdo aqui apresentado será de extrema utilidade para a disseminação de informações e treinamento da futura geração de cientistas que terão a grande responsabilidade de bem posicionar o Brasil em vertentes de inovação cada vez mais dependentes de métodos e modelos computacionais em bioinformática. Cumprimentamos os editores e colaboradores por tornar realidade esta importante obra!

Maurício Antonio Lopes
Presidente
Empresa Brasileira de Pesquisa Agropecuária - Embrapa

1

Desafios e perspectivas da bioinformática

Benilton de Sá Carvalho
Wagner Arbex

O SETOR da bioinformática tem-se apresentado como de grande interesse para a comunidade científica do País. Apesar disso, este setor enfrenta uma série de desafios que dificultam o seu pleno desenvolvimento. A multidisciplinaridade intrínseca da área funciona simultaneamente como agente catalisador e obstáculo para esta esfera. Os envolvidos precisam ser capacitados em diferentes ramos da ciência, entre os quais destacam-se a estatística, computação e biologia. Este trabalho apresenta alguns destes desafios, tendo como principal foco a interação necessária para a modelagem, análise e interpretação de dados.

1.1 Introdução

Apesar da bioinformática ser uma área de pesquisa estabelecida há cerca de 30 anos, até hoje, se discute uma definição para essa área e, possivelmente, para qualquer proposta que se faça, sempre haverá um questionamento, um complemento ou uma alteração sugerida por algum interlocutor.

Entretanto, nada mais normal do que tal reação, pois, apesar de, praticamente, já ter atingido a “maturidade científica”, a bioinformática apresenta muitos e variados aspectos que, dependendo da forma como se percebe, podem ser tomados diferentes atributos para sua definição. Além disso, ainda existem outras áreas de estudo próximas, tal como, a biologia computacional, que dificultam ainda mais a definição de bioinformática.

Algumas referências marcam o trabalho da bioinformática, além de sua própria origem, tais como as suas atividades no Projeto Genome Humano, além das suas aplicações em diversos projetos genoma de diferentes espécies, quando foram utilizados sistemas de computação de grande aporte computacional no tratamento do volume de dados gerado por tais projetos.

Tais ações necessitavam de profissionais especializados na área e, nesse contexto, foi possível perceber o surgimento da figura de um novo profissional. Ou melhor, de um novo perfil de profissional que deveria entender o problema biológico, tratá-lo de forma a criar um modelo de representação do mesmo, implementar o modelo e, por fim, analisar o resultado concebido.

Era necessário um profissional que possuísse conhecimento suficiente de biologia molecular, para, no mínimo, entender o problema; matemática e probabilidade, de forma que fosse capaz de estabelecer um modelo de representação do problema; computação, possibilitando a implementação da solução correta e apropriada, sendo esta inicialmente conhecida ou não; e estatística, para que as soluções fossem comprovadamente confiáveis.

Predominantemente, a bioinformática atua sobre as moléculas e sequências biológicas – DNA, RNA e proteínas – e, ainda, a interação entre tais moléculas e processos celulares. Assim, por uma visão estrita, a bioinformática consiste no desenvolvimento e no uso de técnicas de computação científica, de modelagem matemática e computacional e de modelagem probabilística e estatística

para resolver problemas de biologia molecular, estabelecendo uma convergência tecnológica a partir do uso do conhecimento proveniente dessas áreas que a fundamentam.

Assim, não se deve ter a expectativa de que o bioinformata fosse capaz de atender a tantas exigências, com o nível de conhecimento e complexidade esperados. A bioinformática é multidisciplinar, o que normalmente provoca a especialização do profissional em determinada área do conhecimento ou, eventualmente, em um grupo de disciplinas correlacionadas. Além disso, pela natureza das atividades em bioinformática, promove-se o trabalho não só em equipes, mas entre equipes, devido à amplitude das áreas de conhecimento envolvidas, estabelecendo as redes de pesquisa.

No cenário brasileiro, como destaque de trabalhos de pesquisa em bioinformática, o sequenciamento da *Xylella fastidiosa* definiu um marco. O DNA desta bactéria, causadora de prejuízos significativos à cultura cítrica do País, foi completamente sequenciado em 1999 pelo Projeto Genoma Xylella, financiado pela FAPESP, concluído em novembro daquele ano.

Este projeto não teria sido concluído com sucesso se não fosse pela integração de disciplinas da biologia, ciência da computação, estatística, física e química. Isto permitiu que fenômemos biológicos associados às moléculas de DNA e RNA fossem melhor compreendidos e definiu uma nova vertente na ciência. O sucesso da bioinformática depende fortemente da integração dos diversos ramos da ciência e consequente reconhecimento da crescente multidisciplinaridade deste campo. Do bioinformata espera-se a capacidade de lidar com habilidade com pontos específicos da biologia molecular, delineamento de experimentos e programação, além da compreensão dos mecanismos de funcionamento dos próprios equipamentos de genômica e proteômica.

Com esta característica multidisciplinar cada vez mais evidente, a bioinformática no Brasil lida, cada vez mais frequentemente, com uma série de desafios. Para estes podem-se empregar estratégias já utilizadas em outros países e, assim, promover o efeito catalisador da integração de ramos distintos da ciência.

1.2 Experimentos com sequenciamento de nova geração

A conjunção de ferramentas analíticas, computacionais e de coleta e geração de dados biológicos tem afetado positivamente a ciência mundial. Os impactos tomaram proporções maiores à medida que volumes de dados gerados paralelamente aumentaram de modo bastante rápido. Este progresso é mais facilmente observado com os ciclos de renovação de equipamentos utilizados na coleta e produção de dados biológicos.

Um grande salto foi observado quando microarranjos de DNA foram popularizados e utilizados de forma mais frequente em estudos biomédicos. A possibilidade de observar milhares de alvos genômicos simultaneamente – como genes ou marcadores diversos –, revolucionou o modo com o qual investigadores lidam com os dados, visto que esta nova configuração permitiu a inferência de associações entre fatores genéticos e fenótipos de interesse de uma forma até então não considerada.

Atualmente, observa-se a constante renovação de equipamentos laboratoriais e os microarranjos vem sido substituídos por equipamentos de sequenciamento de nova geração. Esta opção, inicialmente associada a altíssimos custos, tem sido a mais comum em diversos grupos, não apenas brasileiros, mas também estrangeiros. Como consequência desta decisão, pesquisadores notam, mais evidentemente, a necessidade de terem à disposição

uma equipe de formação ampla e qualificada na geração, análise e interpretação de grandes bases de dados.

O processo de execução de um experimento usando sequenciamento de nova geração possui diversas fases. Ele inicia-se com a definição da hipótese biológica a ser verificada, passando pelo delineamento estatístico do experimento, preparação e sequenciamento de bibliotecas, análise e interpretação de dados, finalizando com validação dos resultados.

Cada um destes passos requer uma equipe coesa para que o estudo siga de forma eficaz. Eles também apresentam desafios que, se superados, afetarão todos os braços da ciência que estejam ligados, diretamente ou não, às tecnologias de sequenciamento. As próximas seções discutem alguns destes desafios em maiores detalhes.

1.3 Fatores estatísticos

Questões estatísticas são comumente associadas a pontos de alta complexidade em um experimento. Esta preocupação tem razões bem fundamentadas, pois dificuldades na comunicação com o estatístico responsável ou falhas no planejamento experimental podem afetar seriamente a condução e conclusão do estudo em questão.

Comunicação com o estatístico

No que se refere à geração e análise de dados e à interpretação de resultados, um ponto de grande relevância é a interação do pesquisador com o estatístico responsável. O quesito de maior preocupação é a compreensão, mesmo que leiga, da terminologia estatística e sua consequente tradução para um vocabulário que o pesquisador interessado possa compreender. A compreensão

destes conceitos facilitarão a execução de fases importantes: determinação do tamanho de amostra e planejamento experimental.

Compreender conceitos como hipóteses – nula e alternativa –, tipos de erro e *p-valor* são essenciais para que a comunicação com o estatístico flua sem maiores transtornos. Estas definições são, em geral, as que apresentam-se mais complexas para pesquisadores de outras áreas.

Em termos estatísticos, características de interesse de um dado processo são representadas matematicamente e, então, avaliadas de acordo com um modelo que descreva seu comportamento esperado de acordo com condições pré-determinadas. Por exemplo, em um estudo de diferenciação de expressão gênica entre dois grupos de indivíduos para um gene específico, a *hipótese nula*, habitualmente denotada por H_0, descreve o cenário no qual ambos os grupos possuem níveis comparáveis de expressão, ao passo que a *hipótese alternativa*, geralmente designada por H_1 ou H_A, representa o cenário no qual há diferenças significativas de expressão entre os grupos. A idéia de "hipóteses" foi inicialmente discutida por Fisher (1935) e, ali, a hipótese nula nunca pode ser provada, apenas rejeitada ou não.

Neste exemplo, o estatístico reinterpreta os cenários dados pelo experimentalista como a existência ou não de diferenças significativas entre as médias de expressão do gene em questão entre os dois grupos estudados. Assim, a pergunta de interesse inicialmente apresentada como:

> *para o gene em questão, existem diferenças de expressão entre os grupos A e B?*

é vista pelo estatístico como:

$$\begin{aligned} H_0 : &\quad \mu_A = \mu_B \\ H_1 : &\quad \mu_A \neq \mu_B, \end{aligned} \tag{1.1}$$

onde μ_A representa a expressão média do gene em questão para o grupo A e μ_B tem interpretação análoga, mas restrita ao grupo B. Observando a hipótese alternativa, o leitor observa que as diferenças podem ser tanto positivas quanto negativas. Neste caso, diz-se que o teste é bilateral.

Se o experimentalista, *a priori*, define que a questão biológica a ser respondida é:

> *para o gene em questão, a expressão gênica no grupo A é sistematicamente menor que aquela observada no grupo B?*

então o conjunto de hipóteses levadas para o teste é:

$$\begin{aligned} H_0 : &\quad \mu_A \geq \mu_B \\ H_1 : &\quad \mu_A < \mu_B. \end{aligned} \tag{1.2}$$

Uma outra questão biológica que pode ser respondida é:

> *para o gene em questão, a expressão gênica no grupo A é sistematicamente maior que aquela observada no grupo B?*

e, neste caso, o conjunto de hipóteses que será averiguado é:

$$\begin{aligned} H_0 : &\quad \mu_A \leq \mu_B \\ H_1 : &\quad \mu_A > \mu_B. \end{aligned} \tag{1.3}$$

Os testes descritos pelas Equações 1.2 e 1.3 são denominados unilaterais.

Dada a base estabelecida com as definições de hipóteses, o pesquisador tem condições de entender conceitos mais avançados, como Erro Tipo I e como Erro Tipo II. Os erros referem-se a enganos na tomada de decisão baseada na evidência apresentada pelos dados. O Erro Tipo I, representado por α, refere-se a rejeitar a hipótese nula, quando esta é verdadeira. O Erro Tipo II, denotado por β, refere-se à não-rejeição da hipótese nula, quando esta é falsa.

Isto é, na condição de a hipótese nula ser verdadeira, a estratégia empregada para o teste não deveria rejeitar H_0; caso isso ocorra, comete-se um erro do tipo I. Alternativamente, se H_0 é falsa, o teste deveria rejeitar a hipótese nula; quando isso não ocorre, comete-se um erro do tipo II (Tabela 1.1).

Tabela 1.1 – Erros Tipos I e II.

	Não Rejeitar H_0	Rejeitar H_0
H_0 Verdadeira	OK	Erro Tipo I (α)
H_0 Falsa	Erro Tipo II (β)	OK

Um outro conceito de difícil compreensão e interação com o estatístico é *p-valor*. O valor de significância, também conhecido como *valor-p* ou *p-valor*, refere-se ao menor valor requerido para α que causaria a rejeição da hipótese nula. Como o Erro Tipo I é especificado *a priori*, o *valor-p* é comumente utilizado para a decisão entre rejeitar ou não a hipótese nula para um dado experimento. Utilizando o exemplo apresentado pela Equação 1.1, se o pesquisador define α como 5%, então *p-valores* menores que esta referência causam a rejeição da hipótese nula.

Uma interpretação mais probabilística do *valor-p* parte do fato de que ele é determinado assumindo que H_0 é verdadeira.

Desta maneira, o *valor-p*, assumindo a veracidade de H_0, é a probabilidade de resultados pelo menos tão extremos quanto os obtidos no experimento corrente serem observados apenas como consequência do acaso.

Determinação do tamanho amostral

Assim como em outras áreas, um fator determinante nos experimentos que envolvem bioinformática é o tamanho amostral. Isto adiciona-se à lista de desafios com alta prioridade por conta de uma série de fatores, incluindo:

- necessidade de interagir com outros profissionais: nesta fase, é necessário o contato com não apenas o analista de dados, mas também com os profissionais envolvidos na coleta, geração e armazenamento de dados. A característica multidisciplinar da bioinformática exige a quebra do paradigma, ainda comum no País, de que a colaboração entre diferentes áreas é opcional;

- incorporar outros fatores ao custo: a precificação de experimentos devem incluir não apenas os custos de geração de dados (seja microarranjos, sequenciamento etc.), mas também os gastos associados à obtenção das amostras, à análise e armazenamento de dados;

- compreender e determinar, *a priori*, os níveis α e β: é preciso que os níveis aceitáveis dos erros tipo I e II sejam determinados antes da realização do experimento. Estes parâmetros são essenciais para a determinação de tamanho amostral;

- determinar, também *a priori*, a magnitude mínima da diferença, δ, a ser detectada: a determinação de significância na comparação entre grupos é possível apenas por meio

da obtenção de diferenças relativas entre os grupos de interesse. O tamanho amostral é inversamente proporcional à magnitude mínima desta diferença;

- escolha adequada do tamanho amostral: é necessária a determinação precisa do tamanho amostral afim de evitar a perda do experimento (quando o tamanho amostral é tão pequeno, que é impossível detectar a variabilidade e diferenças) ou desperdício de recursos (com uma amostra muito grande, recursos são gastos desnecessariamente).

Estes parâmetros são empregados por algoritmos atuais para a determinação de tamanhos amostrais. Hart et al. (2013) usam como fórmula básica para determinação do número de amostras por grupo a expressão abaixo:

$$n = 2\left(z_{1-\frac{\alpha}{2}} + z_{1-\beta}\right)^2 \left(\frac{\frac{1}{\mu} + \sigma^2}{\log_e \delta^2}\right), \tag{1.4}$$

onde α e β são como já definidos neste trabalho, δ representa a diferença a ser detectada ($\delta = 1.5$ indica um ganho de 50% nos níveis de expressão), μ denota a cobertura e σ^2 representa o coeficiente de variação.

Alocação de amostras

Um outro desafio enfrentado por profissionais envolvidos na bioinformática no País é a alocação de amostras a placas para hibridização e *flowcells* para sequenciamento. O planejamento de experimentos é baseado em três princípios básicos, conforme Fisher (1935):

- replicação: refere-se à multiplicidade de indivíduos pertencentes ao mesmo grupo experimental. Cada um destes indivíduos é denominado réplica. Em experimentos como os

aqui descritos, é possível ter dois tipos de réplicas: biológicas e técnicas. As réplicas técnicas são empregadas na a avaliação de variabilidade técnica de sistemas e são formadas por diferentes amostragens do mesmo material genético (RNA ou DNA). Já as réplicas biológicas referem-se a extrações distintas de indivíduos diferentes e são utilizadas na avaliação de variabilidade biológica entre os grupos de interesse;

- aleatorização: considerando a posição na placa de hibridização ou mesmo a posição em uma *flowcell* como um fator a ser levado em consideração, é preciso aleatorizar a alocação das amostras (independente de seus *statuses* de grupo). Esta ação tem como consequência a redução do efeito de fatores externos na mensuração da variável de interesse;

- blocagem: a execução de experimentos possui, de modo intrínseco, uma série de variáveis que não são de interesse primário do pesquisador. Estas variáveis devem ser inseridas pelo analista na modelagem dos dados, caso contrário, elas inflacionarão a variância residual estimada. O controle destas variáveis de interesse secundário é chamado blocagem. Exemplos usuais de blocagem são a incorporação de dados de técnicos que trabalharam na preparação de amostras, sequenciadores ou escâner utilizados.

Um caso particular surge ao trabalhar-se com dispositivos de sequenciamento como equipamentos Illumina. Sequenciadores HiSeq possuem *flowcells* com oito canaletas. Muitos pesquisadores costumam alocar um grupo de amostras a *flowcells* diferentes, como demonstra a Figura 1.1. Esta estratégia é sub-ótima pois os grupos de indivíduos encontram-se confundidos com as *flowcells*. Isso impossibilita, portanto, a correta estimação dos efeitos de grupo.

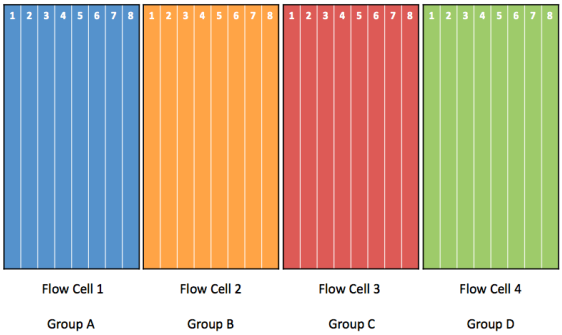


Figura 1.1 – Alocar indivíduos do mesmo grupo à mesma *flowcell* induz o confundimento, o que impossibilita a correta estimação dos efeitos de interesse.

Uma alternativa é garantir que amostras de todos os grupos estejam presentes em todas as *flowcells*, conforme representa a Figura 1.2. Esta estratégia permite que efeitos de *flowcell* sejam estimados, usando o conceito de blocagem apresentado anteriormente. Resultados mais eficientes são obtidos ao combinar a aleatorização das amostras e grupos às canaletas de cada *flowcell*.

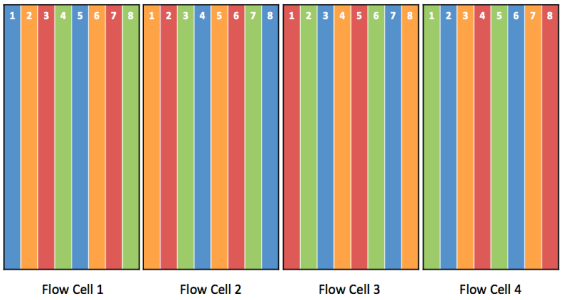


Figura 1.2 – Alocar indivíduos de grupos distintos à mesma *flowcell* permite que efeitos de *flowcell* sejam estimados. Esta alternativa permite melhor desempenho quando combinado com aleatorização de canaletas com respeito a indivíduos e grupos.

Idealmente, neste cenário de múltiplas canaletas por *flowcell*, a alternativa a ser empregada é a multiplexação de amostras. Neste cenário, cada canaleta possui réplicas técnicas de todas as réplicas biológicas. Esta estratégia permite não apenas a remoção de efeitos associados a *flowcells*, como também a estimação e remoção de efeitos de canaletas. Esta forma de delineamento também permite a execução de um controle interno de qualidade, no qual *flowcells* e canaletas podem ser averiguadas com respeito a desvios sistemáticos do padrão estabelecido pelos demais. Esta maneira de planejamento encontra-se representado na Figura 1.3.

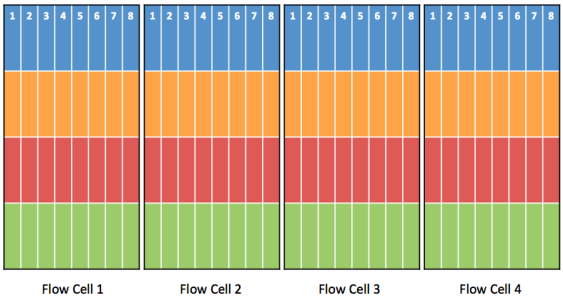


Figura 1.3 – O uso de multiplexação permite a estimação de efeitos específicos de *flowcells* e, também, de canaletas. Esta estratégia também funciona como uma espécie de cópia de segurança, visto que problemas que atinjam *flowcells* ou canaletas específicas não implicam na perda total dos dados das amostras associadas.

1.4 Modelagem de dados de sequenciamento

Em um cenário no qual a tecnologia muda tão frequentemente, um desafio refere-se à tática empregada para a modelagem de dados. Atualmente, os modelos mais frequentemente utilizados baseiam-se em modelos lineares generalizados, GLMs, conforme McCullagh e Nelder (1989).

O modelo mais simples baseia-se no modelo de Poisson, empregado, neste caso, por conta de sua relação direta com dados de contagem. Sob este ponto de vista, o número de observações na região i para o indivíduo j, N_{ij}, tem um valor esperado μ_{ij}, como apresentado pela Equação 1.5.

$$N_{ij} \sim \text{Poisson}\left(\mu_{ij}\right) \tag{1.5}$$

Assim, para modelar o valor médio do número de ocorrências na região acima referida, utliza-se simplesmente:

$$\log\left(\mu_{ij}\right) = X_i\beta, \tag{1.6}$$

onde X_i representa a matriz de delineamento para a i-ésima região e β, o vetor de efeitos a serem estimados.

Uma característica bem conhecida de modelos de Poisson é a igualdade entre a média e a variância. Enquanto isso é concordante com o padrão de variabilidade técnica para sequenciamento, o mesmo não ocorre na modelagem de dados provenientes de réplicas biológicas.

Neste último caso, variações entre indivíduos do mesmo grupo não são capturadas por este modelo e, então, faz-se necessária a atualização do modelo. A forma mais simples de acrescentar uma componente que reflita a variabilidade biológica é partir do modelo de Poisson e acrescentar uma distribuição *a priori* à média da distribuição, conforme mostrado abaixo:

$$\begin{aligned} N|\lambda &\sim \text{Poisson}\left(\lambda\right) && (1.7)\\ \lambda &\sim \text{Gamma}\left(\alpha, \beta\right) && (1.8)\\ N &\sim \text{NB}\left(\alpha, \frac{1}{1+\beta}\right), && (1.9) \end{aligned}$$

A Equação 1.7 representa o modelo de Poisson inicial. Num caso de modelagem da abundância de transcritos, este modelo

explica que o número de fragmentos observados para um transcrito possui uma média, a ser estimada, λ. Entretanto, é natural assumir que indivíduos diferentes possuem médias ligeiramente diferentes. Esta variação biológica é representada pela Equação 1.8, uma distribuição *a priori* do parâmetro λ. Combinando a distribuição condicional dada pela Equação 1.7 com a *priori* da Equação 1.8, obtem-se a distribuição *a posteriori* da Equação 1.9, que nada mais é que uma distribuição Binomial-Negativa, cuja variância possui uma componente técnica e outra biológica, conforme apresentado abaixo:

$$\text{Var}\,(N) = \frac{\alpha}{1+\beta}\left(\frac{1+\beta}{\beta}\right)^2 = \frac{\alpha}{\beta^2} + \frac{\alpha}{\beta} = \frac{\alpha}{\beta}\left(1+\frac{1}{\beta}\right) \quad (1.10)$$

Assim, com o modelo de Binomial-Negativa, é possível obter estimativas que melhor refletem o processo de geração de dados. Esta alternativa incorpora duas componentes para a variância, que podem ser vistas como uma componente técnica e outra biológica.

As mudanças constantes na tecnologia de sequenciamento sempre induzem mudanças nos modelos empregados. Portanto, é bastante importante a interação entre as partes responsáveis pela geração e modelagem dos dados, afim de garantir atualizações que permitam uma modelagem mais precisa do processo.

1.5 Procedimentos na análise

Além da necessidade de constante atualização de métodos de modelagem, como descrito anteriormente, é também preciso despender atenção em pontos adicionais que incluem: estratégias de manipulação de grandes volumes de dados e reprodutibilidade.

Manipulação de grandes volumes de dados

Dados provenientes de equipamentos de alto rendimento são armazenados em arquivos de dados bastante volumosos. Dependendo dos parâmetros do experimento, é possível ter arquivos que ocupam centenas de *gigabytes* por indivíduo amostrado. Ao compor o quadro de participantes do estudo com arquivos tão volumosos, torna-se inviável a manipulação comum destes. Neste cenário, todo o conteúdo do arquivo é carregado na memória de acesso aleatório do equipamento, impossibilitando a execução apropriada de algoritmos de análise.

O desafio neste setor é produzir algoritmos capazes de acessar, de forma eficaz, apenas porções dos dados de todos os indivíduos e proceder com a análise de modo incremental. Formatos dedicados a dados de sequenciamento, como arquivos BAM, auxiliam nesta tarefa, mas não são otimizados para acesso aleatório. Alternativamente, formatos de arquivo de alto desempenho, como HDF5 e NetCDF, permitem acesso imediato a porções de dados, mas ainda não oferecem uma boa representação dos dados em sua forma mais inicial, tais como sequências e componentes de qualidade.

Existe, ainda neste setor, uma demanda crescente por ambientes computacionais de alto desempenho. Vários grupos optam por montar seus próprios núcleos computacionais. Entretanto, iniciativas governamentais devem ser utilizadas em todo o potencial. O modelo nacional de sistemas computacionais de alto desempenho é implementado pelo SINAPAD, Sistema Nacional de Processamento de Alto Desempenho, por meio de investimentos do Ministério de Ciência e Tecnologia.

O SINAPAD possui centros regionais em diferentes localizações do Brasil. Estes centros são os CENAPADs, Centros Nacionais de Processamento de Alto Desempenho, localizados

na: UNICAMP, UFRGS, INPA, UFPE, CPTEC/INPE, UFMG, LNCC, COPPE/UFRJ e UFC. Com estes centros, pesquisadores têm acesso a centros computacionais de alto desempenho administrados por especialistas. Deste modo, faz-se um uso eficiente dos recursos de pessoal atualmente disponível. Deve-se observar que, no *ranking* de supercomputadores da América Latina (Lartop50 2013), o CENAPAD-SP (UNICAMP) aparece na segunda posição, enquanto o CENAPAD-RJ (LNCC) ocupa a sexta posição.

Os algoritmos de análise atuais devem, portanto, ser capazes de utilizar sistemas de alta performance, sejam eles *clusters* de computadores (como os disponibilizados pelo SINAPAD) ou soluções mais simples (como computadores de múltiplos núcleos atualmente comercializados). Estratégias de paralelismo devem ser utilizadas mais frequentemente afim de prover um uso mais eficiente do hardware disponível.

Reprodutibilidade

Uma das tarefas mais desafiadoras em bioinformática é garantir a reprodutibilidade de resultados obtidos em uma dada análise. Isso é simplificado com o uso de software que seja transparente e permita a incorporação de ferramentas externas, incluindo geradores automáticos de relatórios.

Gentleman et al. (2004) propõem um conjunto de ferramentas, baseadas na linguagem de análise e visualização de dados R, denominado BioConductor. Este projeto, livre e de código-aberto, estabelece padrões de representação de dados biológicos e possui um ciclo de atualização bastante acelerado. Duas vezes por ano, habitualmente em abril e outubro, novas versões do projeto são disponibilizadas, o que permite a distribuição rápida de novos métodos de análise e visualização.

Ferramentas disponibilizadas via BioConductor são completamente transparentes, no sentido de que o usuário pode rever o código para compreender os detalhes executados durante a análise. Elas são empacotadas em um sistema, o R, que permite o uso de outras ferramentas, como aquelas disponibilizadas em Python, Perl, C ou Fortran. O projeto foca também na portabilidade, o que permite que um código escrito em uma dada plataforma computacional seja utilizado, sem modificações, em outra.

Desta maneira, ao combinarem-se o ferramental do R e BioConductor com a estratégia descrita em Gentleman (2005), torna-se possível a total integração de texto e código. Neste cenário, o autor de um manuscrito produz um único documento que contém não apenas o texto, mas também o código, em R, utilizado para a análise de dados. O sistema de preparo de documentos dinâmicos, denominado Sweave, extrai e executa os comandos de análise, produzindo resultados, figuras e tabelas que são automaticamente incluídos no texto. O resultado, geralmente um arquivo no formato PDF, pode ser distribuído como produto final.

Este conceito de documentos dinâmicos permite a catalisação de produção científica e é uma ferramenta essencial para a reprodutibilidade de resultados reportados. Com um documento que integra texto e análises, as partes interessadas em reproduzir os resultados necessitam apenas ter um sistema que possua as características mínimas requeridas pelo documento e análise.

A visão modular habitualmente empregada em R e BioConductor pode também ser utilizada de forma alternativa ou até complementar com a ferramenta Snakemake, proposta por Köster e Rahmann (2012). Ela também estimula o foco em módulos de análise e reprodutibilidade. Sendo uma ferramenta desenvolvida em Python, possui grande capacidade de integração com sistemas já em produção. O Snakemake é capaz de automaticamente detectar

passos que não precisam ser executados novamente e também executar tarefas de modo paralelo.

1.6 Formação de pessoal

Como já foi apresentado, no cenário de desenvolvimento constante do setor de bioinformática no País, é preciso observar que a formação de pessoal capacitado é o maior desafio que existe no momento. A multidisciplinaridade desta área exige uma formação ampla dos profissionais do setor. Espera-se que, no mínimo, o bioinformata tenha conhecimentos sólidos em estatística, computação científica e, também, biologia. Além disso, é desejável, que esses conhecimento seja complementado com outros conteúdos, tal como, a matemática aplicada.

Ao longo dos anos, observou-se que um perfil tão amplo não permite a formação adequada do profissional em um espaço de tempo aceitável. Assim, ganhos mais efetivos são obtidos com a formação de uma equipe composta por profissionais de diversos campos – por exemplo, estatística, computação e biologia –, capazes de uma comunicação efetiva e bem integrados à rede de pesquisadores da área em questão.

No País, observam-se várias iniciativas com o propósito de formação de pessoal e desenvolvimento de conhecimento local. Agências financiadoras federais e estaduais tem apresentado um número significativo de chamadas que visam acelerar o desenvolvimento da bioinformática no Brasil. Entretanto, ao invés de recriar as estruturas de desenvolvimento de pessoal, as equipes nacionais devem utilizar os conhecimentos já disponíveis, obtido pela experiência de outras equipes nacionais e equipes internacionais, para acelerar o crescimento da área.

Tão importante quanto a formação de pessoal, é a reciclagem de profissionais. Com as frequentes atualizações de protocolos

e estratégias de análise, é necessário que os profissionais envolvidos na bioinformática no Brasil organizem-se como uma equipe única para atualização de conhecimentos. Cursos especializados, já rotineiros no exterior por meio de iniciativas, por exemplo, da Fundação BioConductor e Instituto Europeu de Bioinformática, precisam ser disponibilizados no País mais frequentemente.

Além da atenção à análise de dados, é essencial o estímulo à produção nacional de dados. A criação de centros de genômica e proteômica no âmbito brasileiro, como o LaCTAD na UNICAMP, é de grande importância. Entretanto, o desenvolvimento nacional deste setor só será de fato observado à medida que pesquisadores utilizem cada vez mais estes centros nacionais para a geração de dados. Com esta estratégia, o País desenvolverá auto-suficiência em estudos avançados utilizando tecnologias de alto desempenho.

1.7 Considerações finais

A bioinformática no País enfrenta uma série de desafios e, neste capítulo, foram discutidos alguns dos mais relevantes para esta área cuja característica maior é a multidisciplinaridade, que envolve a compreensão de conceitos estatísticos, estratégias computacionais para análise de dados e formação de pessoal.

Este setor lida com uma vasta coleção de dados, não apenas dados gerados pelo experimento em si, mas também coleções utilizadas para a anotação funcional dos alvos estudados. Independente da origem, ferramentas estatísticas são as que permitem a inferência de características de interesse do pesquisador. A interação com o estatístico inicia-se antes mesmo da coleta de amostras, durante a fase do planejamento do próprio experimento. Já nesta fase, conceitos estatísticos essenciais são utilizados constantemente. Neste trabalho, foram discutidos alguns destes conceitos,

afim de apresentar ao leitor a forma com que todos estes conceitos se conectam para a análise final dos dados.

Com o volume de dados envolvidos nestas análises, grupos de pesquisa vêem-se na situação de montar seus próprios ambientes computacionais. O Brasil, por meio do Ministério de Ciência e Tecnologia, provê a seus pesquisadores sistemas de computação de alto desempenho por meio do SINAPAD. Núcleos regionais, CENAPADs, fazem a interface com o pesquisador, provendo acesso a estes sistemas, que aparecem muito bem classificados no *ranking* de supercomputadores da América Latina. Com esta combinação de grandes volumes de dados e ambientes computacionais de alto desempenho, surgem novas oportunidades de desenvolvimento de software que utilizem de modo eficaz toda esta estrutura.

Para fechar a tríade de desafios, deve-se adotar no País estratégias de formação de pessoal que valorizem a multidisciplinaridade do setor. Estes métodos já vem sido empregados com sucesso nos Estados Unidos, por meio da Fundação BioConductor, e Europa, com iniciativas do Instituto Europeu de Bioinformática. Com a atração de novos profissionais para o mercado nacional, treinamentos como os oferecidos por ambas as instituições começam a ser oferecidos mais frequentemente. Entretanto, além de oferecer treinamentos que visem a formação de novos profissionais, deve-se também estimular a criação e execução de treinamentos que foquem na reciclagem dos profissionais já no mercado. As iniciativas promovidas por agências financiadoras são de extrema relevância, mas também devem valorizar estratégias de longo prazo, que visem o estabelecimento da auto-suficiência na geração e análise de dados usando tecnologias de alto rendimento no País.

1.8 Referências

FISHER, R. *The design of experiments. 1935*. Edinburgh: Oliver

and Boyd, 1935.

GENTLEMAN, R. Reproducible research: a bioinformatics case study. *Statistical applications in genetics and molecular biology*, Harvard University, USA. rgentlem@fhcrc.org, v. 4, p. Article2, Jan 2005. ISSN 1544-6115. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/16646837>.

GENTLEMAN, R. C. et al. Bioconductor: open software development for computational biology and bioinformatics. *Genome biology*, Department of Biostatistical Science, Dana-Farber Cancer Institute, 44 Binney St, Boston, MA 02115, USA. rgentlem@jimmy.harvard.edu, v. 5, n. 10, p. R80, Sep 2004. ISSN 1465-6914. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/15461798>.

HART, S. N. et al. Calculating sample size estimates for rna sequencing data. *Journal of computational biology : a journal of computational molecular cell biology*, 1 Division of Biomedical Statistics and Informatics, Department of Health Sciences Research, Mayo Clinic , Rochester, Minnesota., Aug 2013. ISSN 1557-8666. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/23961961>.

KÖSTER, J.; RAHMANN, S. Snakemake–a scalable bioinformatics workflow engine. *Bioinformatics (Oxford, England)*, Genome Informatics, Institute of Human Genetics, University of Duisburg-Essen and Paediatric Oncology, University Childrens Hospital, 45147 Essen, Germany. johannes.koester@uni-due.de, v. 28, n. 19, p. 2520–2522, Oct 2012. ISSN 1367-4811. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/22908215>.

MCCULLAGH, P.; NELDER, J. A. *Generalized linear models (Second edition)*. London: Chapman & Hall, 1989. 500 p.

2

Mineração de dados para identificar atributos genéticos associados à características de interesse econômico à pecuária

André Gonzaga
Maurício de A. Mudadu
Roberto H. Higa
Eduardo Hruschka

PESQUISADORES DA área de melhoramento genético possuem cada vez mais acesso a dados genéticos e genômicos e demandam por um método ou ferramenta robusta que atendam às suas necessidades na descoberta de conhecimento. Esse trabalho investiga algoritmos e métodos de mineração de dados aplicados ao contexto de melhoramento genético bovino, concentrando-se fundamentalmente na aplicação e no aperfeiçoamento de algoritmos para seleção de atributos, buscando identificar os atributos genéticos associados às características fenotípicas de um indivíduo em bases de dados simuladas. Para tanto, foram utilizados algoritmos existentes, baseados em agrupamento de dados e em teoria de informação, bem como foi desenvolvido um novo método para seleção de atributos e que é baseado no conceito de janela

deslizante. Os resultados gerados são compatíveis com o domínio do problema e equivalentes aos obtidos por algoritmos tradicionais da área de bioinformática, os quais são usualmente mais caros computacionalmente.

2.1 Introdução

O Brasil possui o maior rebanho bovino comercial do mundo, com mais de 200 milhões de cabeças. No ano de 2013 abateu por volta de 43 milhões de cabeças e exportou 2 milhões de toneladas de equivalente carcaça (ABIEC, 2014). A identificação dos atributos genéticos que determinam características como maciez da carne e porcentagem de gordura no leite podem ter valor econômico direto nesse mercado.

Uma das maneiras tradicionais de se realizar melhoramento de bovinos é por seleção de animais, direcionando os cruzamentos com o uso de informações de desempenho, *pedigree* e cálculo do valor ou mérito genético de um animal (FALCONER; MACKAY, 1996). Alternativamente é possível o uso de atributos genômicos (marcadores SNP, do inglês *single nucleotide polymorphism*, ou polimorfismo de um único nucleotídeo) para assistir na seleção de animais e também para realizar seleção e predição genômica (MEUWISSEN; HAYES; GODDARD, 2001).

O genoma bovino é composto por 31 cromossomos com milhares de genes, que são responsáveis pela expressão de características individuais (LIU et al., 2009), as quais podem variar desde a tendência em desenvolver uma determinada doença, até a capacidade de crescimento do animal. Em termos práticos, torna-se interessante identificar quais regiões do genoma bovino estão relacionadas às características de interesse econômico (essas regiões também são denominadas QTL – Quantitative Trait Loci) para a produção de bovinos de corte e de leite, a fim de tornar possível estimar

o mérito genético-genômico de um determinado indivíduo (HAYES et al., 2009).

Avanços na tecnologia de sequenciamento de DNA e genotipagem levaram ao desenvolvimento de *chip* de DNA de alta densidade que podem ser usados para caracterizar centenas de milhares de marcadores SNP em um único ensaio. A busca e identificação da importância de um dado SNP para uma dada característica de desempenho de um animal (ou os fenótipos, usando terminologia biológica) é feita por meio de testes de associação. Essa busca por marcadores relevantes pode ser ser realizada em todo o genoma via estudos amplos de associação (GWAS – Genome-wide Association Studies) (BALDING, 2006).

Existem diversas metodologias para realizar GWAS. As mais comuns são métodos paramétricos de regressão linear e logística, nos quais são usadas funções para regredir os fenótipos (características de desempenho dos animais) aos atributos genéticos (genótipos ou conjunto de marcadores SNP, usados como covariáveis). Essas funções podem ser interpretadas como uma aproximação para os valores genéticos verdadeiros e desconhecidos e são geralmente modelos matemáticos que envolvem os genótipos, as interações entre os genótipos e condições ambientais (CAMPOS et al., 2013). É importante lembrar que métodos paramétricos usualmente assumem normalidade, linearidade e ausência de colinearidade, o que nem sempre é a regra nesse contexto.

Metodologias não paramétricas são uma alternativa aos métodos tradicionais, principalmente na tentativa de se modelar interações não-lineares que não são capturadas pelos métodos paramétricos. Interações não-lineares são consequência de uma relação mais complexa entre o fenótipo e os genótipos, como, por exemplo, heterogeneidade de *locus* (diferentes trechos do genoma levando ao mesmo fenótipo), fenocópia (fenótipos determinados

exclusivamente pelo ambiente e sem base genética) e epistasia (interações entre genes) (MOORE; ASSELBERGS; WILLIAMS, 2010). Entre as metodologias não-paramétricas utilizadas nesse contexto estão aquelas baseadas em técnicas de mineração de dados e o de aprendizado de máquina, que são métodos computacionais, os quais são essencialmente métodos de modelagem computacional – por exemplo, árvores de classificação e decisão, redes neurais, *support vector machines* (HOWARD; CARRIQUIRY; BEAVIS, 2014).

O presente estudo reporta o uso de GWAS em bases de dados simuladas de fenótipos e genótipos voltadas para o contexto da pecuária, de forma a testar e comparar algumas metodologias não-paramétricas com metodologias paramétricas tradicionais.

2.1.1 Seleção genômica e GWAS

O uso da genômica no melhoramento animal vem sendo adotado por criadores e produtores nos últimos anos (ROLF et al., 2014). A maioria das características de importância econômica na pecuária são quantitativas ou poligênicas, ou seja, consequência da ação de muitos genes que contribuem cada um com uma pequena parcela para a variância fenotípica. Em 2001, Meuwissen, Hayes e Goddard (2001) propôs o uso de marcadores SNP no cálculo mais acurado do mérito genético de animais, em um método denominado seleção genômica. Essa metodologia é indicada principalmente para características em que a seleção genética tradicional é pouco eficiente, como no caso de características complexas e de baixa herdabilidade, assim como características obtidas tardiamente como dados de carcaça e eficiência alimentar.

O termo GWAS foi cunhado devido à distribuição quase homogênea desses marcadores pelo genoma de forma a cobrir grande parte de sua extensão. Dessa forma esses marcadores podem servir como ferramenta para capturar de forma eficiente

o efeito poligênico de características quantitativas, em testes de associação. GWAS requer três elementos: (i) muitas amostras, ou animais, se possível de diversas populações; (ii) marcadores genéticos que cubram grande parte do genoma e (iii) métodos analíticos poderosos o suficiente para identificar sem viés a associação entre marcadores e os fenótipos (CANTOR; LANGE; SINSHEIMER, 2010).

2.1.2 Atributos genéticos: marcadores SNP

Os atributos genéticos submetidos à mineração de dados nesse projeto são os marcadores genéticos do tipo SNP. Esses marcadores são um tipo de polimorfismo em que há alteração em um único nucleotídeo do DNA: A, T, C ou G, cujas letras são referentes às bases nitrogenadas adenina, guanina, citosina ou timina.

Em outras palavras, SNP é uma variação pontual na sequência de DNA e que mantém frequência mensurável em uma população. Os SNPs são os mais abundantes de todos os marcadores genéticos existentes, pois são distribuídos uniformemente por todo o genoma do indivíduo. Estima-se que existam mais de 10 milhões de marcadores SNP no genoma humano e que pelo menos 300 mil deles implicam variações genéticas significantes (SHAH; KUSIAK, 2004).

Graças à interação gênica, SNPs podem servir de marcadores para determinar regiões do genoma que estejam associadas à características fenotípicas de interesse econômico como: maciez da carne, tamanho da área de olho de lombo, produção de leite etc.. Essa interação gênica, também denominada desequilíbrio de ligação (DL), indica que marcadores SNP podem não ter efeito direto em um fenótipo, mas podem estar associados a um trecho do

genoma responsável por uma característica, servindo de “marca” para essa região.

Outra conclusão que pode ser tirada do DL é que o conjunto de marcadores SNP é redundante, ou seja, marcadores SNP podem estar associados entre si, de forma que também poderão estar associados simultaneamente a um dado fenótipo. Uma metodologia para se eliminar essa redundância é a construção de blocos de SNP que estão em DL. Tais blocos podem então ser usados nos testes de associação, ou então para selecionar marcadores SNP referência dentre vários de uma região, denominados *tag* SNP (BALDING, 2006).

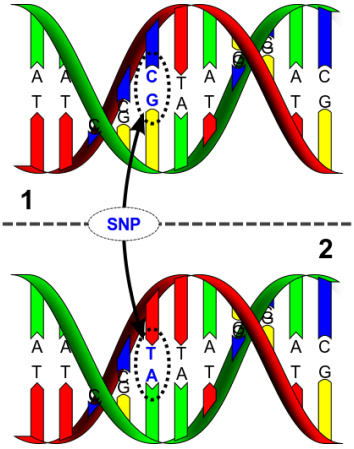


Figura 2.1 – Representação de um polimorfismo de nucleotídeo simples. Fonte: Wikipédia Commons (HALL, 2007).

2.1.3 QTL-MAS: simulação de dados genéticos

A acurácia e poder de métodos de seleção genômica e GWAS podem ser avaliados utilizando dados genômicos simulados (DAETWYLER et al., 2013). As bases de dados utilizadas nesse trabalho são oriundas de dados simulados, disponibilizados

no *workshop* QTL-MAS dos anos 2012 (PANICO et al., 2012) e 2011 (ROY et al., 2011), o qual reúne diversos especialistas em seleção genética de plantas e animais e também fornece dados biológicos artificiais com o conjunto dos parâmetros genéticos utilizados na simulação. Essa base é composta por um conjunto de vários animais (instâncias) com milhares de SNPs associados à uma característica quantitativa ou qualitativa (classe).

A Tabela 2.1 exemplifica as bases de dados utilizadas nesse trabalho. Note que os atributos genéticos são categóricos e podem assumir os valores 11, 12, 21 ou 22, representando as possibilidades de polimorfismo no alelo em questão, considerando que os marcadores SNP são bialélicos. Já o fenótipo é um atributo quantitativo, podendo assumir qualquer valor real ($\mathbb{R}$), representando a quantificação de alguma característica de interesse no indivíduo.

A ordem dos marcadores SNP é de suma importância, pois eles são distribuídos uniformemente pelo genoma simulado, o que é uma aproximação da realidade. Dessa forma, nos dados simulados, a distância entre dois SNP consecutivos permanece constante em todo o cromossomo, ou seja, o SNP 1 é seguido pelo SNP 2 que por sua vez é seguido pelo SNP 3, e assim por diante.

Tabela 2.1 – Representação genérica das bases de dados com N SNPs e M indivíduos.

Indiv.	SNP 1	SNP 2	...	SNP N	Fenótipo
1	{11,12,21,22}	{11,12,21,22}		{11,12,21,22}	$\mathbb{R}$
2	{11,12,21,22}	{11,12,21,22}		{11,12,21,22}	$\mathbb{R}$
...	{11,12,21,22}	{11,12,21,22}		{11,12,21,22}	$\mathbb{R}$
M	{11,12,21,22}	{11,12,21,22}		{11,12,21,22}	$\mathbb{R}$

2.2 Metodologia

Esta seção apresenta, resumidamente, uma visão geral da teoria sobre os métodos tradicionais (paramétricos) e os méto-

dos computacionais (não-paramétricos) que foram utilizados para realizar GWAS com os dados genômicos e fenotípicos simulados do QTL-MAS 2011 e 2012.

2.2.1 Métodos paramétricos

Para se calcular a associação de marcadores genéticos a uma dada característica em um GWAS, pode-se usar um modelo linear paramétrico tradicional em uma regressão:

$$y = \mathbf{X}b + \sum_{i=1}^{m} z_i g_i + e,$$

onde y é um vetor com valores fenotípicos dos animais, b é um vetor de efeitos fixos e X a matriz de incidência relacionando o vetor de efeitos b em y, g_i é o efeito de cada marcador variando de i a m, z_i é o vetor de genótipos de cada indivíduo por marcador i, e o último termo, e, é um vetor de erros residuais aleatórios.

Baseado nesse modelo, diversas outras metodologias foram derivadas como as metodologias de regressão *ridge* BLUP (*best linear unbiased prediction*), rrBLUP (*ridge regression* BLUP) e gBLUP (*genomic* BLUP) (ZHANG; ZHANG; DING, 2011) assim como metodologias bayesianas como o BAYES LASSO, Bayes A, Bayes B, Bayes C e Bayes Cπ (HOWARD; CARRIQUIRY; BEAVIS, 2014).

Tais modelos de regressão testam um SNP por vez, podendo gerar falsos positivos devido ao viés de testes múltiplos, porque ignoram a interação entre os marcadores SNP (MOORE; ASSELBERGS; WILLIAMS, 2010). O uso de blocos de SNP ou de *tag* SNP, ao invés de SNP únicos nos testes de associação pode ser uma alternativa para reduzir esse problema. Para montar os blocos de SNP e calcular os *tag* SNP é necessário calcular o nível de DL entre todos os SNP, par a par. Um dos softwares mais usados para realizar essa tarefa é o Haploview (BARRETT et al., 2005). Antes

de se montar os blocos é necessário reconstruir a “fase de ligação”, que nada mais é do que estimar de qual cromossomo, materno ou paterno, um dado SNP se originou. O Beagle é um software muito utilizado nesse intuito (BROWNING; BROWNING, 2007).

Algumas metodologias paramétricas estimam o efeito de cada SNP no fenótipo (% da variância fenotípica explicada) assim como há metodologias que geram *p-valor*, referentes ao teste de uma hipótese nula de não-associação. Métodos não-paramétricos não fazem uso de tais estatísticas e muitas vezes o parâmetro de medida de associação de um SNP a um fenótipo são scores que medem a importância de um dado SNP para explicar o(s) modelo(s) de classificação. Essa “importância” é geralmente medida como a frequência com que um dado SNP aparece nas repostas dos modelos computacionais usados na classificação.

2.2.1.1 PLINK

O PLINK é uma das ferramentas mais conhecidas e eficientes para se manipular dados de SNP (PURCELL et al., 2007; CLARKE et al., 2011) além de também realizar GWAS. Dentre as várias metodologias disponíveis para realizar estudos de associação com características quantitativas nesse software, encontram-se as regressões lineares. Nesse caso PLINK retorna valores dos coeficientes de regressão (β) e *p-valor* para estatísticas T individuais para cada SNP. Também é possível realizar correções para viés de múltiplos testes, como ajustes de Bonferroni, Sidak, False Discovery Rate (FDR), etc., além de procedimentos de permutação para gerar níveis de significância empiricamente. O PLINK também permite utilizar blocos de SNP para realizar testes de associação.

2.2.1.2 Gensel

O Gensel é um software usado para seleção genômica no qual foram implementados diversos métodos bayesianos, como Bayes A, B, C e Cπ. O software não é aberto ao público mas pode ser acessado por colaboradores da Iowa State University (FERNANDO; GARRICK, 2009). Ao contrário do PLINK que gera *p-valor* para cada SNP, indicando se há associação com o fenótipo, um dos arquivos resposta gerados pelo Gensel indica a porcentagem de variância fenotípica explicada por blocos de SNP, no caso indicando regiões que são possíveis QTL.

2.2.2 Mineração de dados e aprendizado de máquina

A mineração de dados envolve basicamente três etapas: preparação de dados, utilização de um algoritmo para reconhecimento de padrões e análise das informações obtidas (BIGUS, 1996). Estas três etapas são correlacionadas e interdependentes, de tal forma que a abordagem ideal para extrair informações relevantes em bancos de dados consiste em considerar as inter-relações entre cada uma das etapas e sua influência no resultado final. Em linhas gerais, a preparação de dados envolve a seleção de atributos e de objetos (registros) adequados para a mineração, a representação e o pré-processamento dos dados para as ferramentas de mineração, e a purificação dos dados (eliminação de ruído).

Usualmente, o principal objetivo no processo de seleção de atributos é melhorar o desempenho dos algoritmos de modelagem (WITTEN; FRANK, 2005), reduzindo o tamanho da base de dados por meio da remoção de atributos irrelevantes e redundantes. Resumidamente, existem três abordagens fundamentais para selecionar atributos: *filter*, *wrapper* e *embedded* (GUYON; ELISSEEFF, 2003). Filtros baseiam-se apenas nas propriedades intrínsecas dos dados e são computacionalmente mais baratos. Métodos *wrapper*

dependem de um algoritmo de aprendizado para buscar subconjuntos de atributos que sejam representativos da base de dados como um todo, sendo computacionalmente custosos. Já nos métodos do tipo *embedded* a seleção de atributos ocorre como parte integrante do algoritmo indutor do classificador. Ademais, existem também métodos híbridos, que combinam características dos três métodos mencionados.

2.2.2.1 Seleção de atributos via agrupamento de dados

O processo de agrupamento de dados envolve a identificação de um conjunto de categorias – também chamadas de grupos ou de clusters – que descrevam um conjunto de dados (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996), objetivando-se maximizar a homogeneidade entre os objetos de um mesmo grupo e, concomitantemente, maximizar a heterogeneidade entre objetos de grupos distintos.

Existem várias técnicas utilizadas para agrupamento de dados (*clustering*). No contexto desse projeto um dos algoritmos investigados é o filtro baseado em silhueta (SSF) (COVOES; HRUSCHKA, 2011), que se baseia na identificação de grupos de atributos correlacionados. Este algoritmo é bem fundamentado teoricamente, e pode ser visto como um aperfeiçoamento de dois algoritmos bem conhecidos – ACA (AU et al., 2005) e MMP (MITRA; MURTHY; PAL, 2002). Basicamente, o algoritmo agrupa atributos que sejam correlacionados entre si e depois seleciona apenas o atributo que seja o mais representativo de cada grupo.

Mais especificamente, o SSF supre algumas limitações encontradas no ACA e no MMP, bem como permite incorporar a informação da classe ao processo de agrupamento de atributos. Esta característica é de fundamental importância nessa pesquisa, pois

permite identificar grupos de atributos que sejam simultaneamente correlacionados entre si e com o atributo meta (classe).

2.2.2.2 Algoritmo baseado em ganho de informação: InfoGain

A partir da teoria da informação (COVER; THOMAS, 2006), o algoritmo denominado de InfoGain (WITTEN; FRANK, 2005) utiliza as medidas de entropia e informação mútua para selecionar os atributos mais relevantes em relação à classe principal de uma base de dados.

A entropia mede o grau de incerteza em relação aos valores que uma variável aleatória discreta pode assumir. A Expressão 2.1 representa o cálculo da entropia H para uma variável X, sendo x os possíveis valores da variável e p_x a probabilidade da variável X assumir um valor x na base de dados analisada. A entropia pertence ao intervalo $[0, 1]$ no qual 0 (zero) significa ter todas as instâncias com o mesmo valor para essa variável, ou seja, não há incerteza em relação ao seu valor, e 1 significa ter esses valores distribuídos uniformemente entre as instâncias da base de dados, consistindo o grau máximo de incerteza.

$$H(X) = -\sum_{x \epsilon X} p_x \, log_2 \, p_x. \tag{2.1}$$

A informação mútua representada pela Expressão 2.2 utiliza a medida de entropia condicional para avaliar o quanto uma variável contribui para determinar o valor de uma outra variável.

$$I(X;Y) = H(X) - H(X|Y) = \sum_{x,y} p(x,y) \, log \frac{p(x,y)}{p(x) \, p(y)} \tag{2.2}$$

O algoritmo InfoGain utiliza diretamente a medida de informação mútua para selecionar os atributos que estejam correlacionados com a informação da classe. Dessa forma, a medida de informação mútua é calculada pela Expressão 2.3 e a seleção dos

atributos é realizada por meio do ranqueamento dessa medida para cada um dos atributos avaliados.

$$\mathsf{InfoGain(Classe, Atributo)} = \mathsf{H(Classe)} - \mathsf{H(Classe|Atributo)} \quad (2.3)$$

2.2.2.3 Algoritmo baseado em janela deslizante

O algoritmo baseado no conceito de janela deslizante foi desenvolvido particularmente para o contexto dessa pesquisa com o objetivo de avaliar o mérito de atributos sequenciais, como é o caso dos SNP.

O método consiste basicamente em criar um subconjunto de atributos consecutivos, representando uma região de interesse no cromossomo. Esse subconjunto funciona como uma janela deslizante. A cada iteração do algoritmo o subconjunto é utilizado para classificar os dados e, em seguida, o algoritmo percorre o vetor de atributos adicionando, e removendo atributos sequencialmente, como mostra a Figura 2.2. Seleciona-se, então, o atributo central dos subconjuntos que apresentaram a melhor acurácia de classificação.

$$\underbrace{snp1 \;\; snp2 \;\; snp3 \;\; snp4 \;\; snp5} \;\; \text{snp6} \;\; \text{snp7} \;\; ...$$

$$\text{snp1} \;\; \underbrace{snp2 \;\; snp3 \;\; snp4 \;\; snp5 \;\; snp6} \;\; \text{snp7} \;\; ...$$

$$\text{snp1} \;\; \text{snp2} \;\; \underbrace{snp3 \;\; snp4 \;\; snp5 \;\; snp6 \;\; snp7} \;\; ...$$

Figura 2.2 – Funcionamento do algoritmo de janela deslizante.

2.3 Descrição do conjunto de dados

2.3.1 Análise da base de dados

Após a aquisição das bases de dados, realizou-se uma análise estatística da distribuição dos registros coletados. Na Tabela 2.2 observa-se que a quantidade de padrões genéticos[1] em relação ao número total de atributos é desproporcional, visto que dos 10 mil atributos totais menos de 1% são considerados padrões genéticos que influenciam o fenótipo.

Tabela 2.2 – Características das bases de dados utilizadas.

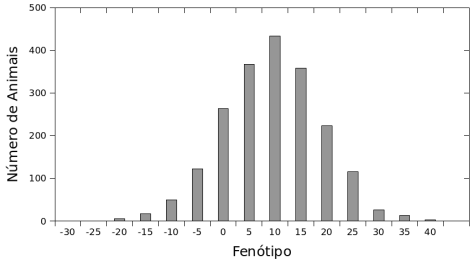
QTL-MAS	Fenótipo	Indivíduos	SNP	Padrões Genéticos
2011	[-30;40]	3 100	9 990	8
2012	[-600;600]	4 100	10 000	50

A principal diferença entre as duas bases de dados é quantidade de padrões genéticos simulados. No conjunto de dados de 2011, apenas oito dos dez mil marcadores genéticos são responsáveis pelo fenótipo dos animais, enquanto na simulação de 2012, cinquenta atributos estão relacionados à manifestação das características. Vale ressaltar que o fenótipo observado nos animais em cada uma das bases de dados é um atributo numérico que compreende diferentes intervalos, como pode ser visto na Tabela 2.2.

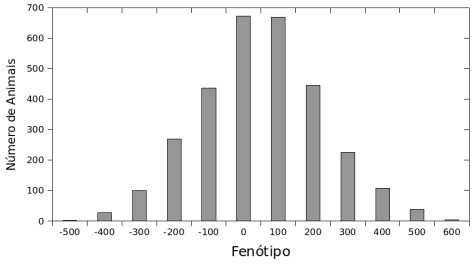
Sabe-se previamente que existem atributos genéticos que contribuem positivamente para uma característica e outros que contribuem negativamente, ou seja, dentre os padrões genéticos gerados na simulação, alguns marcadores são responsáveis por aumentar o valor fenotípico e outros por diminuir. Dessa forma, podem existir animais que possuem apenas os componentes que contribuem positivamente para uma característica, animais que possuem apenas

[1] Padrões genéticos são os atributos que estão correlacionados ao fenótipo (classe).

os componentes negativos, ou mais comumente, os animais que possuem ambos atributos genéticos e acabam por desenvolver um fenótipo médio, como pode ser visto nos histogramas da Figura 2.3.



(a) QTL-MAS 2011



(b) QTL-MAS 2012

Figura 2.3 – Histograma das bases de dados.

Dentre os parâmetros de simulação utilizados na criação dos padrões genéticos, existem alguns atributos que se destacam no quesito “capacidade de influência no fenótipo” e também na frequência de ocorrência na população (Tabela 2.3a). Um atributo genético que esteja presente em 100% da população torna-se impossível de ser identificado, mesmo que tenha um alto poder de influência no fenótipo, pois todos os animais teriam essa variante, tanto os indivíduos geneticamente favorecidos, quanto os desfavorecidos, tornando impossível classificá-los. O mesmo vale para

atributos que possuem uma influência mínima na característica (Tabela 2.3b).

Tabela 2.3 – Exemplos de SNP presentes nas simulações.

(a) Exemplo de SNP importantes

SNP	Efeito	Frequência
6499-6500	+74	47%
2673-2674	-42	65%

(b) Exemplo de SNP irrelevantes

SNP	Efeito	Frequência
3499-3500	+5	100%
2673-2674	-0.73	66%

2.3.2 Descrição do problema

O problema tratado consiste basicamente na seleção de atributos que sejam correlacionados a uma classe principal. Os atributos são componentes genéticos representados pelos SNP e a classe principal é a quantificação de um fenótipo observado no indivíduo, representado por um atributo numérico, como pode ser observado na Tabela 2.1.

A principal complexidade do problema é dada pelo alto número de atributos e o pequeno número de instâncias, caracterizando um caso da denominada “maldição da dimensionalidade” (BELLMAN, 1961), problema no qual os dados se tornam esparsos e pouco representativos. Outra particularidade da base de dados que também contribui para complexidade do problema se refere à grande quantidade de atributos irrelevantes visto que, no contexto biológico, dentre todos os componentes genéticos presentes no genoma de um indivíduo apenas alguns deles estão efetivamente associados ao seu fenótipo.

Devido à natureza do problema existe também a questão de inconsistência dos dados, pois o desenvolvimento de uma característica não é restrita apenas ao genoma do indivíduo, mas também às condições ambientais (GRIFFITHS et al., 2007). Assim sendo, não é raro obter indivíduos que possuem características contraditórias ao seu genótipo, tornando a base de dados suscetível

a ruídos causados por esses casos, como pode ser observado na Figura 2.4, na qual as barras azuis representam o fenótipo apresentado pelo indivíduo e as barras vermelhas representam o fenótipo esperado de acordo com o seu genoma. Pode-se inferir essa informação através dos parâmetros de simulação utilizados no conjunto dos dados do QTL-MAS 2012.

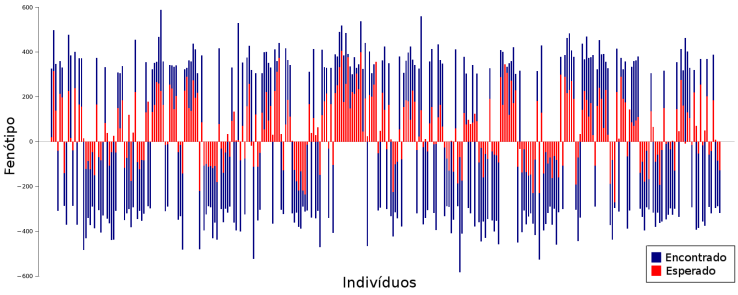


Figura 2.4 – Exemplo de ruídos na base de dados QTL-MAS 2012.

2.3.3 Metodologia dos métodos paramétricos

2.3.3.1 PLINK

PLINK foi utilizado inicialmente para realizar filtros de controle de qualidade nos dados: MAF (frequência do alelo menos frequente, ou do inglês *minimum allele frequency*) 1%, *call rate* para SNP 5% e amostras 10% . Para realizar GWAS em ambas as bases de dados (2011 e 2012), utilizou-se PLINK usando regressão linear. Foram adotadas duas estratégias: (i) testando um SNP de cada vez (*single SNP*) e (ii) testando blocos de SNPs. Em (i) foi usado o comando "*–linear*" e foi realizado um procedimento de permutação adaptativa para se obter *p-valor* com significância empírica, com o comando "*–perm*". Em (ii) foi usado o comando "*–hap-linear*" e 1.000 permutações com o comando "*–mperm 1000*".

Em ambas as estratégias, os resultados foram ordenados pelos *p-valor* obtidos com permutação e os 50 SNP e blocos de SNP com menores *p-valor* foram usados nas análises. Na estratégia (ii) o SNP do centro do bloco foi utilizado nos cálculos de Precision e Recall.

Para se obter os blocos de SNP foi usado o software Beagle para reconstruir a fase de ligação e o software Haploview para construir os blocos com o comando *-blockoutput GAB*.

2.3.3.2 GenSel

O GenSel foi utilizado com o método Bayes A e os parâmetros: chainLength = 41.000; burnin = 1.000; probFixed = 0,95; varGenotypic = 1; varResidual = 1; nuRes = 10; degreesFreedomEffectVar = 4; outputFreq = 100 e windowBV = *yes*, sendo que o último comando habilita a busca por blocos de SNP associados (QTL). Apenas os resultados por blocos foram utilizados nas análises, após ordená-los pela % de variância do fenótipo explicada por cada um, e os 50 blocos mais influentes foram utilizados. A mesma estratégia utilizada com o PLINK foi adotada para cálculo de Precision e Recall.

2.3.4 Metodologia dos algoritmos computacionais

A preparação dos dados limitou-se a obter rótulos de classe, a partir de uma variável associada ao fenótipo e que originalmente é representada por valores contínuos, em um atributo binário. Essa transformação aconteceu por meio de uma seleção dos animais (instâncias) que possuíam valores extremos em relação ao fenótipo em questão, ou seja, separamos os animais em dois grupos. Um grupo é formado por todos os animais cujo fenótipo se destaca pelo lado positivo e o outro grupo é formado por todos os animais que se destacam pelo lado negativo. Desta forma, foram

selecionados apenas os animais cujo mérito genético é muito alto ou muito baixo, eliminando o risco de se ter animais que possuem tanto os atributos que contribuem positivamente quantos os que contribuem negativamente.

A quantidade de animais que constituem os grupos é relativa pois seria necessário estabelecer um limiar para o qual o valor do fenótipo seria considerado extremo positivo ou extremo negativo. Uma alternativa à da escolha de um limiar seria dividir em grupos de mesmo tamanho. Por exemplo, em um caso de teste de 50 animais, metade seriam os indivíduos que desenvolveram os maiores valores para o fenótipo e a outra metade seriam os animais que possuem o fenótipo com os menores valores. No entanto o tamanho do grupo ainda é um valor arbitrário, assim sendo optou-se por criar seis diferentes partições com tamanhos distintos, sendo elas: 10, 20, 40, 80, 160 e 320.

A Figura 2.5 exemplifica o método utilizado para a seleção dos indivíduos candidatos à mineração. A criação de um grupo com 10 animais seria equivalente à selecionar apenas os indivíduos localizados nos extremos do histograma. Conforme o tamanho do grupo aumenta, os animais pertencentes ao grupo tendem a estar localizados mais próximos do centro.

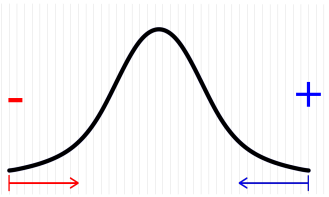


Figura 2.5 – Método para seleção dos indivíduos candidatos à mineração referenciando o histograma de fenótipos da Figura 2.3.

Seleção de atributos

Os experimentos com seleção de atributos foram realizados utilizando três algoritmos distintos que se encaixam no contexto do problema e são computacionalmente eficientes:

1. **Information Gain** (COVER; THOMAS, 2006): Calcula o mérito de um atributo em relação à uma classe principal utilizando a entropia condicional. Os atributos selecionados foram os 50 com maiores méritos.

2. **SSF** (COVOES; HRUSCHKA, 2011): Realiza a redução de dimensionalidade da base de dados agrupando atributos que são semelhantes. Os atributos selecionados são aqueles que melhor representam o grupo ao qual pertencem. A semelhança entre os atributos é determinada por uma medida de distância que pode ou não considerar a classe principal. O número de grupos pode ser fixado a *priori* ou estimado automaticamente a partir dos dados. Neste trabalho, optou-se por fixar o número de grupos em 50 e utilizar uma medida de distância que considera a classe principal.

3. **Janela deslizante:** Para executar os experimentos de seleção de atributos utilizamos uma janela de tamanho igual a 5, que equivale a avaliar o mérito de uma região no cromossomo com cinco SNPs. O classificador utilizado foi o KNN (K-Nearest Neighbour) (WANG; ZUCKER; JEAN-DANIEL, 2000), que consiste em classificar uma instância de acordo com os seus *K* vizinhos mais próximos, ou seja, aqueles indivíduos que mais se parecem com o objeto em questão, de acordo com seus atributos.

2.3.5 Medidas de qualidade: Precision e Recall

Para avaliar a qualidade dos resultados foram utilizadas as medidas Precision e Recall definidas por (POWERS, 2007), adaptando-as para o contexto já mencionado. Assim sendo, a medida Precision contabilizará como verdadeiro positivo todos os SNP relatados pelo algoritmo e que estejam contidos em uma região de interesse[2]. Já o Recall vai desconsiderar SNP relatados que compreendem uma mesma região de interesse, ou seja, mesmo que o algoritmo retorne quatro SNP corretos em uma mesma região apenas um deles será considerado correto visto que pertecem a uma mesma região.

$$\frac{TP}{TP+FP} \qquad \frac{TP}{TP+FN}$$

(a) Precision (b) Recall

Figura 2.6 – Medidas de qualidade.

2.4 Resultados e discussão

Para avaliar os resultados obtidos nos experimentos primeiramente é preciso entender qual tipo de informação um especialista do domínio está interessado em extrair quando se realiza a mineração de dados genéticos. Conforme já discutido anteriormente, pode-se considerar que que os marcadores SNP estão distribuídos de forma contínua e homogeneamente espaçada por todo o genoma de um indivíduo. Dessa maneira, pode-se considerar que marcadores vizinhos pertencem a uma mesma região no genoma. A informação na qual o especialista do domínio está interessado é a posição no genoma que tem efeito sobre o fenótipo. Essa posição pode ser estimada pela associação do fenótipo a um

[2] Definida por um raio de 5 SNP a partir do SNP causador.

ou mais marcadores SNP. Nesse contexto é válido afirmar que qualquer SNP que esteja em uma região próxima, ou em desequilíbrio de ligação com a região do genoma que tem efeito sobre o fenótipo é tida como resposta válida.

Comparando os resultados dos 5 melhores SNPs (TOP 5 SNP) ranqueados pelos algoritmos não-paramétricos com os resultados dos métodos tradicionais paramétricos (Tabelas 2.4 e 2.5) percebe-se que mesmo sem qualquer adequação para interpretar os dados genéticos e sem qualquer outra informação genética, por exemplo *pedigree*, foram obtidos resultados compatíveis ao contexto e bastante semelhantes aos softwares especializados da área. Essa tendência já foi observada em outros estudos (HOWARD; CARRIQUIRY; BEAVIS, 2014).

Tabela 2.4 – Desempenho dos algoritmos no QTL 2011.

Método	TOP 5 SNP	Precisão	Padrões Genéticos
InfoGain	60, 58, 59, 21, 71	60%	1
SSF	60, 71, 153, 4407, 2126	20%	1
Janela Desl.	53, 52, 56, 57, 51	80%	1
PLINK Single	58, 59, 60, 71, 89	60%	1
PLINK SNP Blocks	60, 58, 59, 21, 71	20%	1
GenSel	50, 30, 4106, 3629, 4088	0%	0

Tabela 2.5 – Desempenho dos algoritmos no QTL 2012.

Método	TOP 5 SNP	Precisão	Padrões Genéticos
InfoGain	6499, 1697, 1683, 1682, 528	60%	2
SSF	6499, 1683, 6571, 4674, 9390	40%	2
Janela Desl.	6498, 6497, 6499, 1683, 9592	80%	2
PLINK Single SNP	6499, 6506, 1682, 1683, 6507	60%	3
PLINK SNP Blocks	1683, 1253, 4675, 7283, 1613	20%	1
GenSel	6491, 1691, 1171, 291, 2671	40%	2

Nota-se que a maioria dos algoritmos encontraram os padrões genéticos mais representativos dos dados simulados, por exemplo a região dos SNP 6499-6500 do QTL-MAS 2012 e a região

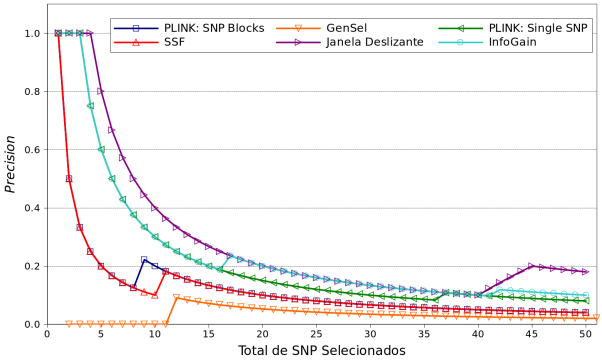
do SNP 57 do QTL-MAS 2011. Algoritmos que utilizam blocos de SNP, no entanto, foram penalizados pela metodologia adotada para avaliar os resultados, como é o caso do GenSel, no qual apesar de ter relatado um bloco de SNP (40 ao 60) que compreendia um padrão genético no QTL-MAS 2011, acabou não sendo contabilizado como acerto, pois foi considerado apenas o SNP médio do bloco, no caso o SNP 50, por exemplo.

Em relação aos gráficos das Figuras 2.7 e 2.8 observa-se que alguns algoritmos apresentam bons resultados em relação ao Precision mas não os mantêm em relação ao Recall, como é o caso da janela deslizante, e vice-versa, como no caso do GenSel. No entanto o algoritmo InfoGain que realiza cálculos mais simples que os demais algoritmos e tem um tempo de execução praticamente instantâneo, apresentou resultados satisfatórios para ambas medidas, mostrando-se uma excelente alternativa para análises cujo tempo de execução é crítico.

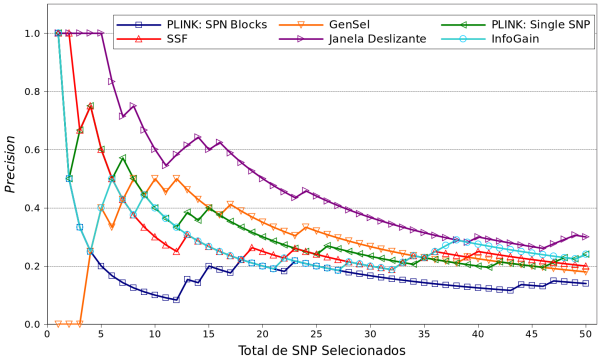
O comportamento dos gráficos de Precision na Figura 2.7 indica que a maioria dos algoritmos tendem a ter uma confiabilidade maior em relação aos atributos mais bem ranqueados durante a seleção, ou seja, quanto mais SNPs selecionados menor é a acurácia deles. Já em relação aos gráficos de Recall o comportamento é o contrário pois conforme o número de SNP selecionados aumentam, maior é a chance de se encontrar um padrão genético.

2.5 Considerações finais

De modo geral todos os algoritmos não-paramétricos apresentaram resultados e tendências semelhantes aos algoritmos paramétricos em relação à seleção dos SNP. Nosso trabalho segue essa tendência e mostra que os métodos não-paramétricos podem ser uma alternativa viável e auxiliar aos métodos tradicionais.

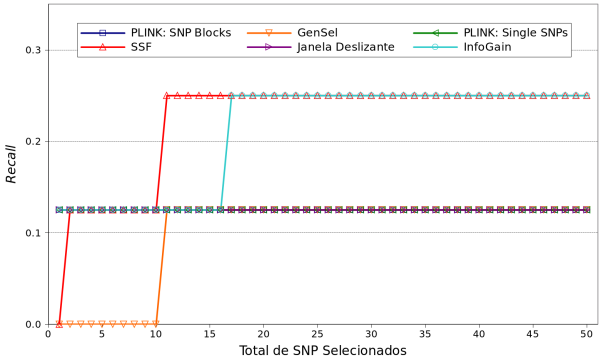


(a) QTL-MAS 2011

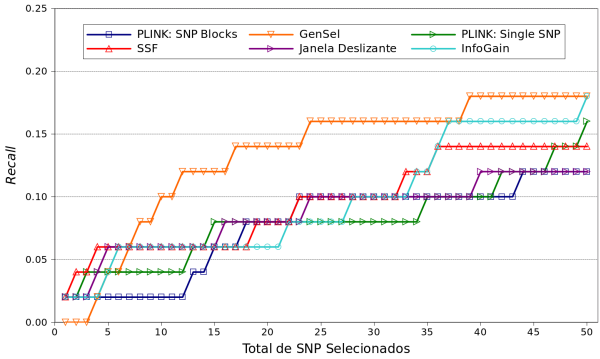


(b) QTL-MAS 2012

Figura 2.7 – Comparação do Precision com diferentes métodos.



(a) QTL-MAS 2011



(b) QTL-MAS 2012

Figura 2.8 – Comparação do Recall com diferentes métodos.

Agradecimentos

Os autores gostariam de agradecer à Patrícia Tholon pela ajuda na parte teórica, à Priscila Neubern pela ajuda com o software GenSel e ao CNPq pelo apoio financeiro.

2.6 Referências

ABIEC. *Estatísticas: balanço da pecuária*. 2014. Disponível em: <http://www.abiec.com.br/>.

AU, W. et al. Attribute clustering for grouping, selection, and classification of gene expression data. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, IEEE Computer Society, Los Alamitos, CA, USA, v. 2, n. 2, p. 83–101, 2005. ISSN 1545-5963.

BALDING, D. J. A tutorial on statistical methods for population association studies. *Nat. Rev. Genet.*, v. 7, n. 10, p. 781–791, Oct 2006.

BARRETT, J. C. et al. Haploview: analysis and visualization of LD and haplotype maps. *Bioinformatics*, v. 21, n. 2, p. 263–265, Jan 2005.

BELLMAN, R. *Adaptive control processes - A guided tour*. Princeton, New Jersey, U.S.A.: Princeton University Press, 1961. 255 p.

BIGUS, J. P. *Data mining with neural networks: solving business problems from application development to decision support*. Highstown: McGraw-Hill, Inc., 1996.

BROWNING, S. R.; BROWNING, B. L. Rapid and accurate haplotype phasing and missing-data inference for whole-genome

association studies by use of localized haplotype clustering. *Am. J. Hum. Genet.*, v. 81, n. 5, p. 1084–1097, Nov 2007.

CAMPOS, G. de L. et al. Whole-genome regression and prediction methods applied to plant and animal breeding. *Genetics*, v. 193, n. 2, p. 327–345, Feb 2013.

CANTOR, R. M.; LANGE, K.; SINSHEIMER, J. S. Prioritizing GWAS results: A review of statistical methods and recommendations for their application. *Am. J. Hum. Genet.*, v. 86, n. 1, p. 6–22, Jan 2010.

CLARKE, G. M. et al. Basic statistical analysis in genetic case-control studies. *Nat Protoc*, v. 6, n. 2, p. 121–133, Feb 2011.

COVER, T. M.; THOMAS, J. A. *Elements of information theory (2. ed.)*. Wiley, 2006. I–XXIII, 1–748 p. ISBN 978-0-471-24195-9. Disponível em: <http://www.elementsofinformationtheory.com/>.

COVOES, T. F.; HRUSCHKA, E. Towards improving cluster-based feature selection with a simplified silhouette filter. *Information Sciences*, v. 181, n. 18, p. 3766 – 3782, 2011. ISSN 0020-0255.

DAETWYLER, H. D. et al. Genomic prediction in animals and plants: simulation of data, validation, reporting, and benchmarking. *Genetics*, v. 193, n. 2, p. 347–365, Feb 2013.

FALCONER, D.; MACKAY, T. *Introduction to Quantitative Genetics*. [S.l.]: Longman, 1996.

FAYYAD, U. M.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From data mining to knowledge discovery: An overview. In: *Advances in Knowledge Discovery and Data Mining*. [S.l.: s.n.], 1996. p. 1–34.

FERNANDO, R.; GARRICK, D. *GenSel - User Manual for a portfolia of Genomic Selection related Analyses*. Iowa State

University Animal Breeding & Genetics, 2009. Disponível em: <http://www.biomedcentral.com/content/supplementary/1471-2105-12-186-s1.pdf>.

GRIFFITHS, A. J. F. et al. *Introduction to Genetic Analysis*. 9. ed. [S.l.]: W. H. Freeman, 2007. Hardcover. ISBN 0-7167-6887-9.

GUYON, I.; ELISSEEFF, A. An introduction to variable and feature selection. *J. Mach. Learn. Res.*, JMLR.org, v. 3, p. 1157–1182, 2003. ISSN 1532-4435.

HALL, D. *Dna-SNP*. Wikipedia Commons, CC BY 2.5, 2007. Acesso em: 11 set. 2014. Disponível em: <http://upload.wikimedia.org/wikipedia/commons/2/2e/Dna-SNP.svg>.

HAYES, B. J. et al. Invited review: Genomic selection in dairy cattle: progress and challenges. *J. Dairy Sci.*, v. 92, n. 2, p. 433–443, Feb 2009.

HOWARD, R.; CARRIQUIRY, A. L.; BEAVIS, W. D. Parametric and nonparametric statistical methods for genomic selection of traits with additive and epistatic genetic architectures. *G3 (Bethesda)*, v. 4, n. 6, p. 1027–1046, Jun 2014.

LIU, Y. et al. Bos taurus genome assembly. *BMC Genomics*, v. 10, n. 1, p. 180, 2009.

MEUWISSEN, T. H.; HAYES, B. J.; GODDARD, M. E. Prediction of total genetic value using genome-wide dense marker maps. *Genetics*, v. 157, n. 4, p. 1819–1829, Apr 2001.

MITRA, P.; MURTHY, C.; PAL, S. K. Unsupervised feature selection using feature similarity. *IEEE Trans. Pattern Anal. Mach. Intell.*, IEEE Computer Society, Washington, DC, USA, v. 24, n. 3, p. 301–312, 2002. ISSN 0162-8828.

MOORE, J. H.; ASSELBERGS, F. W.; WILLIAMS, S. M. Bioinformatics challenges for genome-wide association studies. *Bioinformatics*, v. 26, n. 4, p. 445–455, Feb 2010.

PANICO, B. et al. *QTL-MAS Workshop 2012*. Kassiopea Group srl, 2012. Acesso em: 11 set. 2014. Disponível em: <http://qtl-mas-2012.kassiopeagroup.com/en/index.php>.

POWERS, D. M. W. *Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation*. Adelaide, Australia, 2007.

PURCELL, S. et al. PLINK: a tool set for whole-genome association and population-based linkage analyses. *Am. J. Hum. Genet.*, v. 81, n. 3, p. 559–575, Sep 2007.

ROLF, M. M. et al. Genomics in the united states beef industry. *Livestock Science*, v. 166, n. 0, p. 84 – 93, 2014. Genomics Applied to Livestock Production. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1871141314003138>.

ROY, P. L. et al. *QTL-MAS Workshop 2011*. The Animal Genetics Division at INRA, 2011. Acesso em: 11 set. 2014. Disponível em: <https://colloque4.inra.fr/qtlmas>.

SHAH, S. C.; KUSIAK, A. Data mining and genetic algorithm based gene/SNP selection. *Artificial Intelligence in Medicine*, v. 31, n. 3, p. 183–196, 2004. Disponível em: <http://dx.doi.org/10.1016/j.artmed.2004.04.002>.

WANG, J.; ZUCKER; JEAN-DANIEL. Solving multiple-instance problem: A lazy learning approach. In: LANGLEY, P. (Ed.). *17th International Conference on Machine Learning*. [S.l.: s.n.], 2000. p. 1119–1125.

WITTEN, I.; FRANK, E. *Data Mining: Practical Machine Learning Tools and Techniques*. 2. ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2005. ISBN 0120884070.

ZHANG, Z.; ZHANG, Q.; DING, X. Advances in genomic selection in domestic animals. *Chinese Science Bulletin*, Chinese Science Bulletin, v. 56, n. 25, p. 2655, 2011. Disponível em: <http://csb.scichina.com:8080/kxtbe/EN/abstract/article_504345.shtml>.

3

Estudo de associação genômica ampla utilizando Random Forest: estudo de caso em bovinos de corte

Roberto H. Higa
Fabiana B. Mokry
Maurício de A. Mudadu
Francisco P. Lobo
Luciana C. A. Regitano

A APLICAÇÃO de técnicas de aprendizado de máquina, que consideram o efeito de múltiplos SNPs a problemas de estudos de associação em genônica ampla (GWAS) é de grande interesse, pois essas técnicas são potencialmente capazes de identificar variantes onde o modelo causal é desconhecido e de lidar com o problema de alta dimensionalidade dos dados. Nesse cenário, Random Forest (RF) desponta como uma das técnicas mais interessantes, devido à sua simplicidade, flexibilidade, escalabilidade e capacidade de lidar com um grande número de variáveis de entrada sem incorrer em sobre-ajuste. Embora, RF já seja utilizada em GWAS em humanos, na área de ciência animal, sua utilização ainda é muito tímida. O objetivo deste trabalho é demonstrar o potencial de aplicação de

RF em GWAS na área de ciência animal, apresentando estudos de casos que utilizam dados simulados e reais.

3.1 Introdução

O sequenciamento do genoma bovino, com o subsequente mapeamento de haplótipos e descoberta de milhões de marcadores SNP (*single nucleotide polymorphisms*), distribuídos ao longo do genoma (BOVINE GENOME SEQUENCING; ANALYSIS CONSORTIUM, 2009; BOVINE HAPMAP CONSORTIUM, 2009), viabilizou a construção de painéis de genotipagem em escala genômica, contendo centenas de milhares de SNPs, e a sua utilização em estudos visando desvendar as bases genéticas de características de interesse econômico em bovinos. Esses estudos, denominados estudos de associação genômica ampla (GWAS), consistem em analisar genótipos e fenótipos de uma amostra de poucos milhares de indivíduos ou menos, procurando identificar regiões do genoma onde ocorrem variações, em uma população, associadas ao fenótipo estudado.

Em função da complexidade dos mecanismos moleculares que controlam a manifestação de fenótipos, há grande interesse em abordar GWAS por meio de técnicas multivariadas, que considerem o efeito de múltiplos SNPs. Nesse sentido, é crescente a utilização de técnicas de aprendizado de máquina e mineração de dados, principalmente em estudos envolvendo doenças em humanos (BUREAU et al., 2005; ZIEGLER; DESTEFANO; KÖNIG, 2007; SCHWARZ et al., 2007; SZYMCZAK et al., 2009; MOORE; ASSELBERGS; WILLIAMS, 2010; GOLDSTEIN et al., 2010).

Como essas técnicas não dependem do pressuposto de que o mecanismo genético subjacente assume um determinado modelo, elas são potencialmente mais apropriadas para identificar variantes, onde o mecanismo causal é desconhecido e envolve

diversos SNPs. Além disso, muitas dessas técnicas foram desenvolvidas para lidar com o problema de alta dimensionalidade dos dados, quando o número de variáveis p é muito maior que o número de observações n, $p >> n$, que é o caso de GWAS.

Random Forest (RF), uma técnica de aprendizado de máquina que constrói preditores a partir de conjuntos de árvores de classificação e regressão (CART) (BREIMAN, 2001b), figura entre as mais poderosas técnicas de aprendizado de máquina disponíveis, com estudos empíricos mostrando desempenho igual ou superior a técnicas populares como Boosting (FREUND; SHAPIRE, 1996) e máquinas de vetores de suporte (SHAWE-TAYLOR; CRISTIANINI, 2004).

Dentre suas características mais atraentes estão a sua simplicidade, seu potencial de paralelização e a capacidade de lidar com um grande número de variáveis de entrada sem incorrer em sobre-ajuste.

RF foi introduzida em GWAS em humanos por Bureau et al. (2005) e, desde então, apresenta uma lenta, mas crescente tendência de adoção e utilização (GOLDSTEIN et al., 2010). Em estudos aplicados a ciência animal, sua utilização, embora ainda pequena, também encontra-se em crescimento tanto em GWAS (MOKRY et al., 2013; YAO et al., 2011) quanto em seleção genômica (GWS) (GONZÁLEZ-RECIO; FORNI, 2011).

O objetivo deste trabalho é demonstrar o potencial de utilização de RF em GWAS na área de ciência animal. Para isso, são apresentados os resultados de dois estudos, o primeiro deles envolvendo conjuntos de dados simulados públicos, que se assemelham a populações típicas de melhoramento animal (ELSEN et al., 2012; QTL-MAS, 2012); e o segundo envolvendo um conjunto de dados reais obtidos de uma população de melhoramento de gado de corte (MOKRY et al., 2013). A apresentação desses estudos

é precedida pela exposição dos conceitos básicos envolvidos na construção de RF.

3.2 Random Forest

RF integra um conjunto de métodos de aprendizado de máquina que envolve a construção de muitos preditores (classificadores ou regressores) e cuja predição consiste na agregação das predições de todos os preditores do conjunto. O método foi inicialmente proposto por Breiman (2001b) como uma extensão de seus trabalhos anteriores com árvores CART (BREIMAN et al., 1984) e *bootstrap and aggregating - bagging* (BREIMAN, 2001a), também influenciado por trabalhos como de Ho (1998) e de Amit e Geman (1997), que construíram conjuntos de árvores aleatórias para problemas de classificação.

O preditor base utilizado para construção de RF é a árvore CART (BREIMAN et al., 1984). Considerando uma variável resposta Y e um conjunto de variáveis preditoras, $X_1, X_2, \ldots, X_p$, uma árvore CART é construída particionando-se, sucessivamente, o espaço de atributos, utilizando a cada passo uma única variável preditora, X_i para gerar dois sub-espaços. A raíz da árvore representa todo o espaço de atributos e cada particionamento corresponde a um nó interno da árvore. Ao particionar um nó interno da árvore, dois novos nós, e os respectivos sub-espaços associados, são criados. Ao final do processo, obtém-se uma árvore binária, que é utilizada para fazer predições por meio de um processo de busca: considerando um novo dado, $(X_1, X_2, \ldots, X_p)$, a cada nó, é realizado um teste (usando X_i) tal que, dependendo do resultado do teste, a busca prossegue pelo ramo direito ou esquerdo, até que se encontre um nó folha, onde se realiza a predição de Y. O valor da variável predita, Y, baseia-se nos valores de Y das amostras do conjunto de treinamento associadas ao nó folha. Para problemas

de regressão, a predição será igual à média dos valores de Y e, no caso de problemas de classificação, ao valor de Y mais frequente.

Bagging (BREIMAN, 2001a) é uma técnica para construção de conjuntos de preditores, construídos sucessivamente de forma independente, utilizando uma amostra *bootstrap* do conjunto de dados de treinamento. Quando comparado com o preditor base, pode-se mostrar que preditores *bagging* apresentam menor erro de predição por meio da redução do componente de variância do erro. Mas, na prática, essa redução no erro de variância é limitada pela correlação entre os preditores. Por isso, RF introduz em *bagging* um elemento adicional de aleatoriedade, visando obter um conjunto de preditores menos correlacionados: durante a construção de uma árvore CART (preditor), ao invés de analisar todas as variáveis p para determinar o melhor particionamento de um nó da árvore, um número menor de variáveis, $m < p$, selecionados de forma aleatória, é examinado.

Duas das vantagens de RF são a sua simplicidade e o número reduzido de parâmetros importantes – *ntree*, o número de árvores na floresta, e *mtry*, o número de variáveis utilizadas para particionar os nós das árvores –, além de ser robusta às variações destes valores (LIAW; WIENER, 2012). Formalmente, o algoritmo para construção de RF pode ser descrito pelo Algoritmo 1 (BREIMAN, 2001b):

Algoritmo 1: Random Forest (ntree, mtry)

```
1  para b ← 1 ... ntree faça
2      selecionar uma amostra bootstrap;
3      repita
4          selecionar aleatoriamente mtry variáveis;
5          encontrar o resultado das melhores partições;
6      até formar a árvore;
7      predizer Y para OOB;
8      predizer Y para X permutado;
9  fim para
10 calcular o erro OOB;
11 calcular a importância das variáveis;
```

Outra vantagem de RF, comparado com outros métodos de aprendizado de máquina, é que, por construção, ele possui mecanismos para estimar tanto o erro de predição quanto a importância das variáveis analisadas (Figura 3.1). RF constrói essas estimativas avaliando as amostras do conjunto de dados de treinamento que não são incluídas no conjunto de amostras *bootstrap*, dados *out-of-bag* (OOB), que em média representa 36% das amostras de treinamento.

Cada árvore construída é utilizada por RF para predizer os valores de Y para os correspondentes dados OOB; então, após finalizar a construção da floresta, essas predições são comparadas com os valores verdadeiros para obter uma estimativa de erro, denominada erro OOB. De forma similar, os dados OOB também são utilizados para identificar as variáveis importantes calculando os erros quando se permuta cada uma das variáveis utilizadas na construção de cada árvore e comparando-os com o erro OOB. A importância de uma variável é medida pelo impacto que a retirada da informação que ela traz (permutação) causa no erro de predição OOB.

Finalmente, diferentes implementações de RF estão disponíveis livremente (BREIMAN; CUTLER, 2013; LIAW; WIENER, 2012; SCIKIT, 2013; FASTRF, 2013; PARF, 2013; ZHANG; WANG; CHEN, 2009; SCHWARZ; KÖNIG; ZIEGLER, 2010), apresentando características bastante diversas. Algumas estão implementadas em Fortran (BREIMAN; CUTLER, 2013; LIAW; WIENER, 2012; PARF, 2013), enquanto outras em C/C++ (SCIKIT, 2013; ZHANG; WANG; CHEN, 2009; SCHWARZ; KÖNIG; ZIEGLER, 2010) ou mesmo em Java (FASTRF, 2013); algumas são utilizadas por linha de comando (BREIMAN; CUTLER, 2013; PARF, 2013; ZHANG; WANG; CHEN, 2009; SCHWARZ; KÖNIG; ZIEGLER, 2010) enquanto outras são utilizadas dentro de ambientes de programação como R (LIAW; WIENER, 2012) ou Python (SCIKIT, 2013); algu-

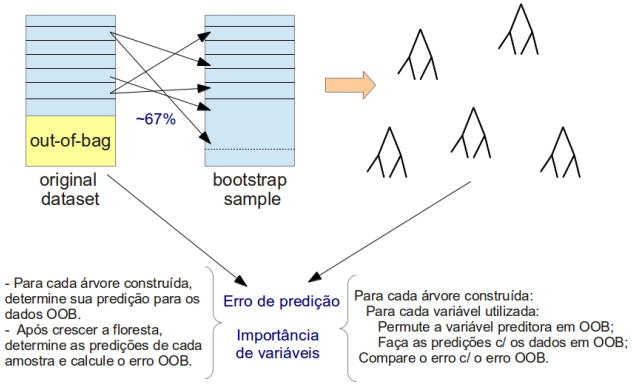


Figura 3.1 – Procedimento embutido em RF para estimar o erro OOB e da importância das variáveis.

mas exploram mecanismos de paralelismo em suas implementações (PARF, 2013; SCHWARZ; KÖNIG; ZIEGLER, 2010), enquanto algumas foram implementadas especificamente para aplicação em GWAS (ZHANG; WANG; CHEN, 2009; SCHWARZ; KÖNIG; ZIEGLER, 2010).

3.3 Estudo com dados simulados

O objetivo do estudo apresentado nesta seção é utilizar dados simulados para avaliar a capacidade de RF em identificar as regiões de QTL (*quantitative trait loci*) (regiões associadas a características quantitativas de interesse relacionadas ao melhoramento animal), utilizando seu mecanismo de mensuração de importância de variáveis, ou seja, verificando se SNPs próximos às regiões de QTL são ranqueados entre as variáveis mais importantes reportadas por RF. Para isso, foram utilizados os conjuntos de dados simulados utilizados nas edições de 2011 e 2012 do *workshop* QTL-MAS, (ELSEN et al., 2012; QTL-MAS, 2012) para avaliar novas

metodologias de GWAS e GWS, utilizando dados com as características encontradas em populações de animais utilizados em programas de melhoramento. Especificamente, os dados de 2011 representam uma população típica de melhoramento de suínos, enquanto os dados de 2012 uma população típica de melhoramento de gado de leite.

No *workshop* QTL-MAS 2011, a população utilizada consistia de 20 famílias de machos não independentes, onde cada macho foi acasalado com 10 fêmeas. Cada fêmea foi acasalada com apenas um macho e gerou dois grupos de progênies, um com 10 indivíduos e outro com 5. O primeiro grupo, contendo 2.000 indivíduos, constituiu a população experimental e continha tanto o genótipo quanto o fenótipo, enquanto o segundo grupo, com 1.000 animais, constituiu o grupo de seleção e continha apenas informação sobre genótipo. Para ambos os grupos também estava disponível o valor genético verdadeiro ou *true breeding value* (TBV). A geração parental de 20 machos e 200 fêmeas foi gerada, a partir da escolha aleatória de 2 gametas de conjuntos de 75 gerados utilizando o software LDSO (YTOURNEL et al., 2012).

A simulação consistiu de dois passos: 1.000 gerações de uma população de 1.000 gametas, seguida de uma forte redução da população (*bottleneck*) em que 150 gametas evoluíram por 30 gerações. A estrutura do genoma considerado contém 5 cromossomos (autossomos) de comprimento igual a 1 Morgan. Em cada cromossomo foram simulados 1.998 SNPs localizados a cada 0,05 cM, resultando em um painel com 9.990 SNPs. Um conjunto de 1.000 gametas foi inicialmente gerado em equilíbrio de ligação durante as 1.150 gerações simuladas, considerando-se uma taxa de mutação de 0,0002 (ELSEN et al., 2012). Foram simulados 8 QTLs, cuja segregação acrescida de ruído ambiental constituíram a variação do fenótipo. Esta variação foi ajustada para representar uma herdabilidade de 0,3 e as características dos QTLs nos diferen-

tes cromossomos escolhidas para representar situações extremas (Tabela 3.1).

Tabela 3.1 – Efeito dos QTLs nos dados do QTL-MAS 2011 (ELSEN et al., 2012). Crom: cromossomo.

QTL	Crom	Pos (cM)	Tipo				
QTL1	1	2.85	4 alelos, aditivo, grande. Alelo 1 = 0., 2 = 2., 3 = 4., 4= 6.				
					11	12	22
QTL2	2	81.9	em fase c/ QTL3	11	-4.	-2.	0.
QTL3		93.75	em fase c/ QTL2	12	-2.	0.	2.
				22	0.	2.	4.
					11	12	22
QTL4	3	5.0	em oposição c/ QTL5	11	0.	2.	4.
QTL5		15.0	em oposição c/ QTL4	12	-2.	0.	2.
				22	-4.	-2.	0.
				11	11	12	22
QTL6	4	32.2	imprinted	2.0	0.0	0.0	0.0
					11	12	22
QTL7	5	36.3	epistasia c/ QTL8	11	2.	1.	0.
QTL8		99.2	epistaisa c/ QTL7	12	0.	0.	0.
				22	0.	0.	0.

Já no *workshop* de 2012, uma população base, *G0*, com 1.020 indivíduos não relacionados, sendo 20 machos e 1.000 fêmeas, foi gerada com 5 cromossomos (autossomos), cada um com comprimento de 99,95 Mb, resultando em um genoma de comprimento 499,750 Mb. Em cada cromossomo, foram distribuídos um conjunto de 2.000 SNPs igualmente espaçados a cada 0,05 Mb (ou cM). A população base foi gerada de forma a obter um decaimento de desequilíbrio de ligação (LD) e distribuição de frequência alélica mínima (MAF) fixos. A partir de *G0* foram geradas 4 gerações, *G1* – *G4*, não sobrepostas, formadas por 20 machos e 1.000 fêmeas por acasalamento aleatório entre cada macho com 51 fêmeas. Cada fêmea gerou 1 fêmea, exceto as mães dos machos da próxima geração que geraram 1 macho e 1 fêmea. Três características correlacionadas para emular produtividade foram geradas, com efeitos distribuídos entre 50 QTLs distribuídos ao longo do genoma (Fi-

gura 3.2). Neste evento, o desafio consistiu em encontrar os QTLs e a ação em pleitropia entre eles (YTOURNEL et al., 2012). Aqui, para avaliação de RF, apenas o fenótipo 1 será utilizado.

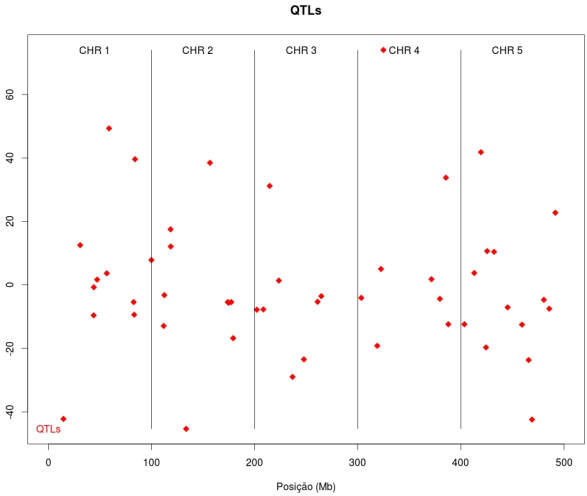


Figura 3.2 – Localização dos 50 QTLs com efeito para o fenótipo 1, ao longo do genoma, para os dados do QTL-MAS 2012. Os efeito dos QTLs são apresentados no eixo y.

Observe que, em ambos os casos, os QTLs com efeito não integram o painel utilizado para genotipar os animais (Figura 3.3).

Figura 3.3 – Ilustração do painel utilizado para genotipagem (QTL-MAS 2011). Os QTLs não integram o painel.

O pressuposto é que tanto o número de QTLs, quanto suas posições ao longo do genoma são desconhecidas, mas que um número pré-determinado, k, de SNPs no topo da lista dos reportados como variáveis importantes serão selecionados como implicados com o fenótipo analisado. Essa é a situação típica que se encontra ao analisar dados reais e, por essa razão o critério de avaliação utilizado para avaliar RF é a medida de precisão (Expressão 3.1):

$$\text{precisão} = \frac{\text{\#SNPs corretos}}{\text{\#SNPs preditos}} \tag{3.1}$$

onde um SNP predito é considerado correto se ele está a uma distância inferior a 0,25 cM (dados de 2011) ou 250 kb (dados de 2012). Os SNPs preditos são os k SNPs selecionados por RF, com os valores de k variando entre 10 e 50. A medida de precisão avalia a proporção de SNPs corretos dentre o conjunto de k preditos.

Em todos os casos, utilizou-se RF com o parâmetro mtry = 0,4p – onde p é o número de SNPs – e ntree = 2.000. Para os demais parâmetros utilizou-se os valores *default* do pacote R randomForest (LIAW; WIENER, 2012).

As Figuras 3.4 e 3.5 mostram os gráficos da precisão em função do número k de SNPs considerados como preditos por RF para os conjuntos de dados do *workshop* QTL-MAS 2011 e 2012. Em ambos os casos, a precisão apresenta um comportamento similar: quando $k = 10$, a precisão é igual a 0,5 para os dados de 2011 e 0,4 para os dados de 2012 e decaem a medida que o valor de k aumenta até 50. A curva contínua em cinza representa a precisão esperada quando a predição consiste em selecionar aleatoriamente k SNPs, enquanto a linha pontilhada cinza representa este valor acrescido de 4 desvios padrão. As Figuras 3.4 e 3.5 representam visualmente o quão estatisticamente significante é o resultado obtido.

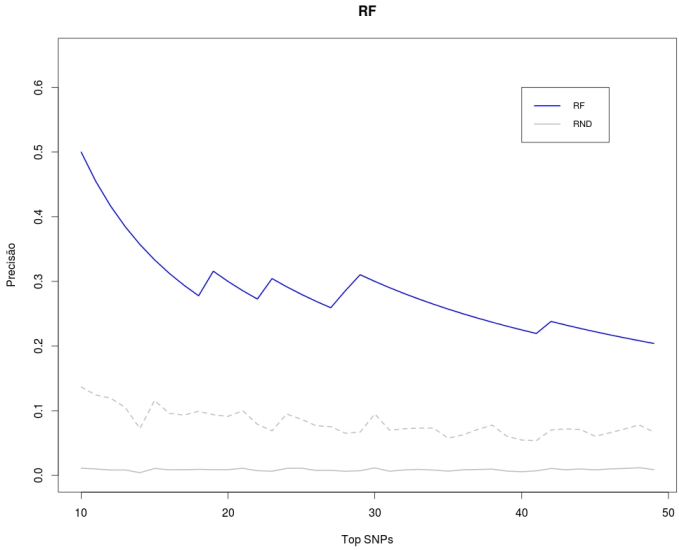


Figura 3.4 – Precisão em função do número de SNPs selecionados por RF para dados do QTL-MAS 2011.

A medida de precisão aqui utilizada considera corretos apenas os 5 SNPs mais próximos em cada lado do QTL, sendo todos os outros considerados incorretos. Para verificar o quão distante dos QTLs estão os outros SNPs preditos por RF, as Figuras 3.6 e 3.7 apresentam suas localizações no genoma, considerando sua posição na lista de ranqueamento. Na Figura 3.6 é possível notar que as predições que se encontram no topo do lista para os dados do QTL-MAS 2011 correspondem a SNPs próximos do QTL 1, que é o QTL de maior efeito. Também é possível notar que todos os 10 SNPs no topo da lista de preditos estão bem próximos de um QTL (QTL 1), apesar de apenas 5 serem considerados corretos, de acordo com a condição imposta para o cálculo de precisão. Examinando os demais SNPs na lista de ranqueamento, nota-se que grupos de SNPs próximos das posições dos QTLs 2, 3, 4 e 5

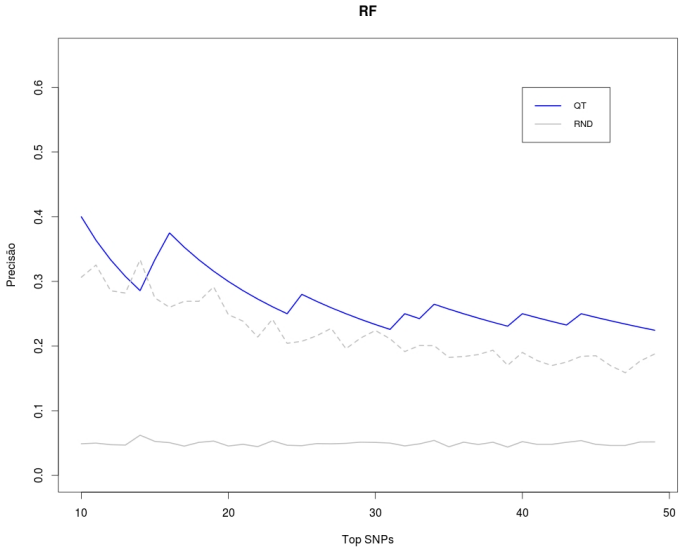


Figura 3.5 – Precisão em função do número de SNPs selecionados por RF para dados do QTL-MAS 2012.

nos cromossomos 2 e 3, mas o mesmo não ocorre para os QTLs nos cromossomos 4 e 5, que são QTLs com efeito de *imprinting* e epistasia.

A estrutura dos QTLs dos dados do QTL-MAS 2012 é bem diferente daquela exibida pelos dados do QTL-MAS 2011 (Tabela 3.1 e Figura 3.2). Enquanto os dados de 2011 apresentam apenas 8 QTLs com tipos de efeitos diversificados, os dados de 2012 contém 50 QTLs com efeitos aditivos espalhados ao longo do genoma. Contudo, da mesma forma que no caso dos dados do QTL-MAS 2011, a Figura 3.7 mostra que vários dos SNPs no topo da lista de preditos estão próximos de QTLs de grande efeito (nos cromossomos 1, 4 e 5). Com foco nos QTLs de maior efeito, isto é, $\text{efeito} \geq \text{abs}(20)$, nota-se uma concentração de SNPs preditos em torno da maioria dos QTLs (8 de 12), evidenciando a mesma

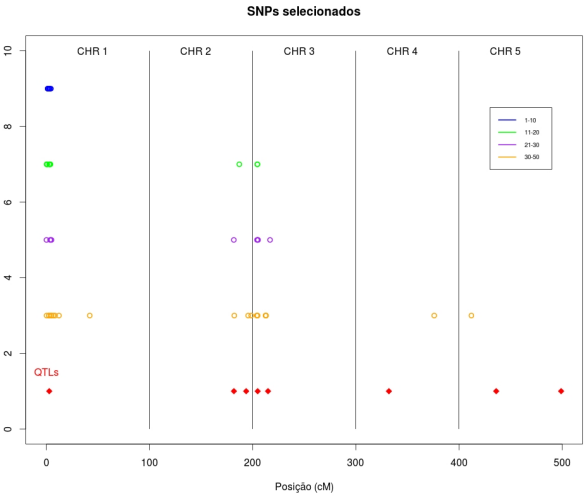


Figura 3.6 – Localização dos SNPs selecionados por RF para os dados de QTL-MAS 2011.

tendência exibida para os dados de 2011 de viés para QTLs de maior efeito.

Em função da alta dimensionalidade dos dados que, neste caso é de 10.000 SNPs, mas que para dados reais pode chegar a centenas de milhares de SNPs – por exemplo, 770.000 SNPs –, é conveniente paralelizar a tarefa de predição de SNPs associados ao fenótipo. Por construção, RF apresenta dois pontos que se prestam à paralelização da implementação: (i) o processo de crescimento das árvores de classificação/regressão e (ii) o processo de avaliação de variáveis para particionamento de um nó de uma árvore (vide Seção 3.2). Entretanto, dada a divisão natural do genoma em cromossomos, é possível pensar em executar RF em dois passos. No primeiro passo, RF é utilizado separadamente, e em paralelo, para selecionar os t SNPs mais importantes em cada cromossomo; e, no segundo passo, RF é novamente utilizado para selecionar os

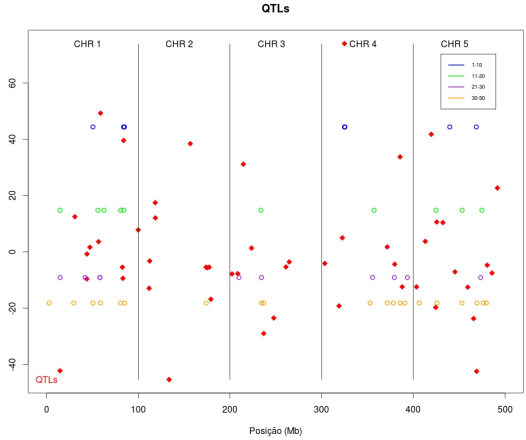


Figura 3.7 – Localização dos SNPs selecionados por RF para os dados de QTL-MAS 2012.

k SNPs mais importantes dentre a união dos conjuntos de t SNPs selecionados para cada cromossomo. Para avaliar o impacto da utilização desse processo empírico de pseudo-paralelização, as Figuras 3.8 e 3.9 apresentam os gráficos sobrepostos da precisão em função do número k de SNPs considerados como preditos por RF e por RF em dois passos para os conjuntos de dados do *workshop* QTL-MAS 2011 e 2012. As curvas de precisão utilizando RF e RF em dois passos apresentam o mesmo comportamento com níveis de precisão comparáveis e até com uma ligeira vantagem no caso dos dados do QTL-MAS 2012 (Figura 3.9).

Finalmente, mesmo não sendo o foco deste trabalho predizer os valores genéticos dos animais, em seguida comparou-se, por meio da correlação de Pearson, as estimativas obtidas utilizando RF e os 50 SNPs identificados como mais importantes com os valores de fenótipo e os TBVs fornecidos com os conjuntos de dados. No caso do fenótipo, as predições foram realizadas utilizando-se apenas as populações experimentais (2.000 animais para o ano

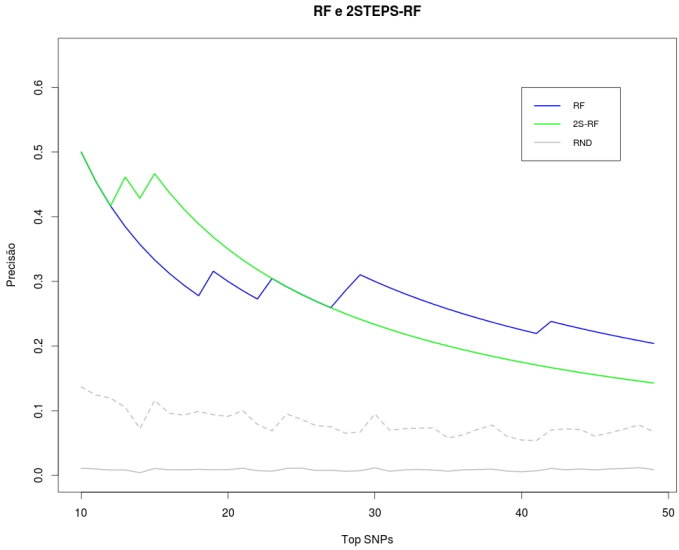


Figura 3.8 – Comparação entre a precisões em função do número de SNPs selecionados por RF e RF em dois passos para os dados do QTL-MAS 2011.

de 2011 e 3.000 animais para o ano de 2012, correspondentes às gerações, *G1 – G3*), por meio de um processo de validação cruzada com 10 partições. Já para comparação com os valores genéticos verdadeiros, foram utilizados os dados experimentais de cada ano para treinar RF e a estimativa feita para os animais do grupo de validação (1.000 animais para o ano de 2011 e 1.000 animais para o ano de 2012, correspondente aos animais da geração *G4*). Os valores de correlação obtidos em todos os casos são apresentados na Tabela 3.2 e são muito superiores aos valores de correlação obtidos utilizando-se o mesmo número de SNPs selecionados de forma aleatória ao longo do genoma. Relembrando que os SNPs foram selecionados baseados em dados de fenótipos obtidos pela combinação dos efeitos dos QTLs e ruído (aleatório) ambiental, é

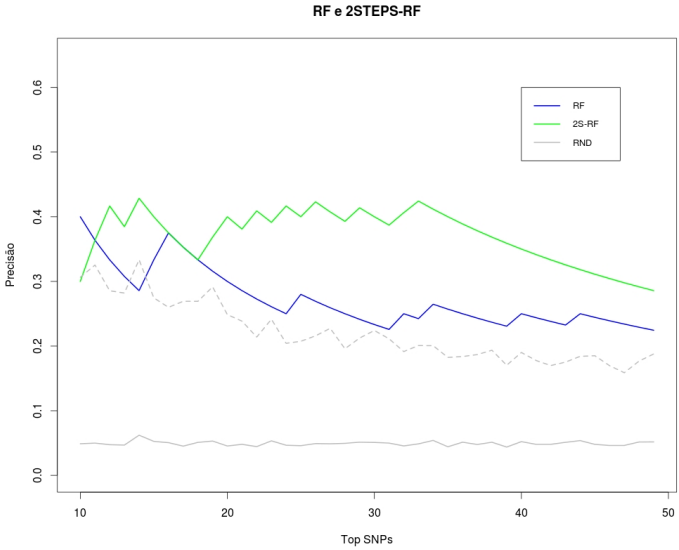


Figura 3.9 – Comparação entre a precisões em função do número de SNPs selecionados por RF e RF em dois passos para os dados do QTL-MAS 2012.

Tabela 3.2 – Correlação entre valores preditos e verdadeiros para fenótipo e valor genético verdadeiro (TBV).

	2011		2012	
	Fenótipo	TBV	Fenótipo	TBV
RF	0,49	0,73	0,44	0,74
RF em dois passos	0,51	0,77	0,45	0,70
aleatório	$0,27 \pm 0,04$	$0,5 \pm 0,05$	$0,20 \pm 0,03$	$0,31 \pm 0,05$

notável que o valor genético predito com os SNPs selecionados por RF apresentem uma correlação com o TBV acima de 0,70 em todos os casos.

3.4 Aplicação usando dados reais

A aplicação apresentada nesta seção é um estudo de associação para espessura de gordura em gado Canchim (MOKRY et al., 2013). A raça Canchim é uma raça sintética formada por $\frac{3}{8}$ Zebu e $\frac{5}{8}$ Charolês e desenvolvida no Brasil no início da década de 60 com a intenção de combinar as características de adaptação da raça Zebu e a eficiência de produção e qualidade de carne das raças taurinas (VIANNA, 1960).

Neste estudo, foram utilizados animais, registrados na Associação Brasileira de Criadores de Canchim, pertencentes a 7 diferentes rebanhos nos estados de São Paulo e Goiás. Uma amostra de 987 animais contendo machos e fêmeas, nascidos entre os anos de 2003 e 2005 e criados a pasto, foi avaliada para espessura de gordura por ultrassom *in vivo* sobre a 12ª costela. Os valores genéticos (EBVs) desses animais foram estimados por máxima verossimilhança restrita, usando o software MTDFREML (BOLDMAN et al., 1995). O modelo animal incluiu efeitos fixos de grupo de contemporâneos (sexo, ano, rebanho e grupo genético) e idade à época da medida como co-variáveis lineares, efeito genético aditivo e o erro. A partir desse conjunto de animais, foram selecionados 400, levando em consideração o EBVs, acurácia, tamanho de família e proporção entre machos e fêmeas. Esses animais foram, então, genotipados com o *chip* Illumina BovineHD, que contém 777.000 SNPs. Para o estudo de associação, considerou-se como fenótipo o valor genético derregredido (dEBV) (GARRICK; TAYLOR; FERNANDO, 2009), um pseudo fenótipo que leva em conta a matriz de *pedigree*, a herdabilidade estimada (0,16), o valor de EBV, suas correspondentes acurácias e o mesmo modelo animal descrito acima. Para a estimação de dEBV o conjunto de dados inicial foi suplementado com dados de animais nascidos entre 2005 e 2008,

totalizando 1.648 animais com fenótipos para espessura de gordura e 6.801 animais na matriz de *pedigree*.

Após um procedimento de controle de qualidade (*call rate* < 0,90 para amostras e SNPs; MAF < 0,01 e heterozigosidade < 3 desvios padrões), foram utilizados na sequência do estudo 396 animiais e 708.641 SNPs, com um *call rate* médio de 0,99. Para o estudo de associação entre SNPs e o dEBV foi utilizado o pacote randomForest (LIAW; WIENER, 2012) do software R (R CORE TEAM, 2013).

A estratégia utilizada foi a de RF em dois passos, conforme explicado na Seção 3.3. No primeiro passo, os 1% SNPs melhor ranqueados por RF em cada cromossomo foram selecionados e re-analisados em conjunto por RF em um segundo passo, novamente retendo os 1% SNPs melhor ranqueados. Foram utilizados os parâmetros ntree = 5.000 e mtry = 0,4p, enquanto que para os demais parâmetros foram adotados os valores *default* fornecidos pelo pacote randomForest. Os SNPs selecionados foram, então, utilizados no ajuste de um modelo de regressão *stepwise* usando o software SAS/STAT (SAS INSTITUTE INC., 2011) para estimar a quantidade de variância explicada pelo conjunto de SNPs selecionados. No final, 21 SNPs foram selecionados para compôr o modelo de regressão final (Tabela 3.3), explicando 53% da variância observada (R^2) no fenótipo (dEBV).

Uma segunda estratégia para utilização de RF consistiu em aplicar o procedimento acima descrito a 10 sub-amostras de 198 animais, construídas da seguinte forma: i) o primeiro animal foi escolhido de forma aleatória dentre os 396 animais genotipados; ii) o próximo animal foi escolhido considerando o menor parentesco com os animais previamente selecionados, mas representativo entre os demais animais genotipados; iii) o passo (ii) foi repetido até que 198 animais tivessem sido selecionados.

Tabela 3.3 – Os cinco primeiros SNPs selecionados ao ajustar o modelo animal. QTLs: SF – gordura subcutânea; MS – escore de marmoreio; FT12R – gordura subcutânea na 12^a costela; IF – gordura intramuscular; OAC – conteúdo de ácido oléico; PAC – conteúdo de ácido palmitoleico.

dbSNP	Crom	Pos	Genes	QTL
rs133046994	10	18129602	*THSD4, LRRC49*	SF, MS
rs137294146	1	132385787	*SOX14, CLDN18, DZP1L*	FT12R, IF
rs109349988	3	15814096	*KCNN3, EFNA3, EFNA4, DCST2, LOC100294774, PMVK, ADAR, CHRNB2, ADAM15, ZBTB7B, DCST1, LOC100294857, FLAD1, PYGO2, CKS1B, PBXIP1, SHC1, LOC100294894*	FT12R, MS
rs136717249	19	37969870	*B4GALNT2, GNGT2, ABI3, NGFR, GIP, PHOSPHO1, ZNF652, PHB, IGF2BP1*	OAC, PAC
rs134790147	13	20780821	*CCDC7, ARL5B, MGC152301, LOC100848675, LOC100847992*	FT12R

Os SNPs comuns entre a estratégia utilizando 398 animais e essas 10 sub-amostras foram, então, analisados para ajustar o mesmo modelo de regressão *stepwise* descrito acima. Neste caso, foram selecionados 19 SNPs que explicaram 50% da variância observada (R^2) no fenótipo (dEBV). Além disso, nas duas estratégias, os primeiros 5 SNPs do modelo de regressão foram os mesmos, com a mesma ordem de importância (rs133046994, rs137294146, rs109349988, rs136717249, rs134790147) e explicam 34% da variância observada (R^2) no fenótipo (dEBV).

Em seguida, os haplótipos para cada cromossomo foram reconstruídos utilizando o software fastPHASE, versão 1.4.0 (SCHEET; STEPHENS, 2006), e analisados por meio do software Haploview (BARRETT et al., 2005), usando seus parâmetros *default*. A estimação de blocos de haplótipos e LD baseou-se no coeficiente de correlação quadrado entre pares de SNPs (r^2). Assumiu-se o valor médio de r^2 = 0,12 como determinante da extensão de LD, tal que a distância de 250 kb em torno de cada SNP foi considerado para descobrir genes candidatos.

Esses genes e as informações contidas nas bases de dados NCBI BioSystems database (GEER et al., 2010) e Kyoto Encyclopedia of Genes and Genomes (KEGG) (KANEHISA; GOTO, 2000; KANEHISA et al., 2012) foram utilizadas para obter informações sobre processos biológicos relacionados com espessura de gordura. Dentre os 21 SNPs identificados, um SNP (cromossomo 3: rs42021729) não possui genes descritos em sua vizinhança, enquanto dentre os outros 20, apenas 4 (cromossomo 12: rs136348926; cromossomo 11: rs110833507; cromossomo 2: rs42923911; cromossomo 9: rs110025080) não estão em regiões de QTL previamente descritas na literatura (MOKRY et al., 2013). Já quando os genes candidatos identificados são analisados, observa-se que diversos deles contém anotação relacionando-os com metabolismo de lipídeos, por exemplo:

- o gene *THSD4* codifica uma proteína com domínios de disintegrina e metaloprotease, que possui uma função importante no processo de adipogênese (MCDANIEL et al., 2006);

- o gene *PMVK* (*phosphomevalonate kinase*), que cataliza a conversão de mevalonato-5-fosfato e ATP para mevalonato-5-difosfato e ADP, que é uma das reações iniciais envolvidas na via de síntese de colesterol (HERDENDORF; MIZIORKO, 2007);

- o gene *ADAR* (*adenosine deaminase, RNA-specific*), que em um estudo com humanos, foi implicado com níveis de triglicérides, adiponectina, circunferência abdominal e índice de massa corporal (OGURO et al., 2012); e
- o gene *SHC*1 (*Src homology 2 domain containing – transforming protein 1*), que foi implicado com a obesidade em humanos (EIGELSON et al., 2008).

Diferente do que ocorre quando se analisa dados simulados (Seção 3.3), ao se analisar dados reais, as respostas (posições dos QTLs) não são conhecidas. Por essa razão, não é possível avaliar o quão próximos os SNPs selecionados estão de variações com efeito no fenótipoo. Contudo, a análise funcional de genes próximos dos principais SNPs revela que eles estão envolvidos com atividades biológicas relacionadas à síntese e acúmulo de gordura, o que confere ao resultado (lista de SNPs) uma alta plausabilidade. A lista completa dos SNPs selecionados, genes candidatos e suas relações com o fenótipo analisado podem ser encontrados em Mokry et al. (2013).

3.5 Discussão e conclusão

Para demonstrar a aplicabilidade de RF a problemas de GWAS em ciência animal, nas seções anteriores utilizou-se: (i) dados simulados com a estrutura genética tipicamente encontrada em populações utilizadas em programas de melhoramento animal; e (ii) dados reais de bovinos de corte, cujos resultados foram recentemente publicados pelo nosso grupo (MOKRY et al., 2013).

Os dados simulados foram obtidos dos sites do *workshop* QTL-MAS, anos de 2011 e 2012, e se assemelham à estrutura encontrada em populações de suínos e bovinos de leite, respectivamente. Esses conjuntos de dados foram gerados por terceiros, o

que elimina a possibilidade de enviesamento em favor da metodologia avaliada. RF foi avaliado quanto à sua capacidade de identificar os QTLs presentes nesses conjuntos de dados, resultando em uma precisão entre 40% e 50%, considerando os 10 primeiros SNPs reportados e uma tolerância de 5 SNPs de distância (250 kb) para a posição da variação causal. Esses resultados demonstram a capacidade de RF de encontrar SNPs relacionados ao fenótipo de interesse em situações em que os mecanismos moleculares que controlam os fenótipos são muito diversos.

Ainda considerando os dados simulados, utilizou-se os 50 SNPs selecionados por RF (50 primeiros no ranqueamento gerado por RF) para verificar a correlação do fenótipo estimado com esses SNPs e o valor genético verdadeiro. Este variou entre 0,7 e 0,77 em ambos os casos, o que é bastante significativo quando se considera que RF baseou a seleção dos SNPs na medida de fenótipo, que inclui o valor genético acrescido de ruído ambiental. Esses resultados demonstram a robustez de RF a ruídos presentes nos dados e a diferentes estruturas genéticas relacionadas com os QTLs estudados.

Apesar de sua natureza paralelizável, os conjuntos de dados simulados também foram utilizados para avaliar uma estratégia de aplicação de RF em dois passos, o que resulta em uma pseudo-paralelização. Os níveis de desempenho obtidos utilizando essa estratégia foram comparáveis aos obtidos anteriormente, demonstrando a flexibilidade de RF, e potencial de escalabilidade para problemas envolvendo milhares de amostras e genótipos de centenas de milhares de SNPs, que é o caso encontrado em GWAS.

RF também foi aplicado a um conjunto de dados reais, a espessura de gordura em Canchim, sendo que o conjunto de 21 SNPs selecionados explicou tanto quanto 50% da variância observada no fenótipo. Uma análise funcional das regiões a que pertencem os SNPs selecionados mostrou diversos genes em QTLs

relacionados a metabolismo envolvendo gordura e genes implicados com processos biológicos como adipogênese e metabolismo de lipídeos ou associados a fenótipos fortemente relacionados a acúmulo de goudura como obesidade em humanos. Esses resultados são bastante plausíveis à luz da característica estudada, a espessura de gordura, e constitui mais uma evidência do potencial de aplicação de RF em GWAS na área de ciência animal.

Os resultados obtidos nos estudos apresentados fornecem evidências que suportam a adequabilidade da utilização de RF para GWAS, especificamente aplicado à área de ciência animal. Apesar de simples, a técnica de RF é, ao mesmo tempo, flexível, robusta e escalável para problemas da ordem encontrada ao se analisar dados de GWAS.

3.6 Referências

AMIT, Y.; GEMAN, D. Shape quantization and recognition with randomized trees. *Neural Computation*, v. 9, p. 1545–1588, 1997.

BARRETT, J. C. et al. Haploview: analysis and visualization of ld and haplotype maps. *Bioinformatics*, v. 21, p. 263–265, 2005.

BOLDMAN, K. G. et al. *MTDFREML. A Set of Programs to Obtain Estimates of Variances and Covariances*. Washington. DC: U.S.: Department of Agriculture, Agricultural Research Service, 1995.

BOVINE GENOME SEQUENCING; ANALYSIS CONSORTIUM. The genome sequence of taurine cattle: a window to ruminant biology and evolution. *Science*, v. 342, n. 5926, p. 522–528, 2009.

BOVINE HAPMAP CONSORTIUM. Genome-wide survey of snp variation uncovers the genetic structure of cattle breeds. *Science*, v. 324, n. 5926, p. 528–532, 2009.

BREIMAN, L. Bagging predictors. *Machine Learning*, v. 24, p. 123–140, 2001.

BREIMAN, L. Random forests. *Machine Learning*, v. 45, p. 5–32, 2001.

BREIMAN, L.; CUTLER, A. *Random Forest*. 2013. Disponível em: <http://www.stat.berkeley.edu/~breiman/RandomForests/>. Acesso em: 15.9.2013.

BREIMAN, L. et al. *Classification and Regression Trees*. London: Chapman & Hall, 1984.

BUREAU, A. et al. Identifying snps predictive of phenotype using random forests. *Genetic Epidemiology*, v. 28, p. 171–182, 2005.

EIGELSON, H. S. et al. Genetic variation in candidate obesity genes adrb2, adrb3, ghrl, hsd11b1, irs1, irs2, and shc1 and risk for breast cancer in the cancer prevention study ii. *Breast Cancer Research*, v. 10, p. R57, 2008.

ELSEN, J.-M. et al. XVth QTLMAS: simulated dataset. *BMC Proceedings*, v. 6(Suppl 2), n. S1, 2012. Disponível em: <https://www.biomedcentral.com/1753-6561/6/S2/S1>.

FASTRF. *fast-random-forest: An efficient implmentation of the Random forest classifier for Java*. 2013. Disponível em: <https://code.google.com/p/fast-random-forest/>. Acesso em: 15.9.2013.

FREUND, Y.; SHAPIRE, R. Experiments with a new boosting algorithm. In: SAITTA, L. (Ed.). *Proceedings of the 13th International Conference of Machine Learning*. San Francisco: Morgan Kaufmann, 1996. p. 148–156.

GARRICK, D. J.; TAYLOR, J. F.; FERNANDO, R. L. Deregressing estimated breeding values and weighting information for genomic

regression analyses. *Genetics, Selection, Evolution*, p. 41–55, 2009.

GEER, L. Y. et al. The ncbi biosystems database. *Nucleic Acid Research*, v. 38, p. D492–D496, 2010.

GOLDSTEIN, B. A. et al. An application of random forests to a genome-wide association: Methodological considerations & new findings. *BMC Genetics*, v. 11, n. 49, 2010. Disponível em: <http://www.biomedcentral.com/1471-2156/11/49>.

GONZÁLEZ-RECIO, O.; FORNI, S. Genome-wide prediction of discrete traits using bayesian regressions and machine learning. *Genetics Selection Evolution*, v. 43, n. 7, 2011. Disponível em: <http://www.gsejournal.org/content/43/1/7>.

HERDENDORF, T. J.; MIZIORKO, H. M. Functional evaluation of conserved basic residues in human phosphomevalonate kinase. *Biochemistry*, v. 46, p. 11780–11788, 2007.

HO, T. K. The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Learning Intelligence*, v. 20, n. 8, p. 832–844, 1998.

KANEHISA, M.; GOTO, S. Kegg: kyoto encyclopedia of genes and genomes. *Nucleic Acid Research*, v. 28, p. 27–30, 2000.

KANEHISA, M. et al. Kegg for integration and interpretation of large-scale molecular data sets. *Nucleic Acid Research*, v. 40, p. D109–D114, 2012.

LIAW, A.; WIENER, M. Classification and regression by randomforest. *R News*, v. 2/3, p. 18–22, 2012.

MCDANIEL, A. H. et al. A locus on mouse chromosome 9 (adip5) affects the relative weight of the gonadal but not retroperitoneal adipose depot. *Mammalian Genome*, v. 17, p. 1078–1092, 2006.

MOKRY, F. B. et al. Genome-wide association study for backfat tickness in canchim beef cattle using random forest approach. *BMC Genetics*, v. 14, n. 47, 2013. Disponível em: <http://www.biomedcentral.com/1471-2156/14/47>.

MOORE, J. H.; ASSELBERGS, F. W.; WILLIAMS, S. M. Bionformatics challenges for genome-wide association studies. *Bioinformatics*, v. 26, n. 4, p. 445–455, 2010.

OGURO, R. et al. A single nucleotide polymorphism of the adenosine deaminase, rna-specific gene is associated with the serum triglyceride level, abdominal circumference, and serum adiponectin concentration. *Biochemistry*, v. 47, p. 183–187, 2012.

PARF. *parf: Parllel Random Forest Algorithm*. 2013. Disponível em: <http://code.google.com/p/parf/>. Acesso em: 15.9.2013.

QTL-MAS. *16th QTL-MAS Workshop*. 2012. Sítio do 16th QTL-MAS Workshop. Disponível em: <http://qtl-mas-2012.kassiopeagroup.com/en/index.php>. Acesso em: 15.9.2013.

R CORE TEAM. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2013. ISBN 3-900051-07-0. Disponível em: <http://www.R-project.org/>.

SAS INSTITUTE INC. *SAS/STAT Software. SAS Institute Inc: Version 9.3*. Cary N, 2011.

SCHEET, P.; STEPHENS, M. A fast and flexible statistical model for large-scale population genotype data: applications to inferring missing genotypes and haplotypic phase. *American Journal of Human Genetics*, v. 78, p. 629–644, 2006.

SCHWARZ, D. F.; KÖNIG, I. R.; ZIEGLER, A. On safari to random jungle: a fast implementation of

random forests for high-dimensional data, bioinformatics. *Bioinformatics*, v. 26, n. 14, p. 1752–1758, 2010. Disponível em: <http://bioinformatics.oxfordjournals.org/content/26/14/1752.long>.

SCHWARZ, D. F. et al. Picking single-nucleotide polymorphisms in forests. *BMC Proceedings*, v. 1(suppl. 1), n. S59, 2007. Disponível em: <http://www.biomedcentral.com/1753-6561/1/S1/S59>.

SCIKIT. *scikit-learn: Machine Learning in Python*. 2013. Disponível em: <http://scikit-learn.org/stable/>. Acesso em: 15.9.2013.

SHAWE-TAYLOR, J.; CRISTIANINI, N. *Kernel Methods for Pattern Analysis*. Cambridge: Cambridge University Press, 2004.

SZYMCZAK, S. et al. Machine learning in genome-wide association studies. *Genetics Epidemiology*, v. 33(supplement 1), p. S51–S57, 2009.

VIANNA, A. T. *Formação do gado Canchim pelo cruzamento charoles-zebu*. Rio de Janeiro: Ministério da Agricultura, 1960.

YAO, C. et al. Random forests approach for identifying additive and epistatic single nucleotide polymorphisms associated with residual feed intake in dairy cattle. *Journal of Diary Science*, v. 96, n. 10, p. 6716–6729, 2011.

YTOURNEL, F. et al. LDSO: a program to simulate pedigrees and molecular information under various evolutionary forces. *Journal of Animal Breeding and Genetics*, v. 129, p. 417–421, 2012.

ZHANG, H.; WANG, M.; CHEN, X. Willows: a memory efficient tree and forest contruction package. *BMC Bioinformatics*, v. 10, n. 130, 2009. Disponível em: <http://www.biomedcentral.com/1471-2105/10/130>.

ZIEGLER, A.; DESTEFANO, A. L.; KÖNIG, I. Data mining, neural nets, trees - probems 2 and 3 of genetic analysis workshop 15. *Genetics Epidemiology*, v. 31(supplement 1), p. S51–S60, 2007.

4

Metodologia para seleção de marcadores com máquina de vetores de suporte com regressão

Fabrízzio Condé de Oliveira
Fernanda Nascimento Almeida
Fabyano Fonseca e Silva
Marcos Vinícius Gualberto Barbosa da Silva
Carlos Cristiano Hasenclever Borges
Wagner Arbex

ESTE TRABALHO propõe uma nova metodologia para selecionar simultaneamente os marcadores SNPs mais relevantes para a caracterização de qualquer fenótipo mensurável descrito por uma variável contínua, usando SVR com o Pearson Universal Kernel. A metodologia proposta é multiatributo no sentido de considerar vários marcadores simultâneos para a explicação do fenótipo e baseia-se conjuntamente em um ferramental estatístico, técnicas de aprendizado de máquina e de inteligência computacional. Atualmente, a maioria dos estudos de associação em escala genômica, chamados de GWAS (Genome-wide Association Studies), quantificam o impacto médio de cada marcador sobre o fenótipo por meio de regressões lineares simples entre um marcador e o fenótipo (modelos monoatributos), com o intuito de indicar os marcadores mais

significativos em relação à característica fenotípica em questão. Porém, tais métodos pressupõem que os efeitos de cada marcador sobre o fenótipo são somente aditivos, desconsiderando a possível ocorrência de interações complexas como epistasia e dominância entre os marcadores.

4.1 Introdução

Polimorfismos de base única (*single nucleotide polymorphisms* – SNPs) são uma forma abundante de variação genômica, que, segundo Brookes (1999), se diferem das variações raras. Além disso, o pressuposto básico para estudos de associação ampla, isto é, GWAS, é que a característica avaliada pode ser explicada a partir desse tipo de marcador. Assim, considera-se que existam SNPs no genótipo com alto desequilíbrio de ligação (*linkage disequilibrium* – LD) em relação ao Quantitative Trait Locus (QTL).

Desta forma, a abordagem tradicional é avaliar quais os marcadores que possuem alta associação com o fenótipo por meio do *valor-p* do beta da regressão linear simples entre cada SNP e o fenótipo. Após esse passo, é verificado se os SNPs mais relevantes estão próximos àlguma região no genoma que está associada àquela característica.

Até o momento, a previsão do risco de doenças em humanos baseada em SNPs validados com base nessa metodologia mostrou pouco poder preditivo (MITTAG et al., 2012), apesar de tais SNPs indicarem alta significância de associação com a característica fenotípica. Esse fato pode ser explicado devido a variância dos marcadores mais significativos possuir baixo poder explicativo em relação à variância fenotípica, conforme Moore e Williams (2010). Desta forma, uma abordagem alternativa é ampliar o número de marcadores, considerando também os que possuem pequenas

correlações sobre a característica avaliada. Mas, tal fato cria dois problemas: o número de marcadores é elevado e muitos deles são correlacionados entre si.

De acordo com Gianola, Perez-Enciso e Toro (2003), tal análise demanda a utilização de métodos estatísticos que considerem a seleção de covariáveis (problema de multicolinearidade) e a regularização do processo de estimação (problema de dimensionalidade). Outras técnicas de regressão foram criadas para abordar esse problema como regressão *ridge* e regressão por mínimos quadrados parciais (MORSER; HAYES; RAADSMA, 2010).

Por outro lado, algoritmos de aprendizado de máquina, tal como, máquina de vetores de suporte ou Support Vector Machine (SVM) em GWAS, considerando múltiplos marcadores em problemas de classificação, vêm demonstrando desempenho satisfatório como em Mittag et al. (2012), Wei et al. (2009) e Ban et al. (2010).

O presente estudo visa propor um método que consiga avaliar simultaneamente vários SNPs em relação ao fenótipo descrito por uma variável contínua, diferentemente de fenótipos dicotômicos caso-controle abordados na maioria de estudos de GWAS. Com isso, têm-se dois benefícios imediatos em relação à metodologia mais comumente usada: um relativo aos diversos níveis de ocorrência do fenótipo e o outro, pelas interações complexas simultâneas que podem ocorrer entre os diversos marcadores.

Com a evolução de novos *chips* para bovinos, com densidades de 500.000 a 800.000 marcadores e se a estrutura genética subjacente é definida por um grande número de pequenas QTLs, a formação de grandes conjuntos de dados para treinamento e medidas fenotípicas precisas serão necessárias para realizar a melhoria da acurácia no aumento da densidade de SNPs (HARRIS; JOHNSON, 2010). Consequentemente, é de suma importância que novas metodologias sejam desenvolvidas para tratar adequadamente dados genômicos com alta dimensionalidade sem a eliminação de

variáveis relevantes. Portanto, após identificação do subconjunto de marcadores suficientes e necessários para a explicação do fenótipo, é possível a redução de custos na confecção de *chips* personalizados com menor número de SNPs para a predição do fenótipo a partir de métodos em seleção genômica.

4.2 Material

Para demonstrar a metodologia proposta, foi utilizado um conjunto de dados de 343 touros genotipados da raça Gir fornecido pela Empresa Brasileira de Pesquisa Agropecuária (Embrapa), sendo que somente 244 animais possuem prole fêmea, permitindo a mensuração do fenótipo avaliado.

4.2.1 Fenótipo

O potencial genético do leite de um animal é computado a partir da produção de leite de sua prole fêmea com base na metodologia utilizada em Verneque et al. (2012). A PTA é a capacidade prevista de transmissão, sendo uma medida do desempenho esperado das filhas do touro em relação à média genética dos rebanhos (VERNEQUE et al., 2012). Assim, por exemplo, uma PTA de 500 kg para produção de leite significa que, se o touro for usado numa população com nível genético igual ao usado para avaliá-lo, cada filha produzirá em média 500 kg por lactação a mais do que a média do rebanho (VERNEQUE et al., 2012). Considerando-se dois touros, um com PTA de 500 kg e outro com –100 kg, espera-se que, em acasalamentos ao acaso, as filhas do primeiro touro produzam em média 600 kg a mais do que as filhas do segundo touro. A metodologia aplicada para a medição da PTA e os resultados obtidos estão descritos detalhadamente em Verneque et al. (2012).

A principal diferença em relação aos estudos apresentados em Mittag et al. (2012) e Wei et al. (2009), sobre a seleção de marcadores, é na troca da classificação, com uso de SVM, pela regressão, com o uso de máquinas de vetores de suporte com regressão, ou Support Vector Regression (SVR)[1]. Isso permite diferenciar vários níveis da característica fenotípica, uma vez que a PTA para leite é descrito por uma variável contínua, diferentemente de problemas de classificação caso-controle.

Para o cálculo da PTA para leite, somente o efeito genético é considerado, eliminando-se todos os outros efeitos ambientais. Assim, é coerente a explicação da PTA a partir de informações dos marcadores moleculares. De acordo com a Tabela 4.1, foi possível notar a grande amplitude de valores da PTA, indicando a necessidade de modelos robustos para o mapeamento dessa medida por meio do genótipo.

Tabela 4.1 – Estatísticas da PTA para leite.

Mínimo	1 quartil	Mediana	Média	3 quartil	Máximo
-479,5	328,0	583,2	641,3	908,3	1978,0

De acordo com o histograma da Figura 4.1, fica evidente que a distribuição da PTA para leite tem assimetria positiva.

A partir do *boxplot* da Figura 4.2, nota-se que há 2 pontos aberrantes na distribuição da PTA. Portanto, para verificar a normalidade do fenótipo, foi aplicado o teste de normalidade Shapiro-Wilk, que mostrou *valor-p* igual a 0,01991, indicando que não há evidências de que a distribuição do potencial genético têm uma distribuição normal a um nível de significância adotado de 0,05. É importante ressaltar que tal fato ocorreu devido ao reduzido tamanho da amos-

[1] A primeira versão para SVR foi proposta em 1996 por Druker et al. (1997)

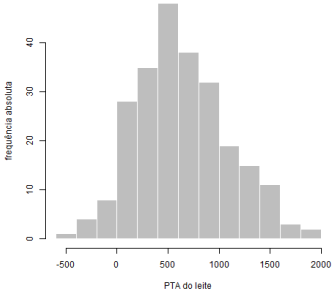


Figura 4.1 – Histograma da PTA para leite.

tra, pois a PTA da população de touros Gir segue uma distribuição normal.

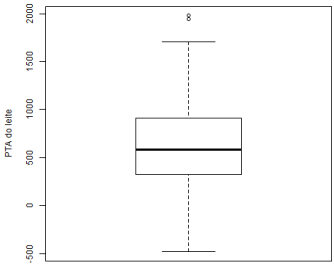


Figura 4.2 – *Boxplot* da PTA para leite.

4.2.2 Genótipo

O genoma bovino tem aproximadamente 3 bilhões de pares de bases. Além disso, ele possui 30 pares de cromossomos, sendo 29 pares autossômicos e 1 par sexual. O genótipo foi obtido com o uso do *chip* de marcadores SNP Illumina 56K, perfazendo

um total de 56.947 marcadores. Logo, as variáveis explicativas, descritas pela frequência do alelo B no *locus*, foram codificadas da seguinte maneira: AA = 0, ausência do alelo B; AB = 1, presença de uma cópia do alelo B; e BB = 2, presença de duas cópias do alelo B. Os valores faltantes, devido a problemas de leitura, foram considerados como heterozigoto AB = 1.

4.2.3 Pré-processamento

Para a comparação dos filtros utilizados para selecionar os marcadores mais importantes foram criadas duas bases de dados: uma sem e outra com controle de qualidade (CQ). Não foi feito nenhum pré-processamento ou filtragem padrão nos dados sem CQ, tais como, *call rate*, *minor frequency allelic* (MAF) e Equilíbrio de Hardy-Weinberg (EHW). O objetivo disso foi não eliminar SNPs com pequenos efeitos isolados, que quando combinados com outros SNPs, sejam importantes na descrição da PTA para leite.

Já para a base com CQ, foram aplicados os filtros *call rate* $\geq$ 0,95; MAF $\geq$ 0,05 e EHW $\geq \frac{0,05}{56.947}$, sendo esse valor a significância de 0,05 ajustada pela correção de Bonferroni e 56.947 é a quantidade de SNPs na base original. Após a aplicação dos filtros descritos acima, restaram 22.799 marcadores para a aplicação do método de seleção criado nesse estudo.

Na amostra sem CQ, 6.192 marcadores não apresentaram informação devido a erro ou inconsistência na leitura do *chip* Ilummina Bovine 56k, restando 50.755 marcadores. Desse restante, 3 marcadores não possuíam variação alélica, logo, foram desconsiderados, totalizando 50.752 para análise subsequente.

4.3 Método

Os passos do método proposto são:

1. calcular o *valor-p* da correlação de Spearman de cada marcador com o fenótipo;

2. agrupar os marcadores através do *valor-p* para construir uma sequência crescente de subconjuntos de marcadores;

3. computar a correlação dos valores preditos pelo SVR e o genótipo observado para os grupos construídos na etapa 2 com Pearson Universal Kernel (PUK), cujos parâmetros são otimizados, e escolher o grupo com o melhor desempenho, ou seja, realiza-se a primeira seleção de marcadores;

4. a partir do melhor grupo calculado na etapa anterior, usa-se o algoritmo genético (AG) para realizar uma segunda seleção de marcadores.

O objetivo da etapa 1 é atribuir um "peso" a todos os marcadores, sendo os mais relevantes, inicialmente, os que apresentam correlações mais significativas sobre o fenótipo.

O passo 2 destina-se a separar os marcadores em grupos cada vez maiores, permitindo a entrada de marcadores menos significativos na construção dos modelos regressores.

O passo 3 tem por objetivo avaliar o desempenho do grupo de SNPs em relação ao PUK a fim de capturar as possíveis interações de SNPs que não foram mapeadas no passo 1. Assim, quando dois SNPs com efeitos principais não significativos são analisados em conjunto, podem gerar efeitos de interação significativos maiores que quando analisados separadamente.

O passo 4 objetiva reduzir o número de marcadores redundantes que estão altamente correlacionados com os principais marcadores associados com QTLs. Isso ocorre porque na etapa anterior o filtro usado não avalia esse ponto. Além disso, o espaço de busca de todas as combinações possíveis de marcadores é 2^n,

em que n é o total de marcadores, sendo este valor extremamente proibitivo na prática.

Portanto, o AG irá selecionar uma amostra de diferentes combinações de marcadores para evoluir em direção ao "melhor" subconjunto de marcadores através de operadores genéticos recombinação e mutação.

Para a melhor compreensão do passo 4, devem ser definidos os AGs e seus operados genéticos, além de melhor entendido o funcionamento básico dos mesmos.

AGs são algoritmos de busca baseados em processos genéticos e seleção natural (GOLDBERG, 1989), isto é, são algoritmos que simulam a evolução "do que se está buscando" por meio de três operadores, denominados "operadores genéticos", que apresentam seus funcionamentos baseados em processos genéticos, são eles: "seleção", "recombinação"[2] (*crossover*) e "mutação".

O procedimento geral de execução de um AG inicia-se com o estabelecimento de uma população inicial, isto é, um conjunto de dados, gerada com um determinado tamanho e, em seguida, é realizada uma seleção dos indivíduos que participarão no processo de reprodução. Após a seleção, é feita a recombinação e, sucessivamente, a reprodução é realizada. A seguir, o operador mutação é aplicado em alguns dos descendentes gerados a partir da recombinação, criando finalmente a nova população.

Esse procedimento é repetido até que um critério de parada seja satisfeito, podendo ser o número máximo de gerações ou um erro máximo estipulado, ambos definidos pelo usuário. Cabe ressaltar que existem diversas técnicas de codificação de cromos-

[2] Segundo Linden (2008), é o processo no qual um "pedaço" de cada cromossomo, um do pai e o outro da mãe, é trocado com seu par. Tal processo permite que a progênie herde características de seus pais, mas com algum nível de diferença em relação a estes.

somos[3] e de seleção de indivíduos, além de variados operadores de recombinação e de mutação.

Os operadores de recombinação e mutação são baseados em probabilidades, logo, nem todos os indivíduos da população cruzarão e/ou sofrerão mutação. Assim, caso as sementes aleatórias adotadas para o AG sejam distintas, possivelmente, as soluções finais também serão diferentes.

AGs podem ser usados para seleção de variáveis, tanto em problemas de classificação como em problemas de regressão. Sua grande vantagem é executar uma pesquisa inteligente no sentido de "caminhar" na busca de bons candidatos, garantindo uma melhoria constante – desde que se adote o "elitismo"[4], mas mantendo alguma variabilidade para aumentar a possibilidade de encontrar o melhor candidato.

No caso da seleção de variáveis, a codificação de cada indivíduo deve ser binária, ou seja, o cromossomo conterá somente zeros, representando as variáveis não selecionadas, e/ou uns, que representam as variáveis selecionadas. Nesse caso, a função de aptidão[5] é alguma medida de erro adequada a ser minimizada.

---

[3] De acordo com Linden (2008), na área de estudo de AGs, os termos indivíduo e cromossomo são sinônimos. Com isso, cromossomo é uma representação de parte do espaço de busca do problema de otimização original a ser resolvido. Por exemplo, para otimizações em problemas cujos valores de entrada são apenas inteiros positivos de valor menor que 255 pode-se usar uma codificação binária representada em um vetor de 8 *bits* para representar um cromossomo do AG. Além disso, cada posição do cromossomo é denominado gene, de forma análoga com as partes fundamentais que compõem um cromossomo biológico. Ressalta-se que existem vários outros tipos de representação cromossional distintas da codificação binária com objetivo de resolver outras classes de problemas, tais como, a representação com números inteiros ou com números reais.

[4] Linden (2008) descreve elitismo como o procedimento onde os n melhores indivíduos de cada geração não devem "morrer" junto com a sua geração, mas sim passar para a próxima, visando que seus genomas sejam conservados.

[5] Função de aptidão é a saída da função objetivo do problema a ser otimizado pelo AG. A função objetivo pode ser qualquer função matemática de um problema de otimização. Desta forma, os indivíduos, representados pelos cromossomos, são avaliados pela função de aptidão para posterior ordenamento dos mesmos no intuito de escolher os "melhores". Uma das principais vantagens dos AGs em

4.3.1 Modelo

Na proposta apresentada e discutida nesse texto, utiliza-se SVR para avaliar o poder explicativo dos grupos de marcadores em relação à PTA para leite.

Essa escolha foi devido à sua grande flexibilidade, pois o SVR não pressupõe linearidade do modelo – desde que seja adotado um *kernel* não-linear –, nem normalidade dos resíduos e adapta-se facilmente a dados de alta dimensionalidade.

4.3.2 Máquina de vetores de suporte com regressão (SVR)

Seja o conjunto $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ com $x_i \in \Re^d$. O objetivo do SVR é encontrar o funcional linear f, descrito pela Expressão 4.1, que mapeia as variáveis do espaço de entrada de $\Re^d$ na variável do espaço de saída $\Re$ minimizando a Expressão 4.2.

$$f(x) = \langle w, x \rangle + b \tag{4.1}$$

Com $w, x \in \Re^d$, onde w e b significam, respectivamente, a inclinação e o intercepto do hiperplano a serem estimados a partir da Expressão 4.2.

$$Min \frac{1}{2}\|w\|^2 + C \sum_{i=1}^{n} L_\varepsilon(f(x_i), y_i) \tag{4.2}$$

onde,

$$L_\varepsilon(f(x_i), y_i) = \begin{cases} 0 & \text{se } |y_i - f(x_i)| \leq \varepsilon; \\ |y_i - f(x_i)| - \varepsilon & \text{se } |y_i - f(x_i)| > \varepsilon. \end{cases}$$

relação à função objetivo, é que não é necessário conhecer o funcionamento da mesma, mas somente tê-la disponível para aplicação nos indivíduos, objetivando compará-los. A partir disso, é possível usar um modelo "caixa-preta" onde somente é conhecido o formato das entradas e o valor de saída da função objetivo, o qual será otimizado pelo AG.

De acordo com a Expressão 4.2, o termo $\frac{1}{2}\|w\|^2$ indica a complexidade do modelo e o termo $L_\varepsilon(f(x_i), y_i)$ traduz a função de perda ε-insensível que não penaliza os valores dentro do "tubo", ou seja, com erros menores que ε. O parâmetro *C* é chamado de constante de regularização e traduz o equilíbrio entre a complexidade de *f* e a quantidade de desvios maiores que ε serão tolerados, como pode ser visto em Ünstün, Melssen e Buydens (2006). Assim, quanto menor o tubo (menor ε), mais complexa é a função *f* e, de forma contrária, quanto maior o tubo (maior ε), menos complexidade é necessária para *f*. Os parâmetros *C* e ε precisam ser otimizados para encontrar um modelo adequado aos dados como mostra Ünstün, Melssen e Buydens (2006).

De acordo com Ünstün, Melssen e Buydens (2006), com a introdução de variáveis de folga ξ_i e ξ_i^* e devidas manipulações algébricas, a Expressão 4.2 se transforma na função objetivo da Expressão 4.3. Tal formulação é chamada de primal, pois a regressão é baseada no espaço original dos dados. As variáveis de folga têm por objetivo possibilitar a ocorrência de vetores fora do tubo, sendo os mesmos chamados vetores de suporte, pois são somente eles que contribuem para a regressão. Desta forma, todos os outros vetores dentro do tubo podem ser removidos após a construção do modelo. Essa propriedade permite que o SVR modele relações em que o número de variáveis dependentes seja muito maior que o tamanho da amostra.

$$Min\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \tag{4.3}$$

Sujeita às restrições:

$$\begin{cases} y_i - \langle w, x\rangle - b \leq \varepsilon + \xi_i \\ \langle w, x\rangle + b - y_i \leq \varepsilon + \xi_i^* \end{cases}$$

No caso do espaço dos dados originais não possuir relação linear com a variável dependente, a função *f* do modelo primal é reformulada para o modelo dual como mostra a Expressão 4.4. Com isso, o espaço original é mapeado para um novo espaço, denominado de espaço de características, por meio da função ϕ e do produto interno $K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$, sendo K chamado de função *kernel*. Esta função traduz a relação subjacente entre os dados de entrada e de saída.

$$f(x) = \left[\sum_{i=1}^{n} (\alpha_i - \alpha_i^*) \langle \phi(x_i), \phi(x_j) \rangle \right] + b \qquad (4.4)$$

As variáveis duais α_i e α_i^* representam os multiplicadores de Lagrange que satisfazem as desigualdades $0 \leq \alpha_i, \alpha_i^* \leq C$ e que podem ser obtidos pelo problema de maximização cuja função objetivo dada pela expressão

$$\mathrm{Max}\left\{ -\frac{1}{2} \left[\sum_{i,j=1}^{n} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) K(x_i, x_j) \right] - \varepsilon \left[\sum_{i=1}^{n} (\alpha_i + \alpha_i^*) \right] + \left[\sum_{i=1}^{n} y_i (\alpha_i - \alpha_i^*) \right] \right\}$$

que está sujeita às restrições

$$\begin{cases} \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) & = & 0 \\ \alpha_i, \alpha_i^* & \geq & 0 \end{cases}$$

A principal vantagem da utilização da função *kernel* é o mapeamento linear entre os dados de entrada transformados pela função *kernel* e os dados de saída. Isso é possível porque o espaço de características tem dimensão superior à dimensão do espaço original. Com isso, a linearidade da regressão é obtida no espaço de características e não no espaço original. Assim, o *kernel* linear é dado pela Expressão 4.5, sendo *C* e ε os únicos parâmetros desse *kernel*.

$$K(x_i, x_j) = \langle x_i, x_j \rangle \qquad (4.5)$$

O *kernel* função de base radial (RBF) é um *kernel* de propósito geral quando não se tem conhecimento a *priori* sobre os dados (KARATZOGLOU; SMOLA; HORNIK, 2004). Esse *kernel* é dado pela Expressão 4.6 possui 3 parâmetros *C*, ε e γ que devem ser escolhidos adequadamente.

$$K(x_i, x_j) = exp(\gamma \|x_i - x_j\|^2) \tag{4.6}$$

O PUK, dado pela Expressão 4.7, foi adotado como o *kernel* para a metodologia proposta neste artigo. Como mostrado em Ünstün, Melssen e Buydens (2006), a função Pearson VII tem a possibilidade de mudar facilmente, adaptando seus dois parâmetros σ e ω, entre as formas da função Gaussiana à Lorentziana e até mesmo a outras funções.

$$K(x_i, x_j) = \frac{1}{\left[1 + \frac{2\sqrt{\|x_i - x_j\|^2}\sqrt{2^{1/\omega} - 1}}{\sigma}\right]} \tag{4.7}$$

Assim, esse *kernel* tem robustez ao mostrar que variações percentuais significativas nos parâmetros provocam variações percentuais inferiores na RMSE[6] (Root Mean Square Error) das predições como pode ser notado em Ünstün, Melssen e Buydens (2006). Logo, o PUK pode substituir os *kernel* comumente aplicados (linear, polinomial e RBF), podendo apresentar resultados iguais ou superiores no que tange ao desempenho da generalização do SVR.

A principal vantagem dessa metodologia é substituir a escolha sobre os *kernel* padrões pela escolha dos melhores parâ-

6

$$\mathrm{RMSE} = \sqrt{\frac{1}{\mathrm{n}}\Big[\sum_{\mathrm{i}=1}^{\mathrm{n}}(\hat{\mathrm{y}}_\mathrm{i} - \mathrm{y}_\mathrm{i})^2\Big]}$$

onde n é o tamanho da amostra,$\hat{y_i}$ e y_i são, respectivamente, os valores predito e observado do indivíduo i.

metros do PUK. Essa troca de escolha gera claramente um ganho no tempo de processamento, pois para cada *kernel* adotado é necessário otimizar seus parâmetros e, caso se adote o PUK, o *kernel* não é trocado por outro, pois o mesmo mimetiza o comportamento de outros *kernel*.

4.3.3 Primeira seleção dos marcadores

Para a construção dos grupos de marcadores mais significativos, utilizou-se o coeficiente de correlação de Spearman, pois o mesmo apresenta algumas vantagens em relação ao coeficiente de correlação de Pearson, a saber: não pressupõe que a relação entre as duas variáveis seja linear, não altera significativamente seu resultado na presença de dados aberrantes e não necessita que as variáveis tenham a *priori* determinadas distribuições de probabilidade conforme Field (2005). Resumidamente, o coeficiente de Spearman é o coeficiente Pearson aplicado aos postos das variáveis originais. A Expressão 4.8 mostra como se calcula o coeficiente de correlação de Spearman (ρ) para dados repetidos com uma amostra de tamanho n. As variáveis com seus valores brutos X_i , Y_i para $i = 1, 2, ..., n$ são transformadas para seus postos, designados por x_i e y_i.

$$\rho = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}} \tag{4.8}$$

Para a ordenação dos efeitos dos marcadores, foi avaliado o coeficiente de correlação de Spearman de cada marcador com o PTA e o seu respectivo *valor-p*. Logo, criou-se uma sequência crescente de subconjuntos de marcadores $A_1 \subset A_2 \subset \ldots \subset A_n$ sendo o de índice 1 referente ao *valor-p* 10^{-9}, o índice 2, ao *valor-p* 10^{-8} e, assim sucessivamente, até o de índice 8 que é referente ao maior *valor-p* 10^{-2}, como pode ser observado na Tabela 4.2.

Tabela 4.2 – Número de marcadores a partir do *valor-p* do coeficiente de correlação de Spearman.

Grupos	Intervalo de *valor-p*	# SNPs sem CQ	#SNPs com CQ
1	$< 10^{-9}$	68	12
2	$< 10^{-8}$	226	17
3	$< 10^{-7}$	431	43
4	$< 10^{-6}$	712	105
5	$< 10^{-5}$	1.181	242
6	$< 10^{-4}$	1.996	595
7	$< 10^{-3}$	3.440	1.397
8	$< 10^{-2}$	6.512	3.356

A título de verificação da similaridade dos resultados encontrados para os *kernel* avaliados por Ünstün, Melssen e Buydens (2006), também foram computados os *kernel* linear e RBF. A intenção foi verificar que, com os parâmetros adequados, o PUK tenha desempenho igual ou superior aos outros *kernel*.

4.3.4 Parâmetros

Para a escolha dos parâmetros C e ε dos modelos SVR com *kernel* linear foram feitas análises com diversos valores para ambos. Em relação aos outros dois *kernels* (RBF e PUK), além dos dois parâmetros anteriores, foram testados o parâmetro específico γ para o *kernel* RBF e os parâmetros específicos ω e σ para o PUK. Cabe ressaltar que, para cada subconjunto avaliado de SNPs, foram realizadas várias simulações para os parâmetros supracitados dos modelos SVR, objetivando encontrar o grupo mais adequado para cada modelo.

4.3.5 Comparação dos modelos

Para a comparação dos modelos SVR, foi utilizada a validação cruzada com 10 partições em cada um dos 8 conjuntos

de dados da Tabela 4.3, sendo esse procedimento repetido 10 vezes com semente aleatória distinta para cada partição criada, perfazendo um total de 10 estimativas para o coeficiente de correlação de Pearson. Desta forma, aumenta-se a confiabilidade na comparação estatística dos modelos.

Tabela 4.3 – Média e desvio padrão do coeficiente de correlação de Pearson em 10 partições com 10 repetições na validação cruzada dos 3 modelos de SVR sem CQ.

Grupos	Intervalo de *valor-p*	# SNPs	*Kernel* linear	*Kernel* RBF	PUK
1	$< 10^{-9}$	68	0,60 (0,14)	0,68 (0,11)	0,68 (0,11)
2	$< 10^{-8}$	226	0,48 (0,17)	0,72 (0,09)	0,72 (0,09)
3	$< 10^{-7}$	431	0,44 (0,16)	0,74 (0,09)	0,75 (0,08)
4	$< 10^{-6}$	712	0,71 (0,09)	0,77 (0,08)	0,74 (0,09)
5	$< 10^{-5}$	1.181	0,76 (0,09)	0,76 (0,08)	0,78 (0,08)
6	$< 10^{-4}$	1.996	0,78 (0,08)	0,74 (0,08)	0,78 (0,08)
7	$< 10^{-3}$	3.440	0,80 (0,08)	0,67 (0,13)	0,80 (0,08)
8	$< 10^{-2}$	6.512	**0,81 (0,08)**	**0,81 (0,08)**	**0,81 (0,08)**

4.3.6 Segunda seleção de marcadores

Com base no grupo 8 que gerou a maior média e o menor desvio padrão da correlação, que pode ser visto nas Tabelas 4.3 e 4.4, um *wrapper*[7], baseado em um AG binário com função de aptidão (*fitness*) dado pela validação cruzada do Mean Square Error (MSE)[8], é aplicado para uma segunda seleção de marcadores.

[7] Segundo Facelli et al. (2011), *wrapper* é uma técnica de seleção de atributos que utiliza o próprio algoritmo de aprendizado como uma caixa-preta para a seleção e, assim, para cada possível subconjunto, o algoritmo é consultado e o subconjunto que apresentar a melhor combinação entre redução da taxa de erro e redução do número de atributos é em geral selecionado.

[8]
$$\text{MSE} = \frac{1}{n}\Big[\sum_{i=1}^{n}(\hat{y}_i - y_i)^2\Big]$$

A descrição dos elementos da expressão acima pode ser encontrada na Seção 4.3.2, nota de rodapé 6, na página 114.

A metodologia utilizada nessa segunda seleção de variáveis é baseada em Goldberg (1989) e em Kohavi (1997).

Como o número de combinações entre os 6.512 marcadores é extremamente elevado, totalizando $2^{6.512}$ combinações, utilizou-se um AG para encontrar os "melhores" marcadores em um tempo computacional hábil, mesmo não garantindo a unicidade da solução "ótima" encontrada. Com isso, o objetivo do AG é verificar a possibilidade de eliminar alguns marcadores do melhor conjunto gerado a partir da primeira seleção, pois acredita-se que o AG consiga avaliar melhor as interações entre os marcadores do que a eliminação feita por meio de filtros de controle de qualidade padrões tais como *call rate*, HWE, MAF e LD.

Os parâmetros adotados para o AG usado para a seleção de SNPs foram as probabilidades de *crossover* e de mutação iguais a 0,60 e 0,033 respectivamente, o tamanho da população e o número de gerações iguais a 20. Tanto o *wrapper* quanto todos os modelos de SVR foram executados no software Weka, versão 3.7.9, como em Hall et al. (2009). Toda a manipulação e codificação dos dados foram feitas no software R (R CORE TEAM, 2013).

4.4 Resultados e discussão

A primeira análise foi referente à acurácia dos modelos SVR com *kernel* linear, RBF e PUK. A partir da Tabela 4.3, nota-se que o melhor modelo com PUK, tanto pela menor média quanto pelo menor desvio padrão do coeficiente de correlação foi o conjunto com *valor-p* menor que 10^{-2} (grupo 8).

De acordo com a Tabela 4.4, três modelos de SVR, baseados nos marcadores do grupo 8 da base com CQ, demonstraram predição e acurácia equivalentes, indicando uma correlação média de 0,80 com desvio padrão igual a 0,08. Além disso, parece que os marcadores do grupo 8 possuem uma associação linear com a PTA

para leite, pois tanto o *kernel* RBF quanto o PUK replicaram esse comportamento.

Tabela 4.4 – Média e desvio padrão do coeficiente de correlação de Pearson em 10 partições com 10 repetições na validação cruzada dos 3 modelos de SVR com CQ.

Grupos	Intervalo de *valor-p*	# SNPs	*Kernel* linear	*Kernel* RBF	PUK
1	$< 10^{-9}$	12	0,67 (0,10)	0,67 (0,10)	0,67 (0,10)
2	$< 10^{-8}$	176	0,64 (0,10)	0,67 (0,10)	0,67 (0,10)
3	$< 10^{-7}$	431	0,59 (0,12)	0,68 (0,09)	0,70 (0,08)
4	$< 10^{-6}$	105	0,31 (0,18)	0,72 (0,07)	0,71 (0,07)
5	$< 10^{-5}$	242	0,67 (0,09)	0,77 (0,08)	0,78 (0,07)
6	$< 10^{-4}$	595	0,77 (0,08)	0,69 (0,09)	0,79 (0,07)
7	$< 10^{-3}$	1.397	0,78 (0,08)	0,79 (0,09)	0,79 (0,08)
8	$< 10^{-2}$	3.357	**0,80 (0,08)**	0,75 (0,08)	**0,80 (0,08)**

A Tabela 4.5 mostra que o subconjunto de marcadores extraído do grupo 8 apresentou maior correlação média igual a 0,84 com desvio padrão ligeiramente inferior a 0,07 para o PUK. Isto demonstra um ganho significativo no uso do AG para a seleção dos SNPs mais informativos sem controle de qualidade. Entretanto, na base de dados com CQ, houve um pequeno aumento na média da correlação e foi mantido o mesmo desvio padrão (0,08).

Tabela 4.5 – Média e desvio padrão do coeficiente de correlação de Pearson em 10 partições com 10 repetições no melhor subconjunto encontrado pelo AG com os mesmos parâmetros usados no grupo 8.

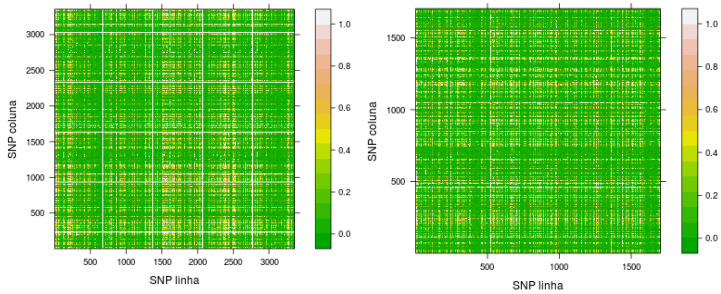
Wrapper	*Kernel* linear	*Kernel* RBF	PUK
AG sem CQ	0,84 (0,07)	0,67 (0,14)	0,84 (0,07)
AG com CQ	0,82 (0,08)	0,44 (0,25)	0,81 (0,08)

Em relação à base com CQ, o AG também eliminou vários marcadores altamente correlacionados conforme as Figuras 4.3a e 4.3b. Porém, o conjunto final de marcadores da Figura 4.3b ainda

indica algum grau de correlação. Na Figura 4.3d é possível verificar que o AG conseguiu eliminar vários marcadores redundantes (alto LD) do grupo 8 da base sem CQ, pois as linhas brancas da Figura 4.3c foram totalmente eliminadas. Além disso, a região de cor amarela (correlação entre 0,4 e 0,6) da Figura 4.3a foi diminuída em relação à Figura 4.3b. A partir dessas observações, parece que os filtros aplicados no grupo 8 das duas bases avaliadas eliminaram vários marcadores correlacionados com os demais.

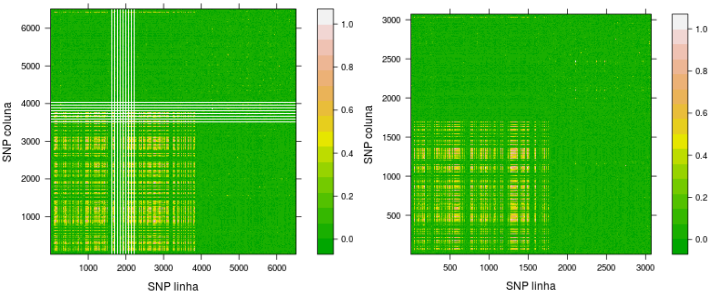
Em relação aos trabalhos de Wei et al. (2009) e Mittag et al. (2012), há um ganho evidente no uso do PUK em relação aos outros *kernel* analisados nos mesmos, pois Wei et al. (2009) utiliza os *kernel* linear e RBF com os valores padrões do pacote e1071 do software R, ou seja, os parâmetros do SVM não foram otimizados. No caso do trabalho de Mittag et al. (2012), somente o *kernel* RBF foi estudado e, para encontrar os "melhores" parâmetros C e γ, realizou-se uma extensa busca em grade. Porém, nenhum dos dois estudos extrai dos grupos de SNPs, construídos a partir do *valor-p*, o "melhor" subconjunto de marcadores para explicação do fenótipo e isso foi realizado pelo método proposto, usando o AG na segunda seleção.

O PUK demonstrou ser robusto para capturar o comportamento dos *kernel* linear e RBF,, desde que sejam usados os parâmetros adequados. Desse modo, na metodologia proposta neste trabalho, basta a avaliação do desempenho do SVR com PUK. Entretanto, independentemente do *kernel* adotado, a formulação matemática do SVR traz consigo uma desvantagem quanto à interpretação biológica, pois não é possível avaliar diretamente do hiperplano ótimo e dos vetores de suportes quais são os efeitos isolados de cada marcador, quais são os marcadores que atuam simultaneamente e o impacto global dos mesmos sobre o fenótipo. Diferentemente do que ocorre com os modelos de regressão linear múltipla utilizados em GWAS, os quais, a partir dos coeficientes



(a) Grupo 8 com CQ.

(b) Marcadores do grupo 8 com CQ.



(c) Grupo 8 sem CQ.

(d) Marcadores do grupo 8 sem CQ.

Figura 4.3 – LD calculado por r^2 entre os marcadores do grupo 8 (a) com CQ e (c) sem CQ, e os subconjuntos de marcadores extraídos do grupo 8 pelo AG (b) com CQ e (d) sem CQ.

estimados de cada marcador com base na amostra de indivíduos, indicam quais SNPs são mais relevantes para a característica fenotípica avaliada.

O método desenvolvido neste estudo indicou que os 68 marcadores mais significativos do grupo 1 sem CQ tem baixo poder preditivo e baixa acurácia em relação à PTA para leite e que o subgrupo com 3.073 marcadores demonstrou elevada predição e maior acurácia. Isso pode indicar que a PTA para leite é um fenótipo que é influenciado por diversos marcadores com pequenos efeitos

sobre o mesmo, além da possibilidade de ocorrência de epistasia e dominância, porém, tais fenômenos genéticos não podem ser comprovados pelo método sugerido nesse trabalho.

Os modelos SVR, utilizando o PUK, dos grupos 8, com e sem CQ, demonstraram elevado poder preditivo mesmo na presença de não-normalidade na variável dependente PTA para leite, além de terem desempenhos e acurácias semelhantes. Porém, quando foi aplicado o filtro do AG, o melhor subconjunto gerado a partir do grupo 8 sem CQ foi superior tanto em predição quanto em acurácia em relação ao grupo 8 com CQ. Esse fato parece mostrar que o grupo 8 sem CQ possui marcadores suficientes e necessários para a explicação do fenótipo e o grupo 8 com CQ possui marcadores necessários, mas não suficientes.

4.5 Conclusões e trabalhos futuros

O método desenvolvido no presente trabalho demonstrou robustez, pois o conjunto inicial dos marcadores sem CQ foi composto de aproximadamente 50.752 marcadores e chegou-se a 3.073 ao final do processo de seleção, garantindo boa precisão e alta acurácia para o modelo SVR com o PUK. Além desse fato, o AG conseguiu eliminar a maior parte da redundância na base sem CQ e em escala menor na base com CQ. Porém, a questão remanescente é tentar entender qual o nível de redundância pelo LD que deve permanecer entre os marcadores e isso pode ser explorado analisando outros limites de corte ou pode ser modelado por meio de variáveis linguísticas da lógica difusa em trabalhos futuros.

Os filtros padrões (*call rate*, MAF e EHW) usados na base com CQ parecem eliminar marcadores essenciais à explicação do fenótipo PTA para leite. A partir desse estudo torna-se necessário entender quais filtros são responsáveis por eliminar os SNPs mais

relevantes. Portanto, os resultados obtidos são promissores para aplicação em GWAS, pois a maioria dos trabalhos nessa área aplicam filtros padrões no pré-processamento da base de dados.

Uma possibilidade de melhoria no ajuste dos parâmetros do PUK seria a aplicação da metodologia sugerida em Ünstün, Melssen e Buydens (2005), a qual utiliza um AG juntamente com o método Simplex para encontrar, simultaneamente, todos os parâmetros necessários para qualquer *kernel* do SVR.

Existe a possibilidade de adaptação do método desenvolvido para problemas de classificação de indivíduos caso-controle. Para isso, basta mudar o *valor-p* da correlação de Spearman para, por exemplo, o *valor-p* do teste qui-quadrado entre o SNP e o fenótipo. Além disso, a metodologia aqui proposta preconiza um novo método para a seleção genômica objetivando a predição do valor genético do indivíduo a partir do seu genótipo, o que será abordado em trabalhos futuros.

Como trabalho futuro fundamental é necessário avaliar no mapa físico quais as distâncias entre os 3.073 marcadores e suas posições com o objetivo de verificar a distribuição destes ao longo do genoma. Além disso, para verificar a eficiência da metodologia desenvolvida aqui é necessária sua aplicação em outras bases de SNPs relacionados com outros fenótipos.

Agradecimentos

Os autores agradecem ao Centro Nacional de Pesquisa de Gado de Leite (Embrapa Gado de Leite) da Empresa Brasileira de Pesquisa Agropecuária (Embrapa), ao Programa de Pós-graduação em Modelagem Computacional da Universidade Federal de Juiz de Fora (UFJF), à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e à Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG).

4.6 Referências

BAN, H. et al. Identification of type 2 diabetes-associated combination of snps using support vector machine. *BMC Genetics*, v. 11, p. 11–26, 2010.

BROOKES, A. J. The essence of snps. *Gene*, v. 1, n. 234, p. 41–56, 1999.

DRUKER, H. et al. Support vector regression machines. *Advances in Neural Information Processing Systems*, n. 9, p. 155–161, 1997.

FACELLI, K. et al. *Inteligência artificial: uma abordagem de aprendizado de máquina*. Rio de Janeiro: LTC, 2011.

FIELD, A. *Discovering statistics using SPSS*. Thousand Oaks: Sage Publications, 2005.

GIANOLA, D.; PEREZ-ENCISO, M.; TORO, M. A. On marker-assisted prediction of genetic value: beyond the ridge. *Genetics*, v. 163, n. 1, p. 445–455, 2003.

GOLDBERG, D. E. *Genetic algorithms in search, optimization and machine learning*. Boston: Addison-Wesley, 1989. ISBN 0201157675.

HALL, M. et al. *The WEKA Data Mining Software: An Update*. 2009. Disponível em: <http://www.cs.waikato.ac.nz/ml/weka/>. Acesso em: 3.7.2013.

HARRIS, B. L.; JOHNSON, D. L. The impact of high density snp chips on genomic evaluation in dairy cattle. *Interbull Bulletin*, n. 42, p. 40–43, 2010.

KARATZOGLOU, A.; SMOLA, A.; HORNIK, K. kernlab: An s4 package for kernel methods in R. *Journal Statistical Software*, v. 11, n. 9, p. 1–20, 2004.

KOHAVI, G. H. J. R. Wrappers for feature subset selection. *Artificial Intelligence*, v. 97, p. 273–324, 1997.

LINDEN, R. *Algoritmos Genéticos*. 2. ed. Rio de Janeiro: Brasport, 2008. ISBN 978-85-7452-373-6.

MITTAG, F. et al. Use of support vector machines for disease risk prediction in genome-wide association studies: concerns and opportunities. *Human Mutation*, v. 33, n. 12, p. 1708–1718, 2012.

MOORE, F. W. A. J. H.; WILLIAMS, S. M. Bioinformatics challenges for genome-wide association studies. *Gene*, v. 26, n. 4, p. 445–455, 2010.

MORSER, G.; HAYES, B. J.; RAADSMA, H. W. Accuracy of direct genomic values in holstein bulls and cows using subsets of snp markers. *Generics Selection Evolution*, v. 42, n. 37, p. 1–15, 2010.

R CORE TEAM. *R: A language and environment for statistical computing*. 2013. Disponível em: <http://www.Rproject.org/>. Acesso em: 3.7.2013.

ÜNSTüN, B.; MELSSEN, W.; BUYDENS, L. Determination of optimal support vector regression parameters by genetic algorithms and simplex optimization. *Anal. Chim. Acta*, v. 504, p. 292–305, 2005.

ÜNSTüN, B.; MELSSEN, W.; BUYDENS, L. Facilitating the application of support vector regression by using a universal Pearson VII function based kernel. *Chemometrics and Intelligent Laboratory Systems*, v. 81, p. 29–40, 2006.

VERNEQUE, R. da S. et al. *Programa Nacional de Melhoramento Genético do Gir Leiteiro: sumário brasileiro de touros – resultado do teste de progênie e terceira prova de pré-seleção de touros - maio/2012*. Juiz de Fora, 2012. 70 p. (Série Documentos, 152).

WEI, Z. et al. From disease association to risk assessment: An optimistic view from genome-wide association studies on type 1 diabetes. *PLoS Genetics*, v. 5, n. 10, p. 1–11, 2009.

5

Seleção de marcadores genômicos com busca ordenada e um classificador de larga margem

Saulo Moraes Villela
Saul de Castro Leite
Raul Fonseca Neto

CLASSIFICADORES DE larga margem, como as máquinas de vetores de suporte, têm sido amplamente utilizados com bastante eficiência em problemas que possuem uma grande quantidade de variáveis. Nesse sentido, torna-se importante a utilização de estratégias que utilizem critérios de seleção associados a esse tipo de classificador. Esse trabalho apresenta um método para a seleção de subconjuntos de variáveis através de um processo de busca ordenada, também conhecido como *best-first*, para exploração do espaço de possíveis candidatos. O algoritmo, denominado AOS, utiliza como medida de avaliação os valores de margem calculados a partir da utilização de um classificador de larga margem. Esse classificador, de grande eficiência computacional, permite grande flexibilidade e rapidez na obtenção dos valores de margem possibilitando a solução de problemas de tamanho razoável sem que ocorra explosão combinatória. O algoritmo foi testado em diferentes

problemas da literatura e seus resultados comparados à outras técnicas de seleção de subconjuntos. Uma importante contribuição do algoritmo se refere à utilização do conceito de margem projetada. Esse valor, computado como a projeção da margem real de um espaço em cada subespaço de dimensão inferior, permite maior eficiência e rapidez na solução dos problemas de classificação e, portanto, no processo de busca como um todo.

5.1 Introdução

Considere, por exemplo, a tarefa de se discriminar entre sexos uma determinada pessoa. O problema é de solução trivial até mesmo para uma criança. Isso ocorre porque ela selecionará certas características que melhor discriminam as pessoas entre as classes (no caso masculino e feminino) e ignorará diversas outras. Agindo dessa maneira ela será capaz de classificar com velocidade, e sem erro, os exemplos a ela apresentados.

De maneira oposta, os classificadores automáticos irão se basear em todas as características dos padrões apresentados, para só então fornecer uma resposta. Portanto, o processo de classificação se torna lento e o número de erros relacionados pode crescer na medida em que a quantidade de atributos para representar os dados aumenta.

Esse fato é de fácil compreensão quando se imagina que um classificador, para efetuar a discriminação citada, utiliza todos os atributos relativos a uma pessoa, como cor dos olhos, cor dos cabelos, cor da pele, altura, peso etc..

O controle da quantidade de erros de classificação é, portanto, essencial em problemas reais. Com isso, uma das formas de se melhorar o desempenho de um classificador, baseia-se em construir meios de selecionar as características que mais influenciam no processo de classificação ou discriminação.

O objetivo principal do processo de seleção de características é a eliminação de variáveis irrelevantes com o intuito de produzir subconjuntos de variáveis relevantes que sejam capazes de generalizarem melhor para um dado problema de classificação. Também, podem-se destacar como importantes questões relativas ao requerimento de tempo de computação, descoberta de variáveis que têm maior poder discriminante, como em análise de genes, bem como uma melhor visualização e interpretação dos resultados. Nesse sentido, considera-se nesse trabalho a investigação da eficiência da utilização das máquinas de vetores de suporte associadas a um processo ordenado de seleção de candidatos na obtenção dos subconjuntos com maior poder de generalização. Adota-se, para tanto, uma estratégia de solução do tipo reversa na qual as variáveis com menor poder de discriminação são retiradas do problema. Ao contrário dos algoritmos míopes, que retiram uma variável por vez de forma irrevogável definindo uma sequência de subconjuntos aninhados, emprega-se um processo de busca ordenada que gera uma árvore de possibilidades e permite uma maior exploração da interdependência entre o conjunto de variáveis do problema.

Como forma de evitar a explosão combinatória decorrente do número exponencial de possibilidades, utiliza-se de duas estratégias que permitem controlar o processo de busca. Primeiramente, a quantidade de variáveis é reduzida até um tamanho gerenciável com a utilização de uma técnica míope, como a eliminação de variáveis sugerida pelo algoritmo Recursive Feature Elimination (RFE) ou por métodos de filtragem que se baseiam no estabelecimento de um *ranking* de variáveis segundo medidas obtidas por critérios estatísticos ou de informação. Em segundo lugar, limita-se o fator de ramificação com a retirada de no máximo três possíveis variáveis a cada nível da árvore de busca, além de se eliminar estados que não tendem a gerar boas soluções.

Com isso, o objetivo do trabalho é a introdução de um algoritmo de seleção de características, AOS, que utiliza critérios e medidas provenientes de classificadores de larga margem. O algoritmo apresenta um processo eficaz para selecionar as características que melhor representam os dados, diminuindo, assim, o espaço representativo do problema.

5.2 Classificadores de larga margem

Rosenblatt (1958) apresentou um algoritmo extremamente simples e eficaz para o treinamento de uma unidade lógica *threshold*, capaz de encontrar uma solução na forma de um classificador linear ou hiperplano. Block (1962) e Novikoff (1963) mostraram que este algoritmo, denominado Perceptron, convergia em um número finito de iterações, caso o conjunto de pontos fosse linearmente separável. Tal algoritmo está diretamente relacionado à técnica matemática de relaxação desenvolvida para a determinação de uma solução viável para um sistema de inequações (AGMON, 1954; MOTZKIN; SCHOENBERG, 1954). Entretanto, como a maior parte dos problemas relacionados ao aprendizado de hipóteses ou aprendizado de conceitos é de natureza não linearmente separável, Minsky e Papert (1969) em sua obra “Perceptron”, lançaram uma série de dúvidas sobre a capacidade de aprendizado desses classificadores lineares.

Anteriormente, Aizerman, Braverman e Rozoner (1964) mostraram a possibilidade de utilização de funções *kernel* pelo algoritmo Perceptron como forma de resolver problemas não linearmente separáveis. Entretanto, somente quando Boser, Guyon e Vapnik (1992) sugeriram a utilização de funções *kernel* em classificadores de ótima margem, o procedimento ficou consagrado e bastante difundido na comunidade de aprendizado de máquina. Desde então, a partir do desenvolvimento de classificadores *kernel*,

abriu-se uma nova perspectiva no uso de classificadores lineares, como o algoritmo Perceptron, em problemas de reconhecimento de padrões.

5.2.1 Problema de classificação linear

Seja $Z = \{z_i = (x_i, y_i) : i \in \{1, \cdots, m\}\}$ um conjunto de treinamento composto de pontos $x_i \in \mathbb{R}^d$ e rótulos (classes) $y_i \in \{-1, 1\}$. Além disso, sejam Z^+ e Z^- definidos como os conjuntos $\{(x_i, y_i) \in Z : y_i = 1\}$ e $\{(x_i, y_i) \in Z : y_i = -1\}$, respectivamente. Um problema de classificação linear consiste em encontrar um hiperplano, que é dado pelo seu vetor normal $w \in \mathbb{R}^d$ e uma constante $b \in \mathbb{R}$, de tal forma que os pontos em Z^+ e Z^- sejam separados nos dois semi-espaços gerados por ele. Assim, define-se (w, b) tal que:

$$y_i \left(w \cdot x_i + b\right) \geq 0, \text{ para todo } (x_i, y_i) \in Z.$$

Esse hiperplano pode não existir. Quando isso acontece, o conjunto Z é denominado não linearmente separável. É dito que Z aceita uma margem $\gamma \geq 0$ quando existe um hiperplano $\mathcal{H} := \{x \in \mathbb{R}^d : w \cdot x + b = 0\}$ tal que:

$$y_i \left(w \cdot x_i + b\right) \geq \gamma, \text{ para todo } (x_i, y_i) \in Z.$$

Nesse caso, define-se dois hiperplanos adicionais paralelos à $\mathcal{H}$, dados por $\mathcal{H}^+ := \{x \in \mathbb{R}^d : w \cdot x + (b - \gamma) = 0\}$ e $\mathcal{H}^- := \left\{x \in \mathbb{R}^d : w \cdot x + (b + \gamma) = 0\right\}$.

A distância entre estes dois hiperplanos paralelos sob uma norma p é dada por (DAX, 2006):

$$\text{dist}(\mathcal{H}^+, \mathcal{H}^-) = \frac{(-b + \gamma + b + \gamma)}{||w||_q} = \frac{2\gamma}{||w||_q},$$

onde q é tal que $\frac{1}{p} + \frac{1}{q} = 1$. Sendo $\gamma_g := \text{dist}(\mathcal{H}^+, \mathcal{H}^-)/2$, chama-se γ_g de margem geométrica (com norma p) entre os dois

hiperplanos $\mathcal{H}^+$ and $\mathcal{H}^-$. Sendo assim, é dito que Z aceita uma margem geométrica ($\gamma_g \geq 0$) quando existe um hiperplano com (w, b) tal que:

$$y_i(w \cdot x_i + b) \geq \gamma_g ||w||_q, \text{ para todo } (x_i, y_i) \in Z. \tag{5.1}$$

5.2.2 Perceptron de Margem Geométrica Fixa

Dado um valor de margem fixo γ_f e um conjunto de treinamento Z, que aceita γ_f como uma margem geométrica, isto é, satisfaz a Expressão (5.1), considera-se o problema de encontrar um hiperplano separador com dados (w, b) tal que:

$$y_i(w \cdot x_i + b) \geq \gamma_f ||w||_q, \text{ para todo } (x_i, y_i) \in Z. \tag{5.2}$$

A formulação desse problema, para a norma euclidiana, foi proposta em Leite e Fonseca Neto (2008), sendo de modo similar ao Perceptron de Rosenblatt (ROSENBLATT, 1958), e o algoritmo resultante foi denominado de Perceptron de Margem Geométrica Fixa (Geometric Fixed Margin Perceptron – GFMP).

Para uma norma p, a função de erro, ou perda, $J : \mathbb{R}^{d+1} \rightarrow \mathbb{R}$ é definida como:

$$J(w, b) := \sum_{(x_i, y_i) \in \mathcal{M}} \gamma_f ||w||_q - y_i \left(w \cdot x_i + b\right),$$

onde $\mathcal{M}$ é o subconjunto de Z que viola (5.2) para os dados (w, b), assim $\mathcal{M} := \{(x_i, y_i) \in Z : y_i \left(w \cdot x_i + b\right) < \gamma_f ||w||_q\}$.

Seguindo a abordagem *on-line* do gradiente estocástico, o processo de minimização começa com valores iniciais (w^0, b^0), normalmente $(0, 0)$. A cada iteração do algoritmo, um par $z_i = (x_i, y_i)$ é escolhido e verificado contra (w^t, b^t), $t \in \{1, 2, \dots\}$. Se o par z_i escolhido for um erro, ou seja, se $y_i \left(w^t \cdot x_i + b^t\right) < \gamma_f ||w^t||_q$, então um novo vetor normal w^{t+1} e a constante b^{t+1} são construídos

usando o gradiente de J. Tomando as derivadas parciais de J em relação à w_j, $j \in \{1, \ldots, d\}$, e à b quando $1 < q < \infty$, tem-se:

$$\begin{aligned}
\frac{\partial J(w,b)}{\partial w_j} &= \gamma_f \left(\sum_{i=1}^{d} |w_i|^q \right)^{\frac{1}{q}-1} |w_j|^{q-1} \text{sinal}(w_j) - y_i x_{ij} \\
\frac{\partial J(w,b)}{\partial b} &= -y_i
\end{aligned}$$

Sendo assim, para $1 < q < \infty$, obtém-se a seguinte regra de correção para o GFMP quando z_i for um erro:

$$\begin{aligned}
w^{t+1} &= w^t - \eta \left(\gamma_f ||w^t||_q^{1-q} |w^t|^{q-1} \text{sinal}(w^t) + y_i x_i \right) \\
b^{t+1} &= b^t + \eta y_i,
\end{aligned}$$

onde $\eta \in (0, 1]$ é a taxa de aprendizado, $|w| := (|w_i|, \ldots, |w_d|)'$, e $\text{sinal}(w) := (\text{sinal}(w_i), \ldots, \text{sinal}(w_d))'$.

Para se ter uma regra de atualização para os casos onde $p = 1$ ou $p = \infty$, pega-se o limite $q \uparrow \infty$ e $q \downarrow 1$, respectivamente, na expressão da $\partial J / \partial w_j$. Para $p = \infty$, a análise é trivial, seguindo a regra derivada

$$w^{t+1} = w^t - \eta \left(\gamma_f \text{sinal}(w^t) + y_i x_i \right).$$

Para $p = 1$, reescreve-se $\partial J / \partial w_j$ como:

$$\frac{\partial J(w,b)}{\partial w_j} = \frac{|w_j|^{q-1}}{\sum_{i=1}^{d} |w_i|^q} \gamma_f ||w||_q \text{sinal}(w_j) - y_i x_{ij},$$

e considera-se o limite:

$$L := \lim_{q \to \infty} \frac{|w_j|^{q-1}}{\sum_{i=1}^{d} |w_i|^q}.$$

Nota-se que:

$$\frac{|w_j|^{q-1}}{d \max_i |w_i|^q} \leq \frac{|w_j|^{q-1}}{\sum_{i=1}^{d} |w_i|^q} \leq \frac{|w_j|^{q-1}}{\max_i |w_i|^q}.$$

Assim, para $|w_j| < \max_i |w_i|$, tem-se $L = 0$.

Agora, para tratar o caso onde $|w_j| = \max_i |w_i|$, seja $M := \max_i |w_i|$ e o conjunto de índices

$$N := \{k \in \{1, \ldots, d\} : |w_k| = M\}$$

e seja n a cardinalidade de N. Então, tem-se:

$$\begin{aligned} L &= \lim_{q\to\infty} \frac{M^{q-1}}{nM^q + \sum_{i\notin N} w_i|^q} \times \frac{\frac{1}{(d-n)M^q}}{\frac{1}{(d-n)M^q}} \\ &= \lim_{q\to\infty} \frac{1/(d-n)M^{-1}}{n/(d-n) + \frac{\sum_{i\notin N}|w_i|^q}{(d-n)M^q}} = \frac{1}{nM} = \frac{1}{n||w||_\infty}, \end{aligned}$$

e, com isso, tem-se a regra de atualização:

$$w_j^{t+1} = \begin{cases} w_j^t - \eta\left(\gamma_f \text{sinal}(w_j^t)/n^t + y_i x_{ij}\right) & \text{se } |w_j| = ||w||_\infty \\ w_j^t - \eta\left(y_i x_{ij}\right) & \text{se } |w_j| < ||w||_\infty \end{cases}$$

onde n^t é a cardinalidade do conjunto

$$N^t := \{k \in \{1, \ldots, d\} : |w_k^t| = ||w^t||_\infty\}$$

O Algoritmo 2 descreve o Perceptron de Margem Geométrica Fixa (GFMP) em sua formulação primal.

Para que o novo algoritmo tenha o poder de classificação de uma máquina *kernel*, é necessário que o novo modelo Perceptron e o processo de otimização sejam desenvolvidos no plano de variáveis duais. Sendo assim, a equação de correção deve ser modificada, a fim de proporcionar a correção dos respectivos multiplicadores. Considerando a expansão do vetor w em função do conjunto de variáveis duais, tem-se, para a ocorrência de um erro, representado pela condição $y_i\left(\sum_i \alpha_i K\left(x_i, x\right)\right) < \gamma_f \cdot ||w||_2$, o valor do multiplicador associado atualizado pela expressão:

$$\alpha_i = \alpha_i + \eta \cdot 1,$$

Algoritmo 2: Perceptron de Margem Geométrica Fixa Primal

Entrada: conjunto $Z = \{(x_i, y_i)\}, i \in \{1, \ldots, m\}$;
margem geométrica fixa γ_f;
limite superior no número de iterações max;
Saída: vetor de pesos w e bias b;

início
  inicializar (w^0, b^0);
  $j \leftarrow 0$;
  $t \leftarrow 0$;
  $stop \leftarrow$ falso;
  enquanto $j \leq max$ **e** $\neg stop$ **faça**
    $erro \leftarrow$ falso;
    para i **de** 1 **até** m **faça**
      se $y_i \left(w^t \cdot x_i + b^t\right) < \gamma_f ||w^t||_q$ **então**
        $w^{t+1} \leftarrow$
        $w^t - \eta \left(\gamma_f ||w^t||_q^{1-q} |w^t|^{q-1} \text{sinal}(w^t) + y_i x_i\right)$;
        $b^{t+1} \leftarrow b^t + \eta y_i$;
        $t \leftarrow t + 1$;
        $erro \leftarrow$ verdadeiro;
      fim se
    fim para
    se $\neg erro$ **então**
      $stop \leftarrow$ verdadeiro;
    fim se
    $j \leftarrow j + 1$;
  fim enquanto
fim

após a realização de um escalonamento no valor do vetor de multiplicadores, representado por:

$$\alpha_{(t+1)} = \alpha_{(t)} \cdot \left(1 - \left(\eta \cdot \gamma_f\right) / ||w||_2\right),$$

para as componentes não nulas do vetor.

A fração $\left(\eta \cdot \gamma_f\right) / ||w||_2$ é responsável pelo decréscimo no valor dos multiplicadores, sendo de fundamental importância na correta determinação dos vetores de suporte.

A atualização do parâmetro *bias* é feita de modo independente, considerando a existência de uma componente adicional nos pontos de valor igual a $+1$. Essa atualização pode ser *online*, a cada apresentação de uma amostra, ou na forma *batch*, considerando a ocorrência de toda uma época no processo de treinamento.

Se for considerada a execução do algoritmo na sua forma dual, torna-se necessário avaliar o valor da norma do vetor w, $||w||_2$, utilizando somente as variáveis duais. Para tanto, considera-se o cálculo da norma com base na seguinte expressão:

$$\begin{aligned} w^t w &= \left(\sum_i \alpha_i y_i \Phi\left(x_i\right) \right)^t \left(\sum_j \alpha_j y_j \Phi\left(x_j\right) \right) \\ &= \sum_i \sum_j \alpha_i y_i \cdot \alpha_j y_j \cdot K\left(x_i, x_j\right) \end{aligned}$$

$$||w||_2 = \left(\sum_i \sum_j \alpha_i y_i \cdot \alpha_j y_j \cdot K\left(x_i, x_j\right) \right)^{\frac{1}{2}}$$

Para aumentar a precisão do algoritmo, tornando-o compatível com o processo *online* de aprendizado, e diminuir o esforço de cálculo do valor da norma, é possível adotar uma forma aproximada de atualização do valor da mesma, que considera somente o efeito da modificação, após a ocorrência de cada atualização de um multiplicador α_i, na forma:

$$||w||_2^2 \cong ||w||_2^2 + \sum_j \left(\Delta\alpha_i y_i\right) \cdot \alpha_j y_j \cdot K\left(x_i, x_j\right)$$

Durante a atualização do valor da norma, a cada modificação no valor de uma variável dual, deve-se tomar uma atenção especial para que não haja um argumento negativo ou de valor zero relacionado ao valor do produto interno $w^t w$. Entretanto, a garantia de que este valor não seja negativo está associada à propriedade da matriz *kernel* ser semi-definida positiva. Ainda assim, tem-se a possibilidade de o vetor w assumir o vetor nulo. Tal situação pode ser facilmente evitada se for garantida, igualmente, a propriedade de positividade estrita da matriz *kernel*, ou seja:

$$\sum_i \sum_j \alpha_i y_i \cdot \alpha_j y_j \cdot K\left(x_i, x_j\right) > 0,$$

para quaisquer escalares representados pelos produtos $\alpha_i y_i$ e $\alpha_j y_j$.

É possível assegurar a propriedade de positividade da matriz *kernel* aumentando adequadamente o valor da sua diagonal na forma $K = K + \lambda I$. Essa operação está diretamente relacionada à imposição de uma margem flexível na construção do classificador. Nesse sentido, quanto maior o valor do parâmetro λ, maior poderá ser o valor da margem a ser obtido, observando-se, entretanto, o grau de violação das restrições, possibilitando uma melhor generalização do classificador.

Em uma perspectiva Bayesiana, Herbrich (2002) e Shawe-Taylor e Cristianini (1999), onde a matriz *kernel* é vista como a matriz de covariância dos dados, a adição do valor λ às componentes da diagonal principal pode ser interpretada como a soma de uma variância, σ_i^2, relacionada à existência de um ruído no valor de saída, $y_i \left(w \cdot (x_i) + b\right)$, de todos os exemplos do conjunto de treinamento. Nessa visão estatística, pode-se considerar que a escolha da função *kernel*, bem como da variância associada, constituem todo o conhecimento a *priori* que pode ser considerado ou incluído na construção do classificador.

Entretanto, como observado por Shawe-Taylor e Cristianini (2004), deve haver uma certa precaução na soma de valores às componentes diagonais da matriz *kernel*. A existência de uma diagonal com valores demasiadamente altos em relação aos valores situados fora da diagonal pode causar uma espécie de *overfitting* devido ao fato de o *kernel* representar de forma acentuada o conceito de identidade. Da mesma forma, uma matriz *kernel* com valores muito uniformes, pode levar à ocorrência de *underfitting*.

O Algoritmo 3 descreve o pseudocódigo do algoritmo dual relativo ao treinamento do Perceptron de Margem Geométrica Fixa (GFMP).

Algoritmo 3: Perceptron de Margem Geométrica Fixa Dual

Entrada: conjunto $Z = \{(x_i, y_i)\}$, $i \in \{1, \ldots, m\}$;
margem geométrica fixa γ_f;
limite superior no número de iterações max;
Saída: vetor de multiplicadores α e bias b;

início
  $stop \leftarrow$ falso;
  inicializar α;
  $j \leftarrow 0$;
  $b \leftarrow 0$;
  $norma \leftarrow \left(\sum_i \sum_j \alpha_i y_i \cdot \alpha_j y_j \cdot K(x_i, x_j)\right)^{\frac{1}{2}}$;
  enquanto $j \leq max$ **e** $\neg stop$ **faça**
    $erro \leftarrow$ falso;
    para i **de** 1 **até** m **faça**
      $\alpha_i \leftarrow \alpha_i - \delta$;
    fim para
    para i **de** 1 **até** m **faça**
      se $y_i \left(\sum_j \alpha_j \cdot K(x_i, x_j)\right) < \gamma_f \cdot norma$ **então**
        $\alpha_i \leftarrow \alpha_i + \eta \cdot 1 + \delta$;
        $norma \leftarrow \left(norma^2 + \sum_j (\Delta\alpha_i y_i) \cdot \alpha_j y_j \cdot K(x_i, x_j)\right)^{\frac{1}{2}}$;
        $erro \leftarrow$ verdadeiro;
      fim se
    fim para
    $b \leftarrow b + \Delta\alpha_i y_i$;
    $norma \leftarrow \left(\sum_i \sum_j \alpha_i y_i \cdot \alpha_j y_j \cdot K(x_i, x_j)\right)^{\frac{1}{2}}$;
    se $\neg erro$ **então**
      $stop \leftarrow$ verdadeiro;
    fim se
    $j \leftarrow j + 1$;
  fim enquanto
fim

5.2.3 Algoritmo de Margem Incremental

Uma nova formulação para o problema de maximização da margem foi proposta por Leite e Fonseca Neto (2008) e desenvolvida a partir da constatação de que, na obtenção da máxima margem, os pontos ou vetores de suporte de classes contrárias se encontram à mesma distância do hiperplano separador, ou seja,

considerando as margens das classes de rótulos positivo e negativo, tem-se $\gamma^+ = \gamma^-$, onde:

$$\gamma^+ = \text{Min } y_i \left(w \cdot x_i + b\right), \text{ para todo } x_i \in Z^+$$
$$\gamma^- = \text{Min } y_i \left(w \cdot x_i + b\right), \text{ para todo } x_i \in Z^-$$

Considerando a possibilidade da obtenção de soluções de margem fixa, em um número finito de correções, na solução do problema do GFMP:

$$y_i \left(w \cdot x_i + b\right) \geq \gamma_f \cdot ||w||_q, \text{ para valores de } \gamma_f < \gamma^*,$$

onde γ^* é a máxima margem, os autores propõem a formulação e solução aproximada do problema de máxima margem, considerando a maximização explícita e direta da margem geométrica. Adaptando-se para uma norma p, deve-se resolver o seguinte problema de otimização:

$$\text{Max}_{\text{w}}\gamma_{\text{g}}, \text{ sujeito a } \text{y}_{\text{i}} \left(\text{w} \cdot \text{x}_{\text{i}} + \text{b}\right) \geq \gamma_{\text{g}} \cdot ||\text{w}||_{\text{q}}$$

A técnica de solução desenvolvida consiste em uma estratégia de aprendizado incremental, através da qual são obtidas sucessivas soluções do problema do GFMP, para valores crescentes de margem. Esse parâmetro inicia com o valor zero e tem seus valores incrementados de forma consistente, até aproximar-se do valor da margem máxima. Ou seja, para um conjunto de valores $\gamma_f \in [0..\gamma^*)$, sendo:

$$\gamma_f^{t+1} > \gamma_f^t, \text{ para } t = 1, \ldots, T-1, \gamma_f^1 = 0, \gamma_f^T \approx \gamma^*,$$

soluciona-se, sucessivamente, o problema de inequações não lineares:

$$y_i \left(w \cdot x_i + b\right) \geq \gamma_f \cdot ||w||_q, i \in 1, \ldots, m,$$

sendo cada solução equivalente à solução do problema do Perceptron de Margem Geométrica Fixa.

Para a atualização, a cada iteração, do valor da margem fixa, adotam-se duas regras, baseadas em uma estratégia de balanceamento, que garantem um direcionamento para a solução de máxima margem:

Primeira regra: caso a solução do problema forneça as margens, negativa e positiva, diferentes, pode-se dizer que a solução obtida não caracteriza uma solução de máxima margem. Portanto, corrige-se o valor da margem fixa na forma:

$$\gamma_f^{t+1} = \frac{\gamma^+ + \gamma^-}{2},$$

onde γ^+ e γ^- são os valores relacionados, respectivamente, às menores distâncias projetadas dos pontos do conjunto Z^+ e Z^- ao hiperplano separador da t-ésima iteração.

Pode-se observar que, nesse caso, haverá sempre a garantia de solução do novo problema, já que a nova margem fixa é sempre inferior à margem ótima, ou seja,

$$\gamma_f^{t+1} = \left(\gamma^+ + \gamma^-\right)/2 < \gamma^*$$

Tal condição deriva do fato de que se as margens negativa e positiva são desiguais, então a margem total não é máxima, implicando $\gamma^+ + \gamma^- < 2 \cdot \gamma^*$. Também se tem a garantia de que a nova margem fixa obtida é superior à margem fixa anterior, ou seja: $\gamma_f^{t+1} > \gamma_f^t$. Tal condição deriva-se da constatação das seguintes relações de exclusividade:

$$\gamma^+ > \gamma_f^t \text{ e } \gamma^- \geq \gamma_f^t \text{ ou } \gamma^+ \geq \gamma_f^t \text{ e } \gamma^- > \gamma_f^t,$$

garantindo um incremento no valor da nova margem fixa.

Segunda regra: caso a solução do problema forneça as margens, negativa e positiva, iguais, pode ser que a solução obtida seja uma solução de ótimo local. Portanto, torna-se necessário garantir um acréscimo no valor da nova margem fixa, na forma:

$$\gamma_f^{t+1} = (1 + \Delta)\, \gamma_f^t,$$

sendo $\Delta \in (0, 1)$ uma constante de incremento.

Entretanto, adotando esse incremento, não se tem mais a garantia de solução do novo problema, já que o novo valor da margem fixa poderá ser igual ou maior que o valor da margem ótima, ou seja, $\gamma_f^{t+1} \geq \gamma^*$. Para a solução desse contratempo, é suficiente a imposição de um número máximo de iterações no número de épocas do algoritmo de treinamento, a partir do qual, caso não haja uma nova solução do problema GFMP, adota-se como margem obtida o valor anterior da margem fixa, relacionado à última solução.

Uma forma adequada de escolha para essa constante é a de um valor proporcional à aproximação de margem desejada, isto é, se existe um desejo de uma aproximação α da margem ótima (ou seja, a margem final obtida pelo algoritmo é igual ou superior a $(1 - \alpha)\gamma^*, \alpha \in (0, 1)$), o valor de Δ deve ser escolhido como o valor de α. A prova disto é que, considerando um $t \geq 1$, tem-se $\gamma_f^t \in (0, \gamma^*)$ e $\gamma_f^{t+1} = (1 + \alpha)\gamma_f^t$. Consequentemente, $\gamma_f^t \geq \gamma^*/(1+\alpha) > (1-\alpha)\gamma^*$. Sendo assim, adotando-se o último valor viável de margem γ^w, assegura-se que a última margem de parada é superior à aproximação desejada, ou seja,

$$\gamma^w \geq \gamma_f^t \geq (1 - \alpha)\gamma^*$$

Adotando-se como solução inicial, ou ponto de partida, a solução do problema GFMP anterior, pode-se reduzir de forma gradativa a diferença existente entre o valor da margem final e as sucessivas margens fixas, tornando mais rápida a obtenção, a cada problema, do novo hiperplano separador.

O algoritmo de aprendizado (margem) incremental (Incremental Margin Algorithm – IMA) possui um *loop* principal relacionado às correções do valor da margem fixa e um procedimento aninhado relativo ao algoritmo de treinamento do GFMP. Os Algoritmos 4 e 5 apresentam as descrições do IMA considerando a

chamada, a cada iteração, do algoritmo de treinamento do GFMP, resolvido no seu plano primal ou dual, no sentido de fornecer uma solução viável até a obtenção do hiperplano separador final.

Algoritmo 4: Algoritmo de Margem Incremental Primal

Entrada: conjunto $Z = \{(x_i, y_i)\}, i \in \{1, \ldots, m\}$;
limite superior no número de iterações max;
Saída: vetor de pesos w e bias b;
margem de parada γ^w;
início
  $\gamma_f \leftarrow 0$;
  repita
    $(w, b) \leftarrow \text{GFMP}(Z, \gamma_f, max)$;
    $\gamma^+ \leftarrow \text{Min}_{i+} (w \cdot x_i + b)$;
    $\gamma^- \leftarrow \text{Min}_{i-} (w \cdot x_i + b)$;
    $\gamma^w \leftarrow \text{Min} \left(\gamma^+, \gamma^-\right)$;
    se $\gamma^+ \neq \gamma^-$ **então**
      $\gamma_f \leftarrow \left(\gamma^+ + \gamma^-\right)/2$;
    senão
      $\gamma_f \leftarrow (1 + \Delta)\, \gamma_f$;
    fim se
  até *que a convergência do GFMP em* max *iterações não seja atingida*;
fim

Algoritmo 5: Algoritmo de Margem Incremental Dual

Entrada: conjunto $Z = \{(x_i, y_i)\}, i \in \{1, \ldots, m\}$;
limite superior no número de iterações max;
Saída: vetor de multiplicadores α e bias b;
margem de parada γ^w;
início
  $\gamma_f \leftarrow 0$;
  repita
    $(\alpha, b) \leftarrow \text{DualGMFP}(Z, \gamma_f, max)$;
    $norma \leftarrow \left(\sum_i \sum_j \alpha_i y_i \cdot \alpha_j y_j \cdot K(x_i, x_j)\right)^{\frac{1}{2}}$;
    $\gamma^+ \leftarrow \text{Min}_{i+} \left\{y_i \cdot \sum_j \alpha_j y_j \cdot K(x_i, x_j)\right\} / norma$;
    $\gamma^- \leftarrow \text{Min}_{i-} \left\{y_i \cdot \sum_j \alpha_j y_j \cdot K(x_i, x_j)\right\} / norma$;
    $\gamma^w \leftarrow \text{Min} \left(\gamma^+, \gamma^-\right)$;
    se $\gamma^+ \neq \gamma^-$ **então**
      $\gamma_f \leftarrow \left(\gamma^+ + \gamma^-\right)/2$;
    senão
      $\gamma_f \leftarrow (1 + \Delta)\, \gamma_f$;
    fim se
  até *que a convergência do DualGFMP em* max *iterações não seja atingida*;
fim

5.2.4 Formulações especiais

Seja a minimização da norma L_1 do vetor w, definindo um hiperplano separador com margem L_∞. Ou seja, a distância computada dos pontos ao hiperplano separador é tal que maximiza o valor da maior componente do vetor normal. Nesse caso, é possível constatar que a solução, ou hiperplano, proveniente da formulação L_∞ se posiciona sempre quase perpendicular ao eixo da maior componente, tornando-se dependente dessa coordenada. Tal afirmativa (ROSSET; ZHU; HASTIE, 2004) pode ser evidenciada pelo fato de que a projeção L_∞, considerando somente a maior coordenada, pode desprezar as coordenadas com valor nulo sem afetar o valor da distância projetada.

De outra forma, para a minimização da norma L_∞ do vetor w, definindo um hiperplano separador com margem L_1, ou seja, a distância dos pontos ao hiperplano separador é tal que maximiza o somatório em módulo das componentes do vetor normal, Pedroso e Murata (2001) propõem uma formulação que culmina na solução de um único problema de Programação Linear. Considerando o fato de que: $||w||_\infty = \text{Max}_i|w_i|$, o problema de maximização da margem, segundo Pedroso e Murata (2001), pode ser reformulado como:

$$\text{Max}_w\text{Min}_{x_i \in X} \left\{ y_i \left(w \cdot x_i + b \right) / \text{Max}_i|w_i| \right\}$$

Limitando o valor da margem funcional, de modo que $\text{Min}_{x_i \in X} \left\{ y_i \cdot (w \cdot x_i + b) \right\} = 1$, o problema de otimização se torna equivalente a

$$\text{Max}_\text{w} \left\{ 1/\text{Max}_\text{i}|\text{w}_\text{i}| \right\}, \text{ sujeito a } \text{y}_\text{i} \left(\text{w} \cdot \text{x}_\text{i} + \text{b} \right) \geq 1$$

ou

$$\text{Min z, sujeito a } \text{y}_\text{i} \left(\text{w} \cdot \text{x}_\text{i} + \text{b} \right) \geq 1, \ \text{z} \geq +\text{w}_\text{i} \text{ e z} \geq -\text{w}_\text{i}$$

Mesmo com a possibilidade de se resolver um único problema de Programação Linear, percebe-se que a condução do processo de otimização no espaço primal para as formulações L_1 e L_∞, torna inviável a solução de problemas de altíssima dimensão, já que a matriz de coeficientes possuirá posto relacionado à quantidade destes parâmetros. Nesse sentido, verifica-se como oportuna a utilização da formulação proposta na solução do problema de maximização da margem para quaisquer valores de norma.

5.3 Seleção de características

O processo de seleção de características (*feature selection* ou *feature subset selection*), ou seleção de atributos, tem sido um tradicional tópico de pesquisa desde os anos 60 (HUGHES, 1968), e 70 (MUCCIARDI; GOSE, 1971). O processo baseia-se na seleção, segundo algum critério, de um subconjunto do conjunto original de características do problema que produza os mesmos (ou quase mesmos) resultados.

Entre as vantagens em se utilizar seleção de características, é possível citar o fato de que, após a sua aplicação, a dimensionalidade do espaço representativo do problema é reduzida. Assim, o processo remove atributos redundantes (altamente correlacionados e que não agregam informação) e irrelevantes (que não contêm informações úteis). Sua aplicação traz resultados imediatos. Dentre eles, pode-se destacar:

- melhoria da qualidade dos dados;
- extração de conhecimento mais compreensível;
- obtenção de uma maior velocidade na execução dos algoritmos de aprendizado e, consequentemente, no processo de classificação.

5.3.1 Estudo do problema e graus de relevância

Vários problemas requerem o aprendizado de uma função de classificação apropriada. A função atribui um dado padrão de entrada a uma das finitas classes do problema. A escolha das características, atributos, ou medidas usadas para representar os padrões que serão apresentados ao classificador afetam, dentre outros aspectos:

- a precisão da função de classificação;
- o tempo necessário para aprender uma função de classificação;
- o número de exemplos necessário para se aprender uma função de classificação;
- o custo de se executar a classificação usando a função de discriminação aprendida;
- a compreensibilidade do conhecimento adquirido através do treinamento.

Determinar quais características são relevantes à tarefa de aprendizado é uma grande preocupação dos que lidam com aprendizado de máquina. Isso se deve ao fato de que a inclusão de uma característica irrelevante, ou redundante, pode ocasionar uma redução drástica no desempenho dos algoritmos.

Para determinar se uma característica específica é ou não relevante ao processo de aprendizado e classificação, é preciso compreender os conceitos de relevância. Existem várias definições na literatura de aprendizado de máquina sobre o significado de uma característica ser “relevante”. John, Kohavi e Pfleger (1994) e Guyon (2001) definem duas notações para relevância:

- **relevância forte:** um atributo f_i é considerado fortemente relevante se a distribuição de probabilidade dos valores das classes, dado o conjunto completo de características F, modificar-se quando f_i for removido, ou seja, a remoção do atributo causa uma degradação no classificador;

- **relevância fraca:** um atributo f_i é considerado fracamente relevante se ele não for fortemente relevante e existir um subconjunto de variáveis $F^{'}(F^{'} \subset F)$ em que o desempenho do classificador usando $F^{'} \cup \{f_i\}$ é superior ao desempenho do mesmo classificador utilizado somente sobre subconjunto $F^{'}$.

Por sua vez, Yu e Liu (2004) dividem as características fracamente relevantes em redundantes e não redundantes. Uma característica é redundante em relação a outra se seus valores estão correlacionados.

Existem, ainda, características que não possuem relevância forte e nem fraca e, por isso, são denominadas de irrelevantes. As características redundantes e irrelevantes não devem ser selecionadas.

5.3.2 Características gerais

Uma forma conveniente para representar as abordagens para a seleção de atributos é a busca heurística, na qual cada estado no espaço de busca é composto por um subconjunto de possíveis atributos. A Figura 5.1 mostra subconjuntos possíveis para quatro atributos: círculos brancos indicam que o atributo em questão não foi selecionado; círculos pretos indicam a seleção do atributo. Cada estado no domínio de subconjuntos de atributos especifica quais atributos foram selecionados. Nota-se que os estados são parcialmente ordenados, sendo que os estados à direita acrescentam um atributo aos da esquerda.

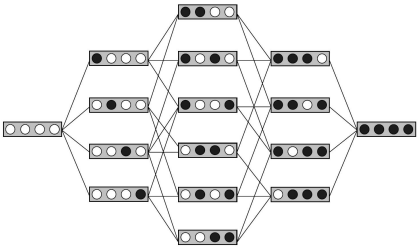


Figura 5.1 – Subconjuntos possíveis para quatro atributos.

Desse modo, qualquer método de seleção de atributos pode ser caracterizado por sua instanciação em relação a quatro questões básicas, as quais determinam a natureza do processo de busca heurística (BLUM; LANGLEY, 1997):

- **ponto de partida:** dependendo do ponto de partida escolhido, a direção de busca irá variar. Quando a busca tem seu estado inicial mais à esquerda, o conjunto vazio de atributos, representado pelos quatro círculos em brancos, ela é conhecida como busca "para frente" (*forward selection*). Já a situação que inicia com o subconjunto contendo todos os atributos e vai, sucessivamente, removendo-os, é denominada de eliminação (ou sentido da busca) "para trás" (*backward selection*). Também podem ser empregadas variações de ambas as técnicas, selecionando-se um estado inicial em algum ponto do espaço de busca e movendo-se a partir desse ponto (*outward selection*);

- **organização da busca:** a cada ponto na busca, modificações locais no conjunto de atributos são consideradas; uma dessas é selecionada e uma nova iteração é realizada. Claramente, se o número de características é muito grande, a busca exaustiva de todos os subespaços é proibitiva, já que existem (2^n) subconjuntos diferentes para n características

(considerando também o subconjunto vazio, sem nenhuma característica). Uma abordagem mais prática é a utilização de uma busca heurística, porém não garante uma solução ótima;

- **critério de avaliação:** a estratégia utilizada na avaliação dos subconjuntos de características é um problema importante. Uma métrica normalmente empregada envolve a habilidade de um atributo discriminar as classes de um conjunto de treinamento. Para classificação, o melhor subconjunto é aquele que provê a maior separação dos dados. A separação de dados é normalmente calculada por algum critério de medida da distância entre as classes existentes. Define-se precisão de classificação como sendo a porcentagem de exemplos classificados corretamente para um determinado conjunto de teste;

- **critério de parada:** durante o processo de avaliação, deseja-se que o algoritmo pare quando é observado que já não existem melhorias na precisão do classificador.

5.3.3 Métodos de seleção em filtro

Técnicas de seleção de atributos em filtro introduzem um processo separado, o qual ocorre antes da aplicação do algoritmo de aprendizado. O modelo foi batizado de filtro por (JOHN; KOHAVI; PFLEGER, 1994) pelo fato de “filtrar” os atributos irrelevantes, segundo algum critério, antes que uma indução ocorra, ou seja, as características são selecionadas antes que o algoritmo de aprendizado seja executado. Esse passo de pré-processamento considera características gerais do conjunto de exemplos para selecionar alguns atributos e excluir outros. Sendo assim, métodos de filtros são independentes do algoritmo de classificação que, simplesmente,

receberá como entrada o conjunto de exemplos contendo apenas os atributos selecionados pelo filtro. A vantagem do modelo em filtro está no fato de que o mesmo não precisa ser reaplicado para cada execução do algoritmo de treinamento. Assim, o modelo em filtro é eficiente ao se abordar problemas que possuam um espaço de características muito elevado.

Qualquer algoritmo que efetue algum tipo de seleção pode ser usado para filtrar atributos. A saída do algoritmo de filtragem é o conjunto de atributos por ele selecionados. Os atributos restantes são removidos do conjunto de exemplos, reduzindo assim sua dimensão. Após isso, o conjunto de exemplos reduzido pode ser usado por qualquer classificador. Contudo, sabe-se, que na maioria das vezes, o subconjunto ótimo de características depende do algoritmo de treinamento a ele associado. Portanto, embora a abordagem apresente-se computacionalmente eficiente, ela encontra apenas soluções sub-ótimas genéricas que independem do classificador. Assim, um subconjunto de características selecionado usando um método em filtro pode resultar em uma alta precisão para determinado classificador e em baixa precisão em outros. O maior problema destes métodos é que cada coeficiente é computado utilizando somente informações do atributo relacionado, não levando em conta a existência de uma interdependência, ou mútua informação, entre as características. De fato, podem existir atributos complementares que individualmente não têm uma relevância, mas que combinados podem ter um papel importante no processo de discriminação. Dentre os métodos de filtro, pode-se citar o Golub, o Fisher e o Relief[1].

[1] Relief é um algoritmo de seleção de características utilizados na classificação binária, generalizável para a classificação polinomial por decomposição em uma série de problemas binários, proposto por Kira e Rendell (1992).

5.3.4 Métodos de seleção embutidos

A estratégia dos métodos embutidos, também chamada de seleção em cápsula, se baseia no fato de que alguns indutores são capazes de realizar sua própria seleção de atributos. Eles fazem uso do algoritmo de indução para estimar o valor do subconjunto de características selecionado durante a fase de treinamento e são geralmente específicos para um dado algoritmo de classificação. A idéia central desses métodos se baseia na otimização direta de uma função objetivo composta geralmente de dois termos. O primeiro, relacionado a uma medida do desempenho do classificador, que deve ser maximizada, e o segundo, uma medida de regularização relacionada à quantidade de variáveis, que deve ser minimizada. Dentre os métodos embutidos, pode-se citar o AROM e o NSC. Outro exemplo de método embutido é fazer a utilização do algoritmo IMA$_p$ na sua formulação L_∞, que minimiza a norma L_1 do vetor w. Nessa versão, ele seleciona as características pelo valor da maior componente do vetor, minimizando as componentes não importantes, sendo essas, assim, facilmente eliminadas.

5.3.5 Métodos de seleção *wrapper*

Em contraste com filtros, a abordagem *wrapper*, popularizada por (KOHAVI; JOHN, 1997), gera vários subconjuntos de atributos como candidatos, executa o indutor individualmente em cada subconjunto e usa a precisão do classificador para avaliar o subconjunto em questão. O processo é repetido até que um critério de parada seja satisfeito. A idéia geral dessa abordagem é que o algoritmo de seleção de características existe como um *wrapper* ao redor do indutor e é responsável por conduzir a busca por um bom subconjunto de atributos. A qualidade de um subconjunto candidato é avaliada pelo próprio indutor como uma “caixa preta”. O objetivo

da busca é encontrar o subconjunto (nó) com a melhor qualidade, utilizando alguma função para guiá-la.

Em geral, a busca é conduzida no espaço do subconjunto de atributos, ou de candidatos. Como estratégia pode-se empregar algoritmos míopes (*greedy*) ou buscas direcionadas (*best-first*) e com direções *forward* ou *backward*. A precisão dos subconjuntos pode ser estimada por uma validação cruzada (*cross-validation*). A principal vantagem do modelo é a dependência entre os algoritmos de seleção e de aprendizado. Por outro lado, essa abordagem pode ser computacionalmente dispendiosa, uma vez que o indutor deve ser executado para cada subconjunto de atributos considerado. Dentre os métodos, pode-se citar o RFE e o AOS.

5.4 Admissible Ordered Search

Técnicas de seleção de características comumente utilizadas com base em análises de variância dos dados, bem como métodos baseados na eliminação recursiva de características, nem sempre encontram classificadores com uma menor quantidade de atributos ou um melhor poder de generalização. Nesse contexto, foi introduzido um novo algoritmo de seleção de características (VILLELA et al., 2011) denominado Admissible Ordered Search (AOS), o qual se baseia na realização de uma busca ordenada admissível, semelhante à empregada pelo algoritmo A* (HART; NILSSON; RAPHAEL, 1968). Esse algoritmo tem a capacidade de encontrar, em cada dimensão do problema, o classificador de maior margem.

Em um processo de busca ordenada, garante-se a admissibilidade do algoritmo se a função de avaliação é monótona. Para problemas de minimização, a função precisa ser monótona crescente, e, para problemas de maximização, ela deve ser monótona decrescente. Nesse sentido, uma vez que se está à procura da maximização da margem, usa-se como função de avaliação o valor

da margem obtido a partir de cada hipótese após a solução do problema de classificação no conjunto de características selecionado por essa hipótese. A admissibilidade do processo é garantida, uma vez que os valores da margem são sempre decrescentes quando a dimensão diminui. Ou seja:

$$\gamma_{g_j}^{d-1} \leq \gamma_g^d, \forall j$$

A estratégia de controle dessa busca ordenada é implementada com a inserção das hipóteses candidatas em uma fila ordenada pelos valores das margens. Uma vez que a ordem das características selecionadas não importa, poderá haver alguma redundância. A fim de evitar esse problema, cria-se uma tabela *hash*, e, para cada nova hipótese, verifica-se a sua unicidade nessa tabela antes de inseri-la na fila. Usando a margem geométrica real exigiria a solução de um problema para a maximização da margem para cada hipótese gerada. Em vez disso, utiliza-se uma estimativa otimista desta margem, que é a sua margem projetada. Esse valor será um limite máximo para a margem geométrica real da mesma hipótese no espaço associado. Com isso, mantém-se a admissibilidade do processo, uma vez que:

$$\gamma_{p_j}^{d-1} \geq \gamma_{g_j}^{d-1}, \forall j$$
$$\gamma_{p_j}^{d-1} \leq \gamma_g^d, \forall j$$

Para cada iteração do algoritmo, a hipótese relacionada com a maior margem é escolhida, independentemente da sua dimensão, para ser expandida e gerar novas hipóteses num espaço de dimensão menor. Dessa forma, pode-se verificar duas situações possíveis:

- para os casos em que o valor da margem para a hipótese escolhida é o valor projetado, pode-se calcular o seu valor real através da solução de um problema de maximização da

margem e compará-lo com o maior valor da fila de prioridade. Se ainda é a melhor opção, fecha-se esse estado e geram-se as suas hipóteses. Caso contrário, substitui-se o valor da margem projetada dessa hipótese pelo valor real e reinsere o mesmo na fila;

- para os casos em que o valor da margem da hipótese escolhida já é o valor real, fecha-se o estado e geram-se suas hipóteses.

Para amenizar a explosão combinatória, o algoritmo desenvolve dois esquemas de podas adaptativas, sendo um baseado na atualização constante de um limite de margem inferior. Esse limite é computado toda vez que uma hipótese escolhida for a primeira a chegar em uma dada dimensão. Para tanto, é utilizada uma estratégia de seleção míope que avalia os valores de margem até uma dimensão inferior escolhida, eliminando, assim, estados de maior dimensão que tenham valores inferiores a esse limite. A outra poda, também realizada quando a hipótese escolhida é a primeira a chegar a uma dada dimensão, é baseada em um corte que elimina todos os candidatos presentes na fila que tenham dimensão superior à hipótese em questão, além do valor do corte. Com isso, são eliminadas hipóteses que já estão há algum tempo na fila e ainda não foram selecionadas, ou seja, hipóteses que tendem a ter valores de margem relativamente baixos. Também, nesse momento, uma estimativa do erro esperado pode ser calculada e utilizada como critério de parada do algoritmo, uma vez que representa o desempenho de generalização do classificador.

5.4.1 Ramificação

Como primeiro parâmetro de ramificação, adota-se a eliminação das componentes relacionadas aos menores valores de

margens projetadas, ou as menores componentes do vetor w, tal como utilizado no RFE. Esse critério está relacionado à escolha da menor variação no valor da função objetivo do problema SVM (Support Vector Machine) de minimização em sua formulação dual:

$$J = \frac{1}{2} \cdot \alpha^T \cdot H \cdot \alpha - \alpha^T \cdot 1, \tag{5.3}$$

sendo α o vetor de multiplicadores e H a matriz definida a partir da matriz K com componentes na forma $y_i \cdot y_j \cdot K_{i,j}$. Assim, sustentando o mesmo vetor de multiplicadores, se for retirada a i-ésima variável, tem-se a variação:

$$\begin{aligned} \Delta J(i) &= J - J(i) \\ &= \tfrac{1}{2} \cdot \alpha^T \cdot H \cdot \alpha - \tfrac{1}{2} \cdot \alpha^T \cdot H(i) \cdot \alpha \\ &= \tfrac{1}{2} \cdot \alpha^T \cdot (H - H(i)) \cdot \alpha \end{aligned} \tag{5.4}$$

Dessa forma, escolhe-se como variável a ser retirada aquela que produzir a menor variação no valor da função, ou seja:

$$k = \text{Arg Min}_i \Delta J(i) \tag{5.5}$$

Para o caso especial de SVM linear na formulação dual, tem-se o valor de cada componente da matriz *kernel* K definido como o produto interno dos vetores respectivos, ou seja:

$$K_{i,j} = \langle x_i, x_j \rangle, \tag{5.6}$$

reduzindo o cálculo de cada componente da matriz diferença $H - H(i)$ ao simples produto das componentes associadas a variável que está sendo retirada, ou seja $(x_i)_k \cdot (x_j)_k$.

Na formulação primal, pode-se associar a variação do valor da função à variação da norma do vetor w, ou seja:

$$\begin{aligned} \Delta J(i) &= J - J(i) \\ &= \tfrac{1}{2} \cdot \alpha^T \cdot H \cdot \alpha - \tfrac{1}{2} \cdot \alpha^T \cdot H(i) \cdot \alpha \\ &= ||w||_2 - ||w_i||_2 \end{aligned} \tag{5.7}$$

O que se torna equivalente, em termos de critério, à escolha da variável associada a componente de menor magnitude do vetor. Ou seja:

$$k = \text{Arg Min}_i (w_i)^2 \tag{5.8}$$

Esse critério de escolha é adotado pelo algoritmo RFE quando combinado com o SVM linear na formulação primal.

Como segundo critério de ramificação utiliza-se uma medida relacionada à minimização da expectativa de um limite superior associado ao erro esperado do processo de classificação. Segundo (VAPNIK, 1995), esse limite pode ser expresso por:

$$\frac{1}{\ell} \cdot E \left\{ \frac{R^2}{\gamma^2} \right\}, \tag{5.9}$$

onde ℓ se refere à quantidade de amostras, R se refere ao tamanho dos dados relacionado ao raio da maior esfera que engloba os mesmos e γ se refere à margem do classificador. Ou seja, o desempenho dos diferentes candidatos em termos de poder de generalização depende não somente de uma larga margem, mas também de uma redução no tamanho dos dados.

Portanto, se o objetivo é minimizar a relação $\frac{R^2}{\gamma^2}$ ou, de forma equivalente, minimizar o valor da expressão $R^2 \cdot ||w||_2^2 = R^2 \cdot \sum_i (w_i)^2$, deve-se retirar a variável, ou característica, que menos altera a diferença desse valor, no sentido de preservar aquelas que mais alteram sua diferença. Ou seja:

$$k = \text{Arg Min}_i (R_i)^2 \cdot (w_i)^2 \tag{5.10}$$

Como terceiro critério de ramificação utiliza-se uma medida relacionada à maximização da distância entre os centros das duas classes. Isto é, o desempenho dos diferentes candidatos em termos de poder de generalização depende não somente de uma larga margem, mas também de uma mínima redução na distância entre os centros das classes.

Portanto, se o objetivo é minimizar o valor da expressão $\left(\frac{1}{C}\right)^2 \cdot \sum_i (w_i)^2$, na qual C é a distância entre os centros, deve-se retirar a variável que menos altera a diferença desse valor. Logo:

$$k = \text{Arg Min}_i \frac{(w_i)^2}{(C_i)^2} \tag{5.11}$$

Como quarto critério, utiliza-se a junção dos dois últimos; uma medida relacionada à minimização do raio juntamente com a maximização da distância entre os centros. Portanto, o desempenho dos diferentes candidatos em termos de poder de generalização depende de uma mínima redução na margem e na distância entre os centros, além de uma maior redução no tamanho dos dados. Ou seja:

$$k = \text{Arg Min}_i \frac{(R_i)^2 \cdot (w_i)^2}{(C_i)^2} \tag{5.12}$$

Adicionalmente, é utilizado um critério associando o conceito dos métodos em filtro. Para isso, utilizou-se a pontuação do método Golub. Assim, o desempenho dos diferentes candidatos em termos de poder de generalização depende, além de uma larga margem, de um mínimo *score* G. Ou seja:

$$k = \text{Arg Min}_i \left(\frac{|\mu_{1_i} - \mu_{2_i}|}{(\sigma_{1_i} + \sigma_{2_i})} \right)^2 \cdot (w_i)^2 \text{ ou } (G_i)^2 \cdot (w_i)^2 \tag{5.13}$$

5.4.2 Medidas de avaliação

Além da medida de avaliação padrão, o valor da margem, foi adotada outra medida, sendo esta a distância entre os centros das classes, que também é uma medida monótona decrescente, uma vez que a distância em uma dimensão $d-1$ é sempre menor ou igual à distância na dimensão d.

Como, para o caso de um classificador SVM, deve haver o comprometimento com a margem, essa medida foi utilizada em conjunto com o valor da margem. Então, como segunda medida de

avaliação, tem-se $\gamma \cdot C$, sendo C a distância entre os centros das duas classes.

5.4.3 Solução inicial

Como foi visto, a escolha da variável associada à componente de menor magnitude do vetor w está relacionada à escolha da margem projetada de maior valor. De fato, se for escolhida uma variável associada a uma componente de valor zero, tem-se o valor da margem projetada igual ao valor da margem máxima real no subespaço associado. Isso facilita bastante o processo de solução dos problemas de classificação com SVM, pois, nesse caso, não há necessidade de se resolver o novo problema de classificação, uma vez que as soluções são as mesmas.

Quando o valor da componente de menor magnitude for muito baixo, tem-se a solução dos dois problemas muito próximas. Nesse caso, utiliza-se a última solução obtida no espaço em questão como solução inicial do problema de classificação do subespaço inferior associado. Se a formulação for primal, elimina-se a componente associada do vetor w. Entretanto, caso a formulação seja dual utiliza-se o mesmo vetor de multiplicadores α.

Para o cálculo da margem projetada na formulação dual utiliza-se a mesma relação de normas do vetor w computados a partir de sua expansão em função do vetor de multiplicadores α. Ou seja:

$$\gamma_{p_j}^{d-1} = \left(\frac{1}{||w||_2}\right) \cdot \gamma_g^d \cdot ||w_j||_2 = \left(\frac{1}{\alpha^T \cdot H \cdot \alpha}\right) \cdot \gamma_g^d \cdot \alpha^T \cdot H\,(j) \cdot \alpha$$

O Algoritmo 6 descreve o pseudocódigo do algoritmo AOS.

Algoritmo 6: Admissible Ordered Search

Entrada: conjunto de treinamento Z;
conjunto de características $F = \{1, 2, 3, \ldots, d\}$;
fator de ramificação b;
profundidade da poda p;
profundidade do corte c;
nível de parada s;
Dados: nível alcançado $nivel$;
limite inferior usado para podar a árvore de busca $limite$;
Saída: último estado aberto;

início
  inicializar o heap H e a tabela hash HT;
  computar a solução SVM para o estado inicial $S_{inicial}$ com o conjunto F;
  $nivel \leftarrow d$;
  inserir $S_{inicial}$ em H;
  enquanto $nivel > s$ **e** H *não vazio* **faça**
    selecionar a melhor hipótese S de H;
    se S *tiver somente uma margem projetada* **então**
      computar a margem real de S usando SVM;
      se *solução SVM convergiu* **então**
        reinserir S em H;
      fim se
    senão
      se *dimensão do estado* S *for igual a* $nivel$ **então**
        usar o RFE até $nivel - p$ e encontrar o novo valor para $limite$;
        eliminar de H todos os estados com margem menor que $limite$;
        eliminar de H todos os estados com nível maior que $nivel + c$;
        $nivel \leftarrow nivel - 1$;
      fim se
      ordenar F em S pelo critério de ramificação;
      para i **de** *1* **até** b **faça**
        criar um novo estado S' com o conjunto $F' = F - \{f_i\}$;
        computar γ_{p_j} para S'; `// onde` γ_{p_j} `é a margem projetada`
        se $\gamma_{p_j} > limite$ **e** F' *não está em* HT **então**
          inserir F' em HT;
          inserir S' em H;
        fim se
      fim para
    fim se
  fim enquanto
  retorna *último estado aberto;*
fim

5.5 Experimentos

5.5.1 Conjunto de dados

Para a análise dos resultados, foram utilizadas dezesseis bases de dados. Onze bases linearmente separáveis, sendo duas sintéticas, cinco de *microarrays*, uma gerada artificialmente e outras três "comuns", e cinco não linearmente separáveis. Com exceção da base artificial, as bases utilizadas no trabalho estão contidas no repositório de aprendizado de máquina da UCI (ASUNCION; NEWMAN, 2007), ou referenciadas por Golub et al. (1999), Alon et al. (1999) ou Singh et al. (2002).

A Tabela 5.1 mostra um resumo das informações das bases de dados, com as quantidades de características (atributos) e amostras (instâncias) de cada uma.

Tabela 5.1 – Informações das bases de dados.

Base	Atributos	Amostras		
		Pos.	Neg.	Total
Iris	4	50	100	150
Mushroom	98	3488	2156	5644
Sonar	60	97	111	208
Synthetic	60	300	300	600
Robot LP4	90	24	93	117
Colon	2000	22	40	62
Leukemia	7129	47	25	72
Prostate	12600	50	52	102
Breast	12625	10	14	24
DLBCL	5468	58	19	77
Artificial	20	800	800	1600
Ionosphere	34	225	126	351
WDBC	30	212	357	569
Bupa	6	145	200	345
Pima	8	268	500	768
Wine	13	107	71	178

5.5.2 *K-fold cross validation*

A ideia principal do método k-*fold cross validation* (validação cruzada) é dividir o conjunto de treinamento M de tamanho n em k subconjuntos. Cada subconjunto M_i, para $1 \leq i \leq k$, será do mesmo tamanho (n/k). Antes da divisão do conjunto, todas as instâncias são embaralhadas. Então são gerados k modelos. O $i-$ésimo modelo é construído com o conjunto de treinamento $M - M_i$, que é a diferença entre o subconjunto de entrada e o subconjunto M_i. O modelo é então testado com o conjunto M_i. Isso se repete para cada i, $1 \leq i \leq k$. Por fim é calculada a média do erro para os k modelos, determinada pelos resultados obtidos com os testes.

Com essa metodologia, não é necessária a separação das instâncias em conjuntos de treinamento e testes antecipadamente. Todos os dados utilizados na geração são utilizados em ambos os papéis: treinamento e testes. A idéia do método foi proposta por Mosteller e Tukey (1968) e revisada por Geisser (1975) e Wahba e Wold (1975). Uma revisão dos métodos de validação cruzada é apresentada por Browne (2000).

Em uma derivação desse método (KOHAVI, 1995), é utilizada a estratificação visando obter melhores resultados. Com a estratificação, é mantida a mesma proporcionalidade entre as classes em cada um dos k conjuntos, isso é, antes de dividir o conjunto de treinamento em k partes de tamanho n/k é verificada a porcentagem de instâncias de cada classe no conjunto de instâncias disponíveis. Então, quando o conjunto de treinamento é dividido, a porcentagem de instâncias de cada classe é mantida, e cada subconjunto conterá a mesma proporcionalidade entre as classes. (KOHAVI, 1995) e (MCLACHLAN; DO; AMBROISE, 2004) sugerem um valor para k igual a 10.

Nos testes realizados, os conjuntos de dados foram divididos em $2/3$ para o processo de seleção de características e, consequentemente, para os treinamentos, e $1/3$ para o processo de teste (validação dos resultados). Para a validação dos conjuntos de treinamento, foi utilizado o valor do erro médio de 10 execuções de um 10-*fold cross validation*. Para comparações mais precisas, foram selecionados, para uma mesma base, sempre os mesmos conjuntos de treinamento e teste e sempre os mesmos 10 conjuntos para as validações cruzadas, preservando as sementes geradoras de aleatoriedade.

5.5.3 *Kernel*

Para a escolha do *kernel* que seria utilizado para as bases não linearmente separáveis, foram realizadas variações nos *kernel* polinomial e gaussiano (descritos a seguir), e nos seus parâmetros, e foram verificados os erros médios de 10 execuções de um 10-*fold* no conjunto de treinamento e o erro da validação no conjunto de teste. Para o *kernel* polinomial foram utilizados os graus, d, iguais a 2 e 3. Já para o gaussiano, foram utilizados valores de σ iguais a 0,01, 0,1, 1, 10 e 100. Os resultados foram obtidos através da execução do algoritmo AOS com as parametrizações "padrões", ou seja, com ramificação completa e sem nenhum tipo de poda. A Tabela 5.2 mostra esses resultados. As definições dos *kernel* polinomial e gaussiano são, respectivamente:

$$K\left(x_i, x_j\right) = \left(\langle x_i \cdot x_j \rangle + 1\right)^d \tag{5.14}$$

$$K\left(x_i, x_j\right) = \exp\left(-\frac{||x_i - x_j||^2}{2\sigma^2}\right) \tag{5.15}$$

De posse dos resultados, o *kernel* escolhido para a realização dos experimentos foi o gaussiano e o valor do parâmetro σ igual a 1.

Tabela 5.2 – Escolha do *kernel*.

Base	*kernel*	Parâmetro	Atributos	10-*fold*	Valid.
Ionosphere	Polinomial	2	6	8,55%	10,26%
		3	5	9,38%	12,82%
	Gaussiano	0,01	3	22,26%	17,95%
		0,1	3	19,67%	15,38%
		1	3	10,74%	10,26%
		10	5	8,12%	12,82%
		100	7	11,54%	9,40%
WDBC	Polinomial	2	não converge	–	–
		3	3	48,31%	36,32%
	Gaussiano	0,01	2	37,20%	37,37%
		0,1	2	36,68%	36,32%
		1	2	37,20%	37,37%
		10	2	37,47%	37,37%
		100	2	32,99%	36,32%
Bupa	Polinomial	2	não converge	–	–
		3	não converge	–	–
	Gaussiano	0,01	3	40,87%	40,87%
		0,1	3	40,87%	40,00%
		1	3	41,30%	40,87%
		10	3	37,39%	42,61%
		100	3	36,09%	41,74%
Pima	Polinomial	2	não converge	–	–
		3	não converge	–	–
	Gaussiano	0,01	3	34,77%	35,16%
		0,1	3	34,77%	35,16%
		1	3	34,57%	36,33%
		10	3	34,97%	42,58%
		100	3	36,13%	28,13%
Wine	Polinomial	2	não converge	–	–
		3	3	56,44%	57,63%
	Gaussiano	0,01	2	40,30%	38,98%
		0,1	2	38,45%	35,59%
		1	2	35,01%	22,03%
		10	2	29,75%	22,03%
		100	2	22,20%	18,64%

5.5.4 Fator de ramificação

Para a escolha dos parâmetros para o fator de ramificação e para as podas de profundidade e de corte, foram escolhidas seis bases e foi utilizado o RFE até a dimensão 20, para tornar a busca em todo o espaço possível. Com isso, tem-se 2^{20} possibilidades de solução para cada base. Excluindo-se o estado inicial, que não precisa entrar na fila, e o estado com nenhuma característica, tem-se 1.048.574 estados possíveis para bases com 20 atributos. Foram

selecionadas duas bases sintéticas (Synthetic e Robot LP4), duas de *microarray* (Prostate e Colon) e duas não lineares (Ionosphere e Sonar – que é uma base linear com todas as características, mas não com somente 20). Para efeito de comparação, todos os valores de margens estão na norma L_2, ou seja, pelo valor da distância euclidiana.

Para a escolha de um fator de ramificação que gerasse boas soluções e fosse computacionalmente possível, foram realizados testes com os valores 2, 3, 5 e 20 (busca completa), além da execução do RFE. Foram comparadas as quantidades de candidatos inseridos (I), reinseridos (RI) (estados com margem projetada que foram treinados e não tinham, de fato, uma maior margem) e expandidos (E), o tamanho máximo da fila de estados (M) e a quantidade de nós que não precisou de treinamento (NT), além da dimensão final (D), com seu valor de margem (γ). Foram realizados testes nas versões IMA$_\infty$, que minimiza a norma L_1 do vetor, e IMA$_2$, que minimiza a própria norma L_2. As Tabelas 5.3, 5.4 e 5.5 mostram essas variações no fator de ramificação.

Tabela 5.3 – Fator de ramificação para as bases sintéticas.

Base	L_q	b	I	RI	E	M	NT	D	γ
Synthetic	L_1	RFE	–	–	–	–	–	6	0,136
		2	1191	753	866	271	103	6	0,563
		3	10134	6884	6992	2599	1081	6	0,571
		5	127976	91577	70693	44962	19126	6	0,571
		20	690518	418068	263024	299125	256779	6	0,571
	L_2	RFE	–	–	–	–	–	7	0,819
		2	1342	995	976	297	142	6	0,514
		3	15031	11044	9875	3850	1676	6	0,571
		5	141063	97538	76825	42648	30621	6	0,571
		20	690922	429210	270130	279325	257890	6	0,571
LP4 Robot	L_1	RFE	–	–	–	–	–	7	0,458
		2	253	186	43	189	27	5	0,585
		3	3182	2187	601	2069	826	4	0,205
		5	36199	22671	7973	20453	12589	4	0,205
		20	864961	446326	348586	407498	393180	4	0,205
	L_2	RFE	–	–	–	–	–	7	0,342
		2	275	182	50	206	25	5	0,583
		3	3775	2582	750	2589	754	4	0,205
		5	45466	29706	11040	27765	12532	4	0,205
		20	877821	470892	313104	426118	406856	4	0,205

Tabela 5.4 – Fator de ramificação para as bases de *microarrays*.

Base	L_q	b	I	RI	E	M	NT	D	γ
Leukemia	L_1	RFE	–	–	–	–	–	4	23,306
		2	263	138	204	47	47	4	23,306
		3	2337	1375	1556	520	626	4	23,306
		5	41618	25128	23831	10582	14336	4	23,306
		20	928709	568739	485215	349554	354374	4	23,306
	L_2	RFE	–	–	–	–	–	6	72,498
		2	673	524	536	124	116	4	23,306
		3	7335	5221	5123	1388	1907	4	23,306
		5	80067	53283	48081	17665	25782	4	23,306
		20	938122	575796	505402	320345	362307	4	23,306
Prostate	L_1	RFE	–	–	–	–	–	7	62,346
		2	250	170	192	50	18	5	9,043
		3	2653	1698	1608	596	630	5	16,683
		5	50670	30528	25858	13324	17080	5	16,683
		20	756809	457765	323150	304999	293076	5	16,683
	L_2	RFE	–	–	–	–	–	7	23,108
		2	409	338	306	73	46	6	26,736
		3	5045	3679	3164	1077	1197	5	16,683
		5	67281	43557	34668	16238	22956	5	16,683
		20	774370	497827	345780	285740	306542	5	16,683

Tabela 5.5 – Fator de ramificação para as bases não lineares.

Base	b	I	RI	E	M	NT	D	γ
Sonar	RFE	–	–	–	–	–	4	0,002
	2	953	796	753	196	145	4	0,003
	3	9862	8221	7684	2201	1601	4	0,003
	5	116011	105616	99616	26892	10163	4	0,003
	20	1045569	1033415	1023637	316622	12154	4	0,003
Ionosphere	RFE	–	–	–	–	–	4	0,006
	2	717	668	614	102	39	4	0,009
	3	7623	6936	5700	1420	641	4	0,012
	5	98717	91631	75947	23413	6828	4	0,012
	20	1043881	1014712	986521	343921	28715	4	0,012

É possível visualizar que a norma L_1, para as bases linearmente separáveis, é de fato melhor para seleção de características. Para ramificações pequenas, ou para o RFE, ela alcança soluções melhores, ou pelo menos iguais. Já para ramificações grandes, aonde a busca é mais ampla, ela encontra as mesmas soluções da norma L_2, porém de forma mais rápida. É possível visualizar, também, que não é necessário mais do que a geração de 3 hipóteses por cada estado escolhido, para se ter as mesmas soluções da busca exaustiva. Não é possível garantir que, para bases com mais

de 20 dimensões, o valor 3 será suficiente para encontrar os melhores resultados, mas é possível prever que sim, uma vez que existe grande possibilidade de, para bases maiores, serem inseridos mais do que um estado com dimensão 20 na fila de abertos, aumentando, assim, o espaço de busca. Com isso, foram escolhidos a norma L_1 para as bases lineares, e o valor 3 para o fator de ramificação b para todas as bases.

5.5.5 Podas

Para as escolhas das podas, para tornar o processo mais rápido, foram realizados testes com os valores 2, 3 e 5 para cada tipo de poda, além de combinações dos valores 2 e 3 para os dois tipos juntos. As Tabelas 5.6, 5.7 e 5.8 mostram as variações das podas de profundidade e de corte. Foram comparadas as quantidades de candidatos inseridos, reinseridos e expandidos, o tamanho máximo da fila, a quantidade de nós não treinados e a quantidade de nós que foram podados da fila (Podas), além da dimensão final, com seu valor de margem.

Como foi utilizado um fator de ramificação de valor 3, uma poda de profundidade alta pode causar uma eliminação excessiva ou inserções desnecessárias. Isso porque fazer uma grande descida míope pode abrir um caminho que não seria escolhido pelos candidatos “normais” ou então a atualização do valor limite pode eliminar candidatos que gerariam boas soluções. Da mesma forma, escolher uma baixa poda de corte, deixando somente candidatos de poucas dimensões, pode ocasionar cortes demais e fazer com que alguns candidatos “bons” sejam eliminados da fila.

Os dois parâmetros combinados geraram sempre os mesmos resultados das execuções sem podas. A combinação que obteve as menores quantidades de estados gerados foi a com valor 2 para ambos os parâmetros. Por esse motivo, foi escolhido uma

Tabela 5.6 – Podas para as bases sintéticas.

Base	p	c	I	RI	E	M	NT	Podas	D	γ
Synthetic	–	–	10134	6884	6992	2599	1081	0	6	0,571
	2	–	483	308	267	149	106	16	6	0,571
	3	–	1990	1383	1291	524	274	30	6	0,571
	5	–	2996	2054	1944	780	419	58	6	0,571
	–	2	214	139	92	66	58	33	6	0,571
	–	3	502	348	266	165	108	16	6	0,571
	–	5	679	457	373	220	144	10	6	0,571
	2	2	217	125	103	70	67	16	6	0,571
	2	3	437	290	235	127	104	20	6	0,571
	3	2	175	91	81	52	59	14	6	0,571
	3	3	370	232	198	101	95	23	6	0,571
Robot LP4	–	–	3182	2187	2069	601	826	0	4	0,205
	2	–	1833	1187	1172	351	425	18	4	0,205
	3	–	2123	1418	1374	409	442	74	4	0,205
	5	–	63	34	22	13	13	27	5	0,583
	–	2	191	116	88	37	61	36	4	0,205
	–	3	320	190	162	59	109	40	4	0,205
	–	5	590	356	345	105	185	21	4	0,205
	2	2	179	96	81	36	59	35	4	0,205
	2	3	299	166	158	60	101	31	4	0,205
	3	2	186	102	85	37	58	38	4	0,205
	3	3	257	145	134	51	82	27	4	0,205

poda de profundidade p igual a 2, assim como uma poda de corte c também igual a 2.

5.5.6 Critérios de ramificação e medidas de avaliação

Para as escolhas do critério de ramificação e da medida de avaliação dos candidatos, foram realizados testes com as medidas w (1), $w \cdot R$ (2), w/C (3), $w \cdot R/C$ (4), e $w \cdot G$ (5) para as variações do critério de ramificação (r) e as medidas γ e $\gamma \cdot C$ para as variações da medida, ou função, de avaliação (f). Foram realizadas as 10 combinações possíveis dessas medidas e testadas em quatro bases de dados, sendo três lineares (Robot LP4, Mushroom e Prostate) e uma não linear (Ionosphere). Foram comparadas as quantidades de candidatos inseridos, reinseridos, expandidos, não treinados, não inseridos (NI) e podados, além da dimensão final, com seu valor de margem, e o erro médio de 10 execuções de um

Tabela 5.7 – Podas para as bases de *microarrays*.

Base	p	c	I	RI	E	M	NT	Podas	D	γ
Colon	–	–	2337	1375	1556	520	626	0	4	23,306
	2	–	2456	1423	1651	543	691	40	4	23,306
	3	–	1810	1083	1171	363	458	94	4	23,306
	5	–	1771	1051	1152	346	472	78	4	23,306
	–	2	269	165	133	49	71	63	4	23,306
	–	3	654	368	363	140	228	43	4	23,306
	–	5	2233	1315	1446	500	618	9	4	23,306
	2	2	298	159	141	63	92	63	4	23,306
	2	3	511	269	286	103	177	43	4	23,306
	3	2	164	72	74	35	57	39	4	23,306
	3	3	532	296	298	122	177	35	4	23,306
Prostate	–	–	2653	1698	1608	596	630	0	5	16,683
	2	–	2382	1257	1413	417	851	19	5	16,683
	3	–	2307	1434	1410	505	559	1	5	16,683
	5	–	2827	1790	1732	672	652	0	5	16,683
	–	2	211	144	98	53	49	43	6	32,330
	–	3	666	359	343	126	251	59	5	16,683
	–	5	2075	1127	1242	383	726	17	5	16,683
	2	2	215	135	100	54	45	46	5	16,683
	2	3	845	454	444	161	301	87	5	16,683
	3	2	139	81	59	37	28	35	5	16,683
	3	3	540	352	284	135	116	40	5	16,683

10-*fold*. As Tabelas 5.9 e 5.10 mostram as variações do fator de ramificação.

É possível constatar que a utilização da função que utiliza a distância dos centros só é válida quando combinada a critérios de ramificação que também utilizam essa medida. Isso se deve ao fato de que, para os demais critérios, as hipóteses geradas não têm um comprometimento com ambas as medidas, podendo ter, assim, um dos valores muito baixo e não sendo nem inseridos na fila. Porém, mesmo essa função sendo interessante associada com alguns critérios, ela gerou hipóteses demais e, ainda assim, não foi superior do que a medida que utiliza somente o valor da margem como seleção.

Isso porque, para uma solução SVM, é mais importante uma maior distância entre amostras de classes opostas mais próximas do que entre as médias das classes. Talvez a utilização de

Tabela 5.8 – Podas para as bases não lineares.

Base	p	c	I	RI	E	M	NT	Podas	D	γ
Sonar	–	–	9862	8221	7684	2201	1601	0	4	0,003
	2	–	11125	9232	8684	2457	1816	0	4	0,003
	3	–	14367	12042	8619	3166	2228	0	4	0,003
	5	–	17453	14836	11169	3997	2493	0	4	0,003
	–	2	6650	5487	13627	1545	1141	256	4	0,003
	–	3	9412	8208	4756	2118	1541	8	4	0,003
	–	5	9862	8221	7251	2201	1601	0	4	0,003
	2	2	7311	6094	7684	1584	1153	312	4	0,003
	2	3	10886	9239	5292	2336	1575	93	4	0,003
	3	2	7379	6126	8482	1662	1183	316	4	0,003
	3	3	11815	9994	5380	2519	1732	105	4	0,003
Ionosphere	–	–	7623	6936	5700	1420	641	0	4	0,012
	2	–	9548	8765	7273	1707	720	12	4	0,012
	3	–	11435	10436	8591	2109	941	55	4	0,012
	5	–	11694	10649	8786	2164	972	48	4	0,012
	–	2	1075	934	578	301	140	86	4	0,012
	–	3	2760	2410	1711	699	346	92	4	0,012
	–	5	6418	5831	4635	1313	561	26	4	0,012
	2	2	1352	1130	791	353	194	60	4	0,012
	2	3	2114	1804	1330	540	284	29	4	0,012
	3	2	1370	1140	800	352	192	65	4	0,012
	3	3	2154	1837	1389	530	281	29	4	0,012

uma medida que pesasse mais a margem, mas que desse uma importância à distância dos centros, fosse interessante. O critério que associa o *score* Golub, como era de esperar, gerou um resultado satisfatório para algumas bases e para outras não, sendo, por isso, descartado.

Já dentro dos critérios associados à função que utiliza os valores de margem, o critério que maximiza a distância dos centros, juntamente com a maximização da margem (retirada da menor componente do vetor w), foi o que gerou as soluções com os menores erros associados. Por isso, esse critério foi escolhido.

Tabela 5.9 – Critérios de ramificação e medidas de avaliação para as bases lineares.

Base	f	r	I	RI	E	NT	NI	Podas	D	γ	10-*fold*
Robot LP4	λ	1	558	259	203	141	0	272	4	0,246	3,77%
		2	517	244	181	99	0	275	4	0,246	3,77%
		3	407	160	141	89	0	240	4	0,331	3,48%
		4	518	226	187	130	0	250	4	0,246	3,77%
		5	389	144	137	100	19	207	4	0,331	3,48%
	$\gamma \cdot C$	1	106	0	74	6	148	31	5	1,057	4,25%
		2	98	0	67	8	134	30	5	1,057	4,25%
		3	651	262	270	165	13	307	4	0,331	3,48%
		4	658	256	269	188	1	307	4	0,331	3,48%
		5	145	0	76	50	152	69	5	0,087	6,96%
Mushroom	λ	1	267	55	73	66	4	190	5	0,224	0,00%
		2	288	73	84	62	12	204	5	0,224	0,00%
		3	258	63	72	58	5	182	5	0,224	0,00%
		4	264	64	74	65	5	186	5	0,224	0,00%
		5	262	56	68	86	10	194	5	0,224	0,00%
	$\gamma \cdot C$	1	133	0	96	19	192	36	5	0,224	0,00%
		2	140	0	101	19	202	39	5	0,224	0,00%
		3	2072	565	1351	555	57	709	5	0,224	0,00%
		4	1287	274	743	412	39	524	5	0,224	0,00%
		5	177	0	96	65	192	81	5	0,224	0,00%
Prostate	λ	1	858	522	380	143	0	389	4	18,912	2,10%
		2	959	623	382	142	0	484	4	18,912	2,10%
		3	903	604	383	103	0	467	4	18,912	2,10%
		4	821	476	327	162	0	397	4	18,912	2,10%
		5	688	398	263	117	0	375	4	14,414	2,95%
	$\gamma \cdot C$	1	120	0	89	11	178	31	7	25,262	9,10%
		2	124	0	92	11	184	31	7	9,697	10,48%
		3	937	256	356	376	14	526	5	3,624	5,40%
		4	982	257	366	390	6	576	5	3,624	5,40%
		5	166	0	89	58	178	77	7	25,262	9,10%

5.5.7 Resultados finais

Foram comparadas as soluções dos métodos Golub, Nearest Shrunken Centroid[2] (NSC), Recursive Feature Elimination e AOS, além da solução sem a eliminação de variáveis (linha SVM).

Para o Golub, foram removidas as características segundo seus *scores* até não se ter mais uma separabilidade linear ou, para as bases não lineares, até o algoritmo de classificação não mais convergir.

[2] O método Nearest Srhunken Centroid (NSC), desenvolvido por Tibshirani et al. (2002), baseia-se no classificador do protótipo mais próximo.

Tabela 5.10 – Critérios de ramificação e medidas de avaliação para a base Ionosphere.

Base	f	r	I	RI	E	NT	NI	Podas	D	γ	10-*fold*
Ionosphere	χ	1	2081	1498	1296	536	0	148	3	0,001	15,87%
		2	2224	1614	1412	563	0	139	3	0,001	15,87%
		3	593	462	330	69	0	67	3	0,002	10,74%
		4	834	535	522	237	0	74	3	0,001	14,92%
		5	1142	784	661	292	0	157	3	0,001	13,24%
	$\mathcal{C} \cdot \chi$	1	74	0	49	19	98	24	4	0,001	16,60%
		2	73	0	49	18	98	24	4	0,001	16,60%
		3	8085	6011	5062	2010	0	979	3	0,001	15,87%
		4	9335	7190	6130	2085	162	1052	3	0,001	15,87%
		5	87	0	48	32	96	39	4	0,001	16,60%

Para o NSC, foi utilizada a versão do pacote “pamr”, para a linguagem R, referenciada por (TIBSHIRANI et al., 2003). Utilizou-se, conforme sugerem os autores, um valor para o parâmetro Δ (*threshold*) que gerasse a menor quantidade de atributos, sem ocorrer erros demasiados.

Para os testes feitos nos algoritmos RFE e AOS foi utilizado, como mostrado nas seções anteriores, o IMA, na sua versão IMA_∞ para as bases lineares e o IMA dual para as bases não lineares.

Para o RFE, o processo utilizado foi a remoção de uma característica por vez até o problema ficar não linearmente separável, ou o algoritmo de classificação não ter convergência, para as bases não linearmente separáveis.

Com base nos resultados apresentados nas seções anteriores, os parâmetros escolhidos para as execuções do AOS foram: fator de ramificação b: 3; profundidade da poda p: 2; profundidade do corte c: 2; critério de ramificação: w/C; medida de avaliação: γ.

Para as bases de *microarrays*, foi utilizado o RFE até a dimensão 100, para tornar a busca gerenciável. A partir desta quantidade de atributos, então, foi utilizado o AOS, com as mesmas parametrizações definidas acima. Foi realizado um teste com a base Colon, que possui a menor quantidade de atributos dentre as bases de *microarrays*, com o conjunto total de atributos e se obteve o mesmo resultado que o processo utilizando o RFE até a dimensão 100 e só depois utilizando o AOS. Esse resultado, que não garante que ocorrerá para todas as demais bases, mostra que a busca se torna mais difícil quando o problema possui menos atributos, sendo necessária, nesse ponto, uma busca mais ampla.

A fila de estados foi implementada utilizando uma estrutura de *heap*, uma vez que, a cada escolha, somente é necessário o candidato com a maior margem e não que a lista esteja totalmente ordenada. Com isso, o desempenho do algoritmo é aumentado. A tabela de dispersão (*hash*) foi implementada utilizando a seguinte função de espalhamento (dispersão):

$$h\left(k\right) = \left(\sum_{i} f_i^2\right) \text{ mod } n, \tag{5.16}$$

onde f_i são as características pertencentes a um determinado conjunto e n é um número primo, para evitar posições repetidas. Nos testes foi utilizado um valor de n igual a 161.387. Para cada

uma das k posições, foi implementado um vetor com 100 posições para evitar colisões.

Os resultados encontrados para os métodos são apresentados na Tabela 5.11.

Tabela 5.11 – Comparação entre os métodos.

Método	Base	F	γ	10-*fold*	Valid.	Base	F	γ	10-*fold*	Valid.
SVM	Mushroom	98	0,366	0,00%	0,00%	Synthetic	60	7,721	0,03%	0,50%
Golub		39	0,236	0,00%	0,00%		35	0,984	5,10%	7,00%
NSC		3	–	9,70%	11,00%		6	–	15,25%	17,50%
RFE		5	0,224	0,00%	0,00%		7	0,618	0,93%	3,50%
AOS		5	0,224	0,00%	0,00%		5	0,131	0,35%	4,00%
SVM	Leukemia	7129	18181,530	0,00%	8,33%	LP4 Robot	90	7,028	17,84%	12,82%
Golub		5	97,609	8,60%	12,50%		17	0,062	22,23%	10,26%
NSC		4	–	6,25%	12,50%		5	–	20,51%	20,51%
RFE		4	1568,592	3,80%	12,50%		7	0,941	6,57%	7,69%
AOS		2	177,635	0,60%	25,00%		4	0,331	3,48%	2,56%
SVM	Colon	2000	1529,516	16,55%	19,05%	Sonar	60	0,007	27,21%	24,29%
Golub		9	59,396	19,30%	19,05%		41	0,001	28,35%	27,14%
NSC		7	–	21,95%	19,05%		9	–	31,16%	28,57%
RFE		5	116,788	8,40%	14,29%		25	0,003	13,59%	30,00%
AOS		4	41,502	7,45%	28,57%		21	0,001	12,20%	30,00%
SVM	Breast	12625	5222,205	26,50%	12,50%	Ionosphere	34	0,095	6,03%	12,82%
Golub		2	191,930	8,50%	12,50%		7	0,015	11,12%	10,26%
NSC		2	–	43,75%	50,00%		13	–	17,95%	22,22%
RFE		3	440,323	13,00%	25,00%		4	0,001	12,53%	13,68%
AOS		2	400,317	0,00%	12,50%		3	0,002	10,74%	10,26%
SVM	Prostate	12600	529,011	14,82%	5,88%	WDBC	30	0,053	37,20%	37,37%
Golub		15	16,143	17,43%	8,82%		5	0,046	38,13%	37,89%
NSC		3	–	13,24%	5,88%		5	–	11,87%	7,37%
RFE		6	24,496	8,00%	5,88%		2	0,052	37,20%	37,37%
AOS		4	18,912	2,10%	2,94%		2	0,052	37,20%	37,37%
SVM	DLBCL	5468	14688,808	2,93%	7,69%	Wine	13	0,093	40,30%	38,98%
Golub		11	479,156	14,43%	19,23%		3	0,093	35,52%	25,42%
NSC		10	–	15,68%	15,38%		2	–	12,61%	10,17%
RFE		2	156,619	3,73%	11,54%		3	0,093	40,30%	37,29%
AOS		2	269,738	0,17%	7,69%		2	0,088	35,01%	22,03%

Os resultados mostram que o algoritmos AOS foi melhor, ou pelo menos igual, em todas as bases testadas. Sempre conseguiu uma menor quantidade, ou a mesma, de atributos que os

demais métodos, com exceção em relação ao NSC em 3 bases, Mushroom (que o NSC obteve um subconjunto menor, porém com uma taxa de erro alta), Prostate e Sonar. Gerou, ainda, na maioria das bases, menores erros de validação cruzada e de validação no conjunto de teste.

Para a base WDBC, em que o NSC gerou taxas de erros inferiores, foi realizada uma verificação com um classificador SVM, para a comparação ficar mais real. No subconjunto em questão foi detectado um erro de 37,20% de 10-*fold* e 37,37% de teste, ou seja, os mesmos erros do AOS, sendo que o AOS chegou em um subconjunto com somente 2 atributos, contra 5 do NSC, mostrando que, como método de seleção, o AOS foi superior.

Para a base sintética Robot LP4, o NSC classificou todas as amostras como sendo de uma mesma classe. Com isso, ele obteve exatamente a quantidade de amostras da outra classe como erros de 10-$fold$ e de validação.

É importante observar que para a base Wine, tanto o NSC quanto o AOS chegaram no mesmo subconjunto de atributos, porém pelo NSC esse subconjunto obteve erros menores. Isso se deve ao classificador do NSC e não pelo método de seleção, ou seja, para essa base de dados, os dois algoritmos obtiveram a mesma solução.

Para as bases em que o AOS obteve uma menor quantidade de características, porém um erro de 10-*fold* e/ou de validação superior, foram feitas comparações da solução do AOS com a dimensão alcançada pelo RFE. A Tabela 5.12 mostra essas comparações. É possível perceber que o AOS sempre encontra uma solução melhor, com margem superior e erros inferiores (ou iguais), com a mesma quantidade de atributos que a encontrada pelo RFE.

Para testar o poder do classificador dual nas bases lineares, utilizou-se o AOS, a partir dos resultados lineares finais, em algumas bases para tentar a retirada de ainda mais características.

Tabela 5.12 – Comparação entre o AOS e o RFE para a mesma dimensão.

Base	F	RFE			AOS		
		γ	10-*fold*	Valid.	γ	10-*fold*	Valid.
Leukemia	4	1568,592	3,80%	12,50%	1836,574	3,55%	8,33%
Colon	5	116,788	8,40%	14,29%	218,454	2,25%	14,29%
Synthetic	7	0,618	0,93%	3,50%	1,744	0,00%	3,50%

Dentre essas bases estão as que o NSC conseguiu um subconjunto com menos atributos que o AOS, com exceção da Mushroom que, mesmo no espaço dual, o AOS não conseguiu retirar mais características. A Tabela 5.13 mostra os resultados para essas bases, comparando o conjunto total de atributos com os conjuntos selecionados pelos métodos linear e não linear.

Com base nas informações da tabela, conclui-se que, para as bases lineares, não é tão interessante a remoção de mais características do que as que compõem um subconjunto linearmente separável. A utilização de remoções no espaço dual remove poucos atributos a mais e gera erros superiores aos dos subconjuntos encontrados com classificadores lineares. A exceção fica por conta da base Sonar, onde a remoção com um classificador não linear removeu uma quantidade significativa de atributos a mais e obteve erros similares aos gerados com um classificador linear.

É possível concluir que o algoritmo que foi proposto por esse trabalho gerou resultados bastante satisfatórios, tanto para classificadores lineares, em especial o IMA com a formulação L_∞, quanto para classificadores não lineares.

O algoritmo permite uma busca muito mais ampla do que uma busca míope, que não possui a capacidade de realizar *backtracking*. Com essa busca, mesmo não sendo de fato completa, ele é capaz de encontrar subconjuntos de atributos altamente discriminatórios, removendo sempre uma quantidade bastante significativa de características.

Tabela 5.13 – Comparação entre o AOS linear (L) e não linear (NL).

Base	Método	SVM				AOS			
		F	γ	10-*fold*	Valid.	F	γ	10-*fold*	Valid.
Prostate	L	12600	529,011	14,82%	5,88%	4	18,912	2,10%	2,94%
	NL		0,121	48,57%	50,00%	2	0,121	45,90%	50,00%
Colon	L	2000	1529,516	16,55%	19,05%	4	41,502	7,45%	28,57%
	NL		0,165	34,00%	38,10%	2	0,165	34,00%	38,10%
Sonar	L	60	0,007	27,21%	24,29%	21	0,001	12,20%	30,00%
	NL		0,090	13,95%	8,57%	4	0,001	17,97%	25,71%
Synthetic	L	60	7,721	0,03%	0,50%	5	0,131	0,35%	4,00%
	NL		0,050	50,00%	50,00%	2	0,050	5,42%	6,00%
Robot LP4	L	90	7,028	17,84%	12,82%	4	0,331	3,48%	2,56%
	NL		0,140	20,36%	20,51%	2	0,140	9,04%	7,69%

5.6 Considerações finais

O problema de seleção de características não possui uma solução trivial. Para cada tipo de problema, um determinado processo é mais adequado do que outro e em cada problema se consegue eliminar mais ou menos características.

Nesse trabalho foi apresentado o AOS, um algoritmo para a seleção de características, que utiliza critérios e medidas de qualidade provenientes de classificadores de larga margem e ex-

plora com eficiência o espaço de possibilidades. Esse processo é um método bastante eficaz de seleção de características, pois o mesmo algoritmo gera os subconjuntos candidatos e executa o indutor sobre eles, selecionando sempre o melhor estado em todas as dimensões.

Como classificador de larga margem, o algoritmo utilizado foi o IMA, na sua versão primal com suas formulações arbitrárias, destaque para a formulação L_∞, e na sua versão dual.

Como pode ser observado, o AOS apresentou resultados bastante satisfatórios. Ele foi superior aos demais métodos testados em todos os experimentos, sempre obtendo margens iguais ou superiores e erros de generalização iguais ou inferiores.

O algoritmo, além de permitir encontrar o subconjunto de maior margem em cada dimensão, também é capaz de encontrar, dentre esses subconjuntos, a solução que gera a menor quantidade de erros, ou seja, o subconjunto mais discriminatório.

É possível fazer um comparativo com o supercomputador da IBM, Deep Blue, que em 1997 ganhou do campeão mundial de xadrez, Garry Kasparov, feito que, à época, parecia ser impossível. O computador realmente não era capaz de prever o conjunto total de jogadas, mas com a busca heurística e as podas que possuía foi capaz de realizar sempre jogadas muito boas, conseguindo, assim, vitórias sobre seu oponente, mostrando que mesmo para problemas com soluções ótimas “impossíveis” é bastante possível encontrar soluções muito interessantes.

O algoritmo AOS combina técnicas de duas grandes áreas da Inteligência Artificial, a Cognitiva, com a busca de caminhos, e Conexionista, através do aprendizado de máquina, nesse caso as máquinas de vetores de suporte, abrindo possibilidades de estudos nessa combinação.

5.7 Referências

AGMON, S. The relaxation method for linear inequalities. *Canadian Journal of Mathematics*, v. 6, n. 3, p. 382–392, 1954.

AIZERMAN, A.; BRAVERMAN, E. M.; ROZONER, L. I. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, v. 25, p. 821–837, 1964.

ALON, U. et al. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences of the United States of America*, Department of Molecular Biology, Princeton University, Princeton, NJ 08540, USA., v. 96, n. 12, p. 6745–6750, 1999.

ASUNCION, A.; NEWMAN, D. J. *UCI Machine Learning Repository*. Irvine, CA: University of California, School of Information and Computer Science, 2007. Disponível em: <http://www.ics.uci.edu/ mlearn/MLRepository.html>.

BLOCK, H. The perceptron: a model for brain functioning. *Reviews of Modern Physics*, v. 34, p. 123–135, 1962.

BLUM, A. L.; LANGLEY, P. Selection of relevant features and examples in machine learning. *Artificial Intelligence*, Elsevier Science Publishers Ltd., v. 97, n. 1-2, p. 245–271, 1997.

BOSER, B. E.; GUYON, I. M.; VAPNIK, V. N. A training algorithm for optimal margin classifiers. In: *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory*. New York, NY, USA: ACM Press, 1992. p. 144–152.

BROWNE, M. Cross-validation methods. *Journal of Mathematical Psychology*, v. 44, n. 1, p. 108–132, 2000.

DAX, A. The distance between two convex sets. *Linear Algebra and its Applications*, v. 416, p. 184–213, 2006.

GEISSER, S. The predictive sample reuse method with applications. *Journal of the American Statistical Association*, American Statistical Association, v. 70, n. 350, 1975.

GOLUB, T. R. et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, v. 286(5439), p. 531–537, 1999.

GUYON, I. Welcome and introduction to the problem of feature/variable selection. In: *NIPS 2001 Workshop on Feature/Variable Selection*. [S.l.: s.n.], 2001.

HART, P.; NILSSON, N.; RAPHAEL, B. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, v. 4, n. 2, p. 100–107, 1968.

HERBRICH, R. *Learning Kernel Classifiers: Theory and Algorithms*. Cambridge, Massachusetts: MIT Press, 2002. (Adaptive computation and machine learning).

HUGHES, G. On the mean accuracy of statistical pattern recognizers. *Information Theory, IEEE Transactions on*, IEEE, v. 14, n. 1, p. 55–63, 1968.

JOHN, G. H.; KOHAVI, R.; PFLEGER, K. Irrelevant features and the subset selection problem. In: *Proceedings of the Eleventh International Conference on Machine learning*. New Brunswick, NJ: [s.n.], 1994. p. 121–129.

KIRA, K.; RENDELL, L. A. The feature selection problem: Traditional methods and a new algorithm. In: *Proceedings of the Tenth National Conference on Artificial Intelligence*. AAAI Press,

1992. (AAAI'92), p. 129–134. ISBN 0-262-51063-4. Disponível em: <http://dl.acm.org/citation.cfm?id=1867135.1867155>.

KOHAVI, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: *Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2*. San Francisco, CA: Morgan Kaufmann Publishers Inc., 1995. p. 1137–1143.

KOHAVI, R.; JOHN, G. H. Wrappers for feature subset selection. *Artificial Intelligence*, v. 97, n. 1-2, p. 273–324, 1997.

LEITE, S. C.; FONSECA NETO, R. Incremental margin algorithm for large margin classifiers. *Neurocomputing*, v. 71, p. 1550–1560, 2008.

MCLACHLAN, G. J.; DO, K. A.; AMBROISE, C. *Analyzing microarray gene expression data*. Hoboken, New Jersey: Wiley-Interscience, 2004. (Wiley series in probability and statistics).

MINSKY, M.; PAPERT, S. *Perceptrons: An Introduction to Computational Geometry*. Cambridge, MA, USA: MIT Press, 1969.

MOSTELLER, F.; TUKEY, J. W. Data analysis, including statistics. In: LINDZEY, G.; ARONSON, E. (Ed.). *Handbook of Social Psychology*. Boston: Addison-Wesley, 1968. v. 2.

MOTZKIN, T. S.; SCHOENBERG, I. J. The relaxation method for linear inequalities. *Canadian Journal of Mathematics*, v. 6, n. 3, p. 393–404, 1954.

MUCCIARDI, A. N.; GOSE, E. E. A comparison of seven techniques for choosing subsets of pattern recognition properties. *IEEE Transactions on Computers*, IEEE Computer Society, Washington, DC, v. 20, p. 1023–1031, 1971.

NOVIKOFF, A. B. On convergence proofs for perceptrons. In: *Proceedings of the Symposium on the Mathematical Theory of Automata*. [S.l.: s.n.], 1963. v. 12, p. 615–622.

PEDROSO, J. P.; MURATA, N. Support vector machines with different norms: motivation, formulations and results. *Pattern Recognition Letters*, v. 22, n. 12, p. 1263–1272, 2001.

ROSENBLATT, F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, v. 65, p. 386–408, 1958.

ROSSET, S.; ZHU, J.; HASTIE, T. Boosting as a regularized path to a maximum margin classifier. *Journal of Machine Learning Research*, v. 5, p. 941–973, 2004.

SHAWE-TAYLOR, J.; CRISTIANINI, N. Margin distribution and soft margin. In: ____. *Advances in Large Margin Classifiers*. Cambridge, MA: MIT Press, 1999. (Neural information processing series), cap. 2, p. 5–16.

SHAWE-TAYLOR, J.; CRISTIANINI, N. *Kernel Methods for Pattern Analysis*. New York, NY: Cambridge University Press, 2004.

SINGH, D. et al. Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell*, v. 1, n. 2, p. 203–209, 2002.

TIBSHIRANI, R. et al. Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proceedings of the National Academy of Sciences*, v. 99, n. 10, p. 6567–6572, 2002.

TIBSHIRANI, R. et al. Class prediction by nearest shrunken centroids, with applications to dna microarrays. *Statistical Science*, v. 18(1), p. 104–117, 2003.

VAPNIK, V. N. *The nature of statistical learning theory*. New York, NY, USA: Springer-Verlag New York, Inc., 1995.

VILLELA, S. M. et al. Seleção de características utilizando busca ordenada e um classificador de larga margem. In: *X CBIC - Congresso Brasileiro de Inteligência Computacional*. Fortaleza, CE: [s.n.], 2011.

WAHBA, G.; WOLD, S. A completely automatic french curve: Fitting spline functions by cross-validation. *Communications in Statistics*, v. 4, n. 1, 1975.

YU, L.; LIU, H. Efficient feature selection via analysis of relevance and redundancy. *Journal of Machine Learning Research*, MIT Press, Cambridge, MA, USA, v. 5, p. 1205–1224, 2004.

Sobre os autores

ANDRÉ DOS SANTOS GONZAGA, <http://lattes.cnpq.br/9073053051723304>, é acadêmico de Ciência da Computação na Universidade de São Paulo e bolsista de Iniciação Científica na Empresa Brasileira de Pesquisa Agropecuária. Tem experiência na área de ciência da computação, com ênfase em inteligência artificial.

BENILTON DE SÁ CARVALHO, <http://lattes.cnpq.br/0897291971174045>, possui graduação e mestrado em Estatística pela Universidade Estadual de Campinas e doutorado em Bioestatística pela Universidade Johns Hopkins. Foi pesquisador associado no Departamento de Oncologia da Universidade de Cambridge, professor afiliado do Instituto de Biologia Computacional de Cambridge, pesquisador associado honorário no Grupo de Biologia Computacional do Cancer Research UK até Dez/2012. Atualmente, é associado ao Departamento de Genética Médica da Faculdade de Ciências Médicas da Universidade Estadual de Campinas e desenvolvedor no Projeto Bioconductor. Tem experiência em Bioestatística, com ênfase em Biologia Computacional e Bioinformática, atuando principalmente nos seguintes temas: genotipagem, análise de número de cópias, análise de expressão gênica, SNPs, RNA-seq e DNA-seq. Suas

atividades contam com o apoio da FAPESP (2012/21548-1 e 2013/00506-1) e CNPq (407856/2012-9).

CARLOS CRISTIANO HASENCLEVER BORGES, <http://lattes.cnpq.br/2487554612123446>, é graduado em Engenharia Civil pela Universidade Federal de Juiz de Fora, mestre e doutor em Engenharia Civil pela Universidade Federal do Rio de Janeiro. Trabalhou no Laboratório Nacional de Computação Científica até pedir vacância do cargo em 2009. Atualmente trabalha na Universidade Federal de Juiz de Fora no Departamento de Ciência da Computação. Tem experiência na área de modelagem computacional, com atuação em análise numérica, aprendizagem de máquina e inteligência computacional com aplicações em problemas de engenharia estrutural e biologia computacional.

EDUARDO RAUL HRUSCHKA, <http://lattes.cnpq.br/7288946669857591>, é cientista-chefe da Big Data e professor associado II do Departamento de Ciências da Computação, no Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo em São Carlos. Graduado em Engenharia Civil pela Universidade Federal do Paraná, concluiu mestrado e doutorado no Programa Interdisciplinar de Computação de Alto Desempenho da COPPE/UFRJ. Realizou pós-doutoramento na University of Texas at Austin. Obteve seu título de Livre-Docente em Ciências de Computação pela Universidade de São Paulo. Atualmente, é pesquisador 1D do Conselho Nacional de Desenvolvimento Científico e Tecnológico e editor associado do periódico Information Sciences da Elsevier. Atua nas áreas de mineração de dados e aprendizado de máquina, e seus principais interesses atualmente incluem: técnicas de agrupamento e de classificação, aprendizado semi-supervisionado, transferência de aprendizado e de conhecimento, agregação de classificadores e de agrupadores, algoritmos evolutivos, seleção de atributos para agrupamento e

para classificação, preenchimento de valores ausentes, métodos Bayesianos e redes neurais.

FABIANA BARICHELLO MOKRY, <http://lattes.cnpq.br/7288946669857591>, é pesquisadora Recém-Doutora do Programa de Pós-Graduação em Genética Evolutiva e Biologia Molecular da Universidade Federal de São Carlos, desenvolvendo pesquisas nas áreas de genômica funcional, seleção genômica, estudos de associação ampla do genoma e anotação gênica. Possui título de Doutor e Mestre em Genética e Melhoramento Animal pela Universidade Estadual Paulista Júlio de Mesquita Filho/Jaboticabal. Tem experiência na área de zootecnia e bioinformática, com ênfase em genética e melhoramento dos animais domésticos, atuando principalmente nos seguintes temas: estimativas de parâmetros genéticos, estatística aplicada ao melhoramento genético animal, estatística bayesiana, modelagem estatística, análise de dados discretos, análise de dados genômicos, seleção, acasalamentos e cruzamentos de bovinos de corte.

FABRÍZZIO CONDÉ DE OLIVEIRA, <http://lattes.cnpq.br/5149229612250815>, possui graduação em Bacharelado em Matemática pela Universidade Federal de Juiz de Fora, mestrado em Economia Empresarial pela Universidade Candido Mendes/RJ e é doutorando em Modelagem Computacional pela Universidade Federal de Juiz de Fora. Atualmente, é professor da disciplina Métodos Quantitativos do curso MBA em Gestão de Custos e Finanças Empresariais do Instituto Metodista Granbery, professor das disciplinas Cálculo, Álgebra Linear, Geometria Analítica, Pesquisa Operacional, Simulação e Matemática Financeira da Universidade Salgado de Oliveira – Campus Juiz de Fora. Tem experiência em estatística, aprendizado de máquina, algoritmos evolutivos, equações diferenciais, seleção de atributos em bioinformática, com ênfase em seleção de marcadores moleculares do tipo

SNPs (polimorfismos de base única), e na construção de modelos matemáticos para otimização em programação matemática.

FABYANO FONSECA E SILVA, <http://lattes.cnpq.br/6661948983681991>, possui graduação em Zootecnia pela Universidade Federal de Lavras; mestrado e doutorado em Estatística e Experimentação Agropecuária pela Universidade Federal de Lavras; pós-doutorado em Estatística-Genética pela University of Wisconsin e pós-doutorado em Seleção Genômica pela Wageningen University. Especialização em Seleção Genômica Ampla pela Universidad de Zaragoza e especialização em Seleção Genômica Ampla em Suínos no Institute for Pigs Genetics. Atualmente é professor adjunto IV do Departamento de Zootecnia da Universidade Federal de Viçosa, sendo membro do Departamento de Estatística entre 2005 e 2013, e atua como membro permanente dos programas de pós-graduação em Estatística Aplicada e Biometria e em Genética e Melhoramento. Tem experiência nas áreas de probabilidade e estatística aplicada ao melhoramento animal e à experimentação agropecuária, inferência bayesiana, análise de séries temporais e de dados genômicos (marcadores moleculares e bioinformática).

FERNANDA NASCIMENTO ALMEIDA, <http://lattes.cnpq.br/6281634935701289>, possui graduação em Ciências Biológicas, com ênfase em Biomedicina, pela Universidade Estadual de Santa Cruz, mestrado em Modelagem Computacional (Bioinformática), realizado no Laboratório Nacional de Computação Científica e doutorado em Bioinformática, realizado na Universidade de São Paulo. Atualmente atua como colaboradora no Laboratório de Bioinformática e Genômica Animal da Embrapa Gado de Leite. Tem experiência na área de genética e biologia molecular, com ênfase em bioinformática, atuando principalmente nos seguintes temas: proveniência e análise de dados em sistemas de informação

biológicos; banco de dados biológicos; análise e comparação de genomas.

FRANCISCO PEREIRA LOBO, <http://lattes.cnpq.br/9614758933055047>, é bacharel em Biologia, com ênfase em Bioquímica e Imunologia, mestre em Bioquímica e Imunologia e doutor em Bioinformática pela Universidade Federal de Minas Gerais. Fez seu pós-doutorado no Laboratório de Genômica de Parasitos da Universidade Federal de Minas Gerais, estudando genômica comparativa de tripanosomatídeos e predição automática de proteínas imunogênicas. Possui experiência nas área de bioinformática, bioquímica, biologia molecular e genética, em especial em genômica comparativa, metodologias homologia-independentes para genômica comparativa, coevolução de genomas e genômica de vírus e parasitas. Atualmente é pesquisador Empresa Brasileira de Pesquisa Agropecuária, atuando no Laboratório Multiusuário de Bioinformática da Embrapa Informática Agropecuária.

LUCIANA CORREIA DE ALMEIDA REGITANO, <lattes.cnpq.br/9595338480545794>, possui graduação em Medicina Veterinária pela Universidade Federal do Paraná, mestrado em Genética e Melhoramento de Plantas pela Universidade de São Paulo, com parte realizada na Brock University, Canadá, pelo programa sanduíche da CAPES, doutorado em Genética e Melhoramento de Plantas, pela Universidade de São Paulo e Pós-doutorado no Agricultural Research Service do United States Department of Agriculture. É pesquisadora da Empresa Brasileira de Pesquisa Agropecuária e docente do programa de Pós-graduação em Genética e Evolução da Universidade Federal de São Carlos. Tem experiência na área de genética animal, com ênfase em biologia molecular, atuando principalmente no mapeamento de genes para

características de interesse econômico em bovinos criados sob condições tropicais.

MARCOS VINÍCIUS GUALBERTO BARBOSA DA SILVA,, <http://lattes.cnpq.br/6353954532527478>, possui graduação em Zootecnia pela Universidade de São Paulo, mestrado em Zootecnia pela Universidade Federal de Minas Gerais, doutorado em Genética e Melhoramento pela Universidade Federal de Viçosa e pós-doutorado pelo United States Department of Agriculture. Atualmente é pesquisador A da Empresa Brasileira de Pesquisa Agropecuária, coordenador-geral do Programa de Melhoramento Genético do Girolando e lidera pesquisas em seleção genômica em raças leiteiras no Brasil e em sequenciamento do genoma de raças zebuínas. Tem experiência na área de genética, com ênfase em bioinformática, atuando principalmente nos seguintes temas: marcadores moleculares, produção de leite, bovino de leite, sequenciamento de genomas e seleção genômica.

MAURÍCIO DE ALVARENGA MUDADU, <http://lattes.cnpq.br/4415254803722805>, possui graduação em Ciências Biológicas, mestrado em Bioquímica e Imunologia e doutorado em Bioinformática pela Universidade Federal de Minas Gerais. É especialista em Bioinformática pelo Laboratório Nacional de Computação Científica. Fez pós-doutorado no Commonwealth Scientific and Industrial Research Organisation, com foco em estudos de associação genômica ampla e redes gênicas. Tem experiência em desenvolvimento de ferramentas *web*, bancos de dados e programação com ênfase em bioinformática, genética e bioquímica. Trabalha atualmente com melhoramento genético voltado para agropecuária com ênfase em análises de genotipagem em larga escala e sequenciamento de nova geração.

RAUL FONSECA NETO, <http://lattes.cnpq.br/3572434390881704>, possui graduação em Engenha-

ria Civil pela Universidade Federal de Juiz de Fora, especialização em Pesquisa Operacional e mestrado em Engenharia de Transportes pelo Instituto Militar de Engenharia, doutorado em Engenharia de Sistemas e Computação pela Universidade Federal do Rio de Janeiro e pós-doutorado em Modelagem Computacional pelo Laboratório Nacional de Computação Científica. Atualmente é professor associado IV da Universidade Federal de Juiz de Fora onde atua nos programas de pós-graduação em Modelagem Computacional e em Ciência da Computação. Tem experiência na área de ciência da computação e engenharia de sistemas, com ênfase em sistemas de computação, atuando principalmente nos seguintes temas: inteligência artificial, otimização, fluxo em redes, planejamento e *scheduling*, complexidade de algoritmos, aprendizado de máquinas, reconhecimento de padrões, redes complexas e bioinformática.

ROBERTO HIROSHI HIGA, <http://lattes.cnpq.br/9214539588712509>, possui graduação em Engenharia Elétrica pela Universidade Federal de Uberlândia, mestrado e doutorado em Engenharia Elétrica pela Universidade Estadual de Campinas. Atualmente é celetista da Empresa Brasileira de Pesquisa Agropecuária. Tem experiência na área de ciência da computação, com ênfase em matemática da computação. Atuando principalmente nos seguintes temas: reconhecimento de padrões, aprendizado de máquina e bioinformática.

SAUL DE CASTRO LEITE, <http://lattes.cnpq.br/4802548698016081>, possui graduação em Ciência da Computação (Mag Cum Laude) pela Oklahoma State University e doutorado em Modelagem Computacional pelo Laboratório Nacional de Computação Científica. Atualmente é professor adjunto da Universidade Federal de Juiz de Fora, atuando principalmente nos seguintes temas: aproximações por difusão para sistemas de filas,

métodos numéricos para problemas de controle, reconhecimento de padrões.

SAULO MORAES VILLELA, <http://lattes.cnpq.br/3358075178615535>, possui graduação em Ciência da Computação pela Universidade Federal de Juiz de Fora e mestrado e doutorado em Engenharia de Sistemas e Computação pela Universidade Federal do Rio de Janeiro. Atualmente é professor adjunto do Departamento de Ciência da Computação da Universidade Federal de Juiz de Fora. Tem experiência na área de ciência da computação, com ênfase em inteligência computacional, atuando principalmente nos seguintes temas: inteligência artificial, aprendizado de máquinas e otimização combinatória.

WAGNER ARBEX, <http://lattes.cnpq.br/7805432023782520>, possui graduação em Bacharelado em Matemática (Modalidade Informática) pela Universidade Federal de Juiz de Fora, mestrado em Sistemas e Computação pelo Instituto Militar de Engenharia e doutorado em Engenharia de Sistemas e Computação pela Universidade Federal do Rio de Janeiro. Atualmente é conselheiro da Associação Brasileira de Bioinformática e Biologia Computacional, representante institucional da Sociedade Brasileira de Computação, professor convidado do programa de pós-graduação em Estatística Aplicada e Biometria da Universidade Federal de Viçosa, professor adjunto da Universidade Federal de Juiz de Fora, onde atua nos programas de pós-graduação em Ciência da Computação e em Modelagem Computacional como docente e/ou em orientações, e analista da Empresa Brasileira de Pesquisa Agropecuária. Tem experiência na área de ciência da computação, com ênfase em bioinformática, atuando principalmente nos seguintes temas: bioinformática, *single nucleotide polymorphism*, modelagem computacional, melhoramento genético animal, aprendizado de máquina e sistema de inferência difusa.

Índice

Esta publicação foi editada com o TEXShop e processada pelo LATEX 2_ε, utilizando a classe abnTEX2 para o corpo do texto e o pacote e o estilo abnTEX2cite para os formatos das citações e referências bibliográficas. Os projetos abnTEX2 e abnTEX2cite são suítes gratuitas e de código aberto para LATEX que atendem aos requisitos das normas da ABNT para elaboração de documentos técnicos e científicos brasileiros e são desenvolvidos e mantidos pela ação voluntária de seus desenvolvedores e/ou usuários. Para maiores informações sobre os projetos abnTEX2 e abnTEX2cite, acesse a página <http://abntex2.googlecode.com/>.

Arquivo de impressão gerado em 17 de novembro de 2014.

TEX 3.14159265 (TEX Live 2014)