

UNIVERSIDADE FEDERAL DO RIO GRANDE
CENTRO DE CIÊNCIAS COMPUTACIONAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO
CURSO DE MESTRADO EM ENGENHARIA DE COMPUTAÇÃO

Dissertação

Modelos de classificação baseados em painel de genes e gene-específico para a predição de variantes genômicas.

João Pedro Hartmann Salomão

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal do Rio Grande, como requisito parcial para a obtenção do grau de Mestre em Engenharia de Computação

Orientador: Prof. Dra. Karina dos Santos Machado
Co-orientador: Prof. Dra. Mariana Recamonde Mendoza
Colaborador: Prof. Dr. Adriano Velasque Werhli

Rio Grande, 2021

Ficha Catalográfica

S173m Salomão, João Pedro Hartmann.

Modelos de classificação baseados em painel de genes e gene-específico para a predição de variantes genômicas / João Pedro Hartmann Salomão. – 2021.

79 f.

Dissertação (mestrado) – Universidade Federal do Rio Grande – FURG, Programa de Pós-Graduação em Computação, Rio Grande/RS, 2021.

Orientadora: Dra. Karina dos Santos Machado.

Coorientadora: Dra. Mariana Recamonde Mendoza.

Colaborador: Dr. Adriano Velasque Werhli.

1. SNV 2. Aprendizado de Máquina 3. Classificação I. Machado, Karina dos Santos II. Mendoza, Mariana Recamonde III. Werhli, Adriano Velasque IV. Título.

CDU 004

Catálogo na Fonte: Bibliotecário José Paulo dos Santos CRB 10/2344



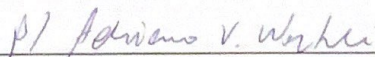
MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO RIO GRANDE
CENTRO DE CIÊNCIAS COMPUTACIONAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO
CURSO DE MESTRADO EM ENGENHARIA DE COMPUTAÇÃO

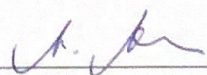
DISSERTAÇÃO DE MESTRADO

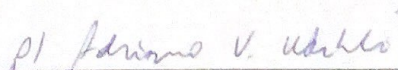
Modelos De Classificação Baseados Em Painel De Genes E Gene- Específico Para A Predição De Variantes Genômicas

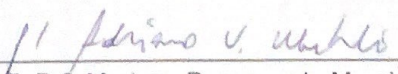
João Pedro Hartmann Salomão

Banca examinadora:


Prof. Dr. Prof. Dr. Bruno Oliveira


Prof. Dr. Andrej Aderhold


Prof.^a. Dr.^a. Karina dos Santos Machado
Orientadora


Prof.^a. Dr.^a. Mariana Recamonde Mendoza
Coorientadora

RESUMO

SALOMÃO, João Pedro Hartmann . **Modelos de classificação baseados em painel de genes e gene-específico para a predição de variantes genômicas..** 2021. 79 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Computação. Universidade Federal do Rio Grande, Rio Grande.

O genoma humano pode conter variações genéticas em seu DNA. Baseado-se em estudos anteriores, estas variações podem ser consideradas como patogênicas ou não, permitindo o uso de ferramentas de aprendizado de máquina capazes de prever o impacto causado por essas variantes nos humanos.

O objetivo deste trabalho, é propor uma metodologia de predição de variantes patogênicas, utilizando-se de técnicas de aprendizado de máquina capazes de criar modelos estatísticos para realizar essas predições.

Esta metodologia, utiliza técnicas de seleção de atributos e filtros nos conjuntos de dados para que seja possível criar modelos especializados para um determinado conjunto de genes, aprimorando a performance da predição.

Conforme os modelos vão se especializando, os resultados apresentam melhor performance em relação aos modelos genéricos demonstrando a possibilidade de construção de classificadores com melhor desempenho.

Palavras-chave: SNV, aprendizado de máquina, classificação.

ABSTRACT

SALOMÃO, João Pedro Hartmann . **Classification models based on gene panels and gene specific for genomic variants predictions.** 2021. 79 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Computação. Universidade Federal do Rio Grande, Rio Grande.

The human genome contain lots of genetic variations in its DNA. Based on previous studies, these variations can be considered pathogenic or not, making possible to use machine learning tools that can take these known pathogenic variants, and predict their pathogenicity.

The main goal of this project, is to propose a methodology for predicting if a variant is pathogenic. Using machine learning techniques capable of creating statistical models to make this predictions.

The proposed methodology uses feature selection techniques and filters in the dataset so that it is possible to create specialized models for a specific gene group, increasing the prediction performance.

As the models are trained, the results show that the specialized model performance is better than the generic ones which means that it is possible to create better classifiers as the models get specialized training.

Keywords: SNV, machine learning, classification.

LISTA DE FIGURAS

Figura 1	Exemplo de SNP em um gene presente no DNA.	15
Figura 2	Exemplo de mutação com troca de aminoácido	16
Figura 3	Exemplo de mutação sem troca de aminoácido	16
Figura 4	Exemplo de uma entrada de um classificador	17
Figura 5	Exemplo de imputação de dados	20
Figura 6	Exemplo de possíveis divisões de planos para o algoritmo SVM (Ray 2017)	21
Figura 7	Exemplo de uma curva ROC.	25
Figura 8	Workflow de execução de treinamento dos modelos	39
Figura 9	Diagrama entidade relacionamento do banco de dados	41
Figura 10	<i>Workflow</i> do processo de obtenção dos dados	42
Figura 11	Pipeline de execução da imputação de dados	44
Figura 12	Pipeline de execução de treinamento dos modelos	45
Figura 13	Histograma dos dados faltantes para o filtro com todos os genes . . .	48
Figura 14	Histograma e distribuição dos dados faltantes para o filtro com todos os genes após a filtragem de variáveis e instâncias	49
Figura 15	Densidade dos atributos selecionados pelo BIC	51
Figura 16	Acurácia dos modelos testados no dataset completo com seleção de atributos	52
Figura 17	Densidade de todos os atributos candidatos	53
Figura 18	Acurácia dos modelos testados no dataset com todos os candidatos sem a seleção de atributos	54
Figura 19	Histograma dos dados faltantes para o filtro com o painel de genes . .	56
Figura 20	Histograma e distribuição dos dados faltantes para o filtro com o pai- nel de genes após a filtragem de variáveis e instâncias	57
Figura 21	Densidade dos atributos selecionados pelo BIC	59
Figura 22	Acurácia dos modelos testados no dataset com a seleção de atributos no painel de genes	59
Figura 23	Densidade de todos os atributos candidatos	60
Figura 24	Acurácia dos modelos testados no dataset com todos os atributos can- didatos para o o painel de genes	61
Figura 25	Histograma dos dados faltantes para o filtro pelo gene BRCA1	62
Figura 26	Histograma e distribuição dos dados faltantes para o filtro com o pai- nel de genes após a filtragem de variáveis e instâncias	63
Figura 27	Densidade dos atributos selecionados pelo AIC	65

Figura 28	Acurácia dos modelos testados no dataset com a seleção de atributos e o filtro para um gene	66
Figura 29	Densidade de todos os atributos candidatos	67
Figura 30	Acurácia dos modelos testados no dataset com todos os atributos candidatos com o filtro de um gene	68
Figura 31	Acurácia dos modelos testados no dataset com todos os atributos candidatos com o filtro do gene TP53	69
Figura 32	Gráfico de comparação dos resultados com a seleção de atributos . .	70
Figura 33	Gráfico de comparação dos resultados sem a seleção de atributos . . .	71

LISTA DE TABELAS

Tabela 1	Matriz de confusão para um modelo de predição de SNPs	24
Tabela 2	Atributos utilizados para o treinamento do classificador para o gene A1BG	28
Tabela 3	Scores de conservação utilizados como atributos pelas ferramentas de predição	32
Tabela 4	Atributos relacionados a propriedades das sequências de aminoácidos	33
Tabela 5	Atributos da estrutura 3D da proteína	34
Tabela 6	Mudanças físico-químicas dos aminoácidos	35
Tabela 7	Atributos relacionados às ligações da proteína	35
Tabela 8	Atributos referentes às anotações das regiões funcionais da proteína .	36
Tabela 9	Atributos referentes aos resultados de outras ferramentas	36
Tabela 10	Resumo do resultado da execução do RFE	50
Tabela 11	Distribuição das classes do atributo alvo nos dados de treinamento . .	50
Tabela 12	Distribuição das classes do atributo alvo nos dados de teste	50
Tabela 13	Matriz de confusão resultante dos testes. As linhas representam os valores reais e as colunas os valores preditos	52
Tabela 14	Matriz de confusão resultante dos testes com todos os candidatos . .	54
Tabela 15	Resumo do resultado da execução do RFE para o painel de genes . .	58
Tabela 16	Distribuição das classes do atributo alvo nos dados de treinamento . .	58
Tabela 17	Distribuição das classes do atributo alvo nos dados de teste	58
Tabela 18	Matriz de confusão resultante dos testes. As linhas representam os valores reais e as colunas os valores preditos	60
Tabela 19	Matriz de confusão resultante dos testes com todos os candidatos . .	61
Tabela 20	Resumo do resultado da execução do RFE para o painel de genes . .	64
Tabela 21	Distribuição das classes do atributo alvo nos dados de treinamento . .	64
Tabela 22	Distribuição das classes do atributo alvo nos dados de teste	64
Tabela 23	Matriz de confusão resultante dos testes. As linhas representam os valores reais e as colunas os valores preditos	66
Tabela 24	Matriz de confusão resultante dos testes com todos os candidatos . .	67
Tabela 25	Distribuição das classes do atributo alvo nos dados de teste do gene TP53	69
Tabela 26	Matriz de confusão resultante dos testes para o filtro de todos os genes	72
Tabela 27	Matriz de confusão resultante dos testes para o filtro do painel de genes	72

SUMÁRIO

1	INTRODUÇÃO	11
1.1	Objetivo Geral	12
1.2	Objetivos específicos	12
1.3	Justificativa	12
1.4	Organização do texto	13
2	REFERENCIAL TEÓRICO	14
2.1	SNPs	14
2.2	Aprendizado de máquina	16
2.2.1	Atributos	17
2.2.2	Instâncias ou entradas	17
2.2.3	Atributo alvo	17
2.2.4	Dados de treinamento e teste	18
2.3	Pré-processamento	18
2.3.1	Seleção de atributos	18
2.3.2	Imputação de dados	19
2.4	Aprendizado supervisionado	20
2.4.1	SVM	20
2.4.2	Random Forest	21
2.5	Seleção de modelos	21
2.5.1	AIC	22
2.5.2	BIC	22
2.6	Avaliação de modelos	23
2.6.1	Validação cruzada	23
2.6.2	Treinamento de um modelo	23
2.6.3	Teste de um modelo	23
2.6.4	Matriz de confusão	23
2.6.5	Acurácia	24
2.6.6	Recall ou Sensibilidade	24
2.6.7	Especificidade	24
2.6.8	Curva ROC	25
2.7	Amostragem	25
3	TRABALHOS RELACIONADOS	26
3.1	Ferramentas de predição genéricas	26
3.1.1	PolyPhen2	26
3.1.2	Mutation Assessor	27

3.1.3	Sift e Provean	27
3.2	Ferramentas de predição específicas	27
3.2.1	iFish	27
4	BASES DE DADOS, DADOS E ATRIBUTOS	29
4.1	Bases de dados	29
4.1.1	UCSC Genome Browser	30
4.1.2	Cosmic	30
4.1.3	UniProt	30
4.1.4	HGMD	31
4.1.5	1000 Genome Project	31
4.1.6	NCBI	31
4.2	Atributos	31
4.2.1	Scores de conservação	32
4.2.2	Atributos relacionados a propriedades das sequências	32
4.2.3	Atributos da estrutura 3D da proteína	34
4.2.4	Mudanças físico-químicas dos aminoácidos	34
4.2.5	Anotações das ligação da proteína	35
4.2.6	Anotações das regiões funcionais	35
4.2.7	Atributos especiais	36
4.2.8	Dados de entrada	36
4.2.9	Dados de saída	37
5	METODOLOGIA	38
5.1	Workflow	38
5.2	Coleta de dados	39
5.2.1	Base de dados	39
5.3	Pré-processamento e obtenção dos modelos	43
5.3.1	Filtros	43
5.3.2	Dados faltantes	43
5.3.3	Imputação de dados	44
5.3.4	Seleção de atributos	44
5.3.5	Treinamento dos modelos	45
5.3.6	Validação dos modelos	46
6	RESULTADOS	47
6.1	Modelos treinados com todos os genes disponíveis	47
6.1.1	Pré-processamento	47
6.1.2	Treinamento dos modelos	50
6.2	Modelos treinados com dados do painel de genes	55
6.2.1	Pré-processamento	55
6.2.2	Treinamento dos modelos	58
6.3	Modelos treinados com dados de um gene específico	61
6.3.1	Pré-processamento	61
6.3.2	Treinamento dos modelos	64
6.4	Treinamento do genes restantes no painel	68
7	DISCUSSÃO	70

8 CONCLUSÃO	74
8.1 Trabalhos Futuros	74
REFERÊNCIAS	76

1 INTRODUÇÃO

Dentro da área da bioinformática, existem diversas bases de dados com informações genômicas de diferentes espécies. Esses dados contêm informações do DNA de uma determinada espécie, compondo toda a sua informação genética. Esses dados podem ser utilizados para, por exemplo, a análise de variações genéticas causadoras de doenças, ou a identificação da função que um gene específico possui.

Na análise de mutações genéticas em humanos, os polimorfismos de nucleotídeo único ou SNP, são as variações genômicas comumente encontradas. Elas podem estar relacionadas a diversas enfermidades, hereditárias ou não, porém nem sempre há essa relação direta. (Yates e Sternberg 2013). Com isso, surge a necessidade de prever se um SNP está relacionado ou não, à uma doença em um indivíduo. A partir de SNPs já conhecidos, modelos de classificação são treinados pelas ferramentas, considerando diversas características das mutações, como estrutura 3D da proteína, características físico-químicas dos aminoácidos da substituição, scores de conservação da mutação, entre outros.

Dada a complexidade da tarefa preditiva e dos padrões inerentes aos dados, técnicas de aprendizado de máquina tem sido aplicadas para se obter uma predição do impacto de uma SNP (Meng e Liping 2016), (Adzhubei et al. 2010) (15). Adicionalmente, estas ferramentas permitem analisar as características das mutações de maneira rápida, auxiliando na tomada de decisão sobre a patogenicidade de um SNP.

Existem diversas ferramentas capazes de fazer a predição do efeito que as SNPs causam nos seres humanos. Muitas delas utilizam classificadores próprios para a obtenção de um score do possível dano que a variação pode causar, resultando em classes indicativas de seu grau de patogenicidade. A grande maioria dos softwares que possuem essa funcionalidade, utilizam atributos e classificadores genéricos para os genes em que são treinados, no entanto, estudos mostram (Meng e Liping 2016) que é possível aprimorar o resultado das predições utilizando-se de atributos específicos para cada gene usado como dado de treinamento do modelo, dependendo do tipo de doença, por exemplo, o gene BRCA1 alterado no fígado, não causa problemas.

Sabendo-se que muitos testes genéticos que investigam a existência de mutações a fim de avaliar as chances de um paciente desenvolver determinada doença são baseados

no sequenciamento de um painel de genes associados à patologia, o presente trabalho buscou avaliar o poder preditivo de modelos de classificação de SNPs treinados a partir de dados de painéis de genes específicos de uma doença, comparando-os com outras abordagens existentes na literatura.

1.1 Objetivo Geral

O objetivo geral deste trabalho é utilizar métodos de aprendizado de máquina para treinar modelos de predição de SNPs a partir de dados relacionados a um gene específico, a um painel de genes, e de todos os genes humanos presentes na base de dados utilizada no treinamento, comparando os resultados.

1.2 Objetivos específicos

Os objetivos específicos deste trabalho incluem:

- Implementar uma metodologia de treinamento de SNPs baseada na literatura de referência
- Testar a metodologia proposta para análise de dados relacionados ao câncer de mama
- Avaliar o impacto da seleção de atributos sobre o desempenho dos modelos..
- Comparar o desempenho obtido por modelos treinados com dados de um gene (gene específico), de um painel de genes associados a uma doença (doença-específico) e com todos os genes humanos.

1.3 Justificativa

Na literatura, existem diversas ferramentas capazes prever a patogenicidade de uma SNP. No entanto, segundo (Meng e Liping 2016), é possível aprimorar os resultados atuais customizando o treinamento dos algoritmos de aprendizado de máquina para que, ao invés de utilizarem o mesmo conjunto de atributos para todos os genes usados no treinamento, são utilizados modelos customizados de atributos.

Com isso, este trabalho propõe implementar uma metodologia capaz de construir modelos de treinamento com atributos customizados, diferenciando-se no fato de que os dados usados para o treino serão agrupados em painéis de genes, ao invés de famílias ou genes isolados, selecionando um grupo de atributos para os mesmos. Por fim, será disponibilizado todo o processo proposto durante este trabalho como um pacote para uso na linguagem *R*.

1.4 Organização do texto

O texto está dividido em 8 capítulos. O capítulo 2, introduz conceitos chaves para o entendimento do funcionamento das ferramentas de predição. Nele, serão introduzidos em mais detalhes os conceitos de aprendizado de máquina e SNP.

O capítulo 3, demonstra algumas ferramentas usadas como base teórica para a construção desse trabalho.

O capítulo 4, disserta sobre os dados que são acessados para a construção dos modelos de classificação das ferramentas estudadas e também fornece uma descrição de alguns atributos considerados pelos classificadores. Em seguida, o capítulo 5, demonstra a metodologia criada, descrevendo todo o processo de obtenção de dados até o resultado dos modelos.

O capítulo 6 apresenta e discute os resultados obtidos.

O capítulo 7 apresenta uma discussão mais detalhada sobre os resultados e levanta hipóteses sobre quais são as possíveis causas para os resultados obtidos.

Por fim, o capítulo 8, apresenta os trabalhos futuros para continuação para a pesquisa e conclui a dissertação.

2 REFERENCIAL TEÓRICO

Este capítulo descreve conceitos importantes que são necessários para o entendimento do funcionamento das ferramentas de predição de SNP. O capítulo começa com uma descrição mais detalhada sobre o que são os SNPs, partindo em seguida para o conceito de aprendizado de máquina. Dentro da seção de aprendizado de máquina, a tarefa de classificação é detalhada, pois é a técnica de aprendizado utilizada nos softwares de predição. Também são descritos algoritmos de seleção de atributos, especificamente os utilizados pela ferramenta iFish, a qual, dentro do escopo deste estudo, é a única capaz de treinar os genes com atributos específicos. É introduzido o conceito de pré-processamento, descrevendo as técnicas utilizadas para a seleção de atributos e imputação de dados.

Por fim, as métricas de avaliação de modelos de aprendizado são descritas, para que seja possível discutir o resultado obtido pela metodologia proposta.

2.1 SNPs

Os polimorfismos de nucleotídeo único SNP (*Single nucleotide polymorphism*)¹, são uma variação no genoma humano que ocorrem em uma posição única na sequência de DNA.

Uma variação somente é considerada como SNP se mais de 1% da população não carrega a mesma base nitrogenada (adenina, citosina, guanina e timina) na posição específica na sequência do DNA. Caso essa variação ocorra em um gene, a SNP pode causar uma variação na cadeia de aminoácidos como um todo. A Figura 1, ilustra 4 versões da mesma sequência de DNA, representando uma SNP nesse trecho do sequenciamento.

¹<https://www.nature.com/scitable/definition/single-nucleotide-polymorphism-snp-295>

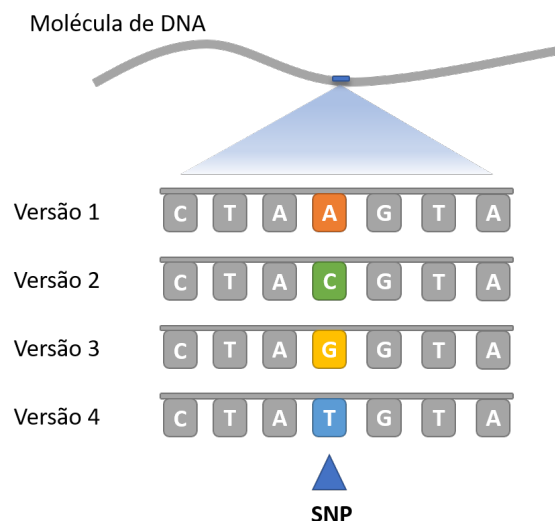


Figura 1: Exemplo de SNP em um gene presente no DNA.

Outros dois conceitos importantes para este estudo são as nsSNPs (*non-synonymous single nucleotide polymorphisms*) (Yates e Sternberg 2013), que representam as SNPs que alteram a sequência da proteína, ou seja, elas causam um impacto na síntese da proteína, enquanto que as sSNPs (*synonymous single nucleotide polymorphisms*), não alteram a sequência. E as SNVs (*single nucleotide variants*) as quais representam quaisquer variações em um nucleotídeo único, não havendo restrição de 1% da população. As ferramentas utilizadas neste estudo realizam as previsões da patogenicidade de nsSNPs. Para fins de simplificação, sempre que houverem referências à SNPs ao longo desse texto, está considerando-se tanto as nsSNPs quanto as nsSNVs.

As SNPs podem estar relacionadas à doenças, Mendelianas ou não, porém nem sempre há essa relação direta. Elas podem ser avaliadas para identificar se um indivíduo tem tendências a desenvolver certas doenças, para isso, utilizam-se ferramentas de aprendizado de máquina capazes de analisar essas informações de maneira rápida. A partir de SNPs já conhecidas, modelos de classificação são treinados pelas ferramentas, considerando diversas características das mutações, como estrutura 3D da proteína, características físico-químicas dos aminoácidos da substituição, *scores* de conservação da mutação, entre outros.

Para exemplificar a razão pelo qual uma SNP pode ou não provocar alteração no aminoácido, são utilizadas as imagens 2 e 3. Na figura 2, a mudança no DNA, da base G por T acarretou na mudança de aminoácido de Trp (triptofano) para Cys (Cisteína). Já na figura 3, a mudança no DNA, da base T por C, não acarretou na mudança do aminoácido.

TAT TGG CTA GTA CAT
 Tyr Trp Leu Val His

TAT TGT CTA GTA CAT
 Tyr Cys Leu Val His

Figura 2: Exemplo de mutação com troca de aminoácido

TAT TGG CTA GTA CAT
 Tyr Trp Leu Val His

TAC TGG CTA GTA CAT
 Tyr Trp Leu Val His

Figura 3: Exemplo de mutação sem troca de aminoácido

2.2 Aprendizado de máquina

Para que se possa entender o processo de analisar as SNPs e prever o seu impacto em um ser humano, é preciso apresentar o conceito de aprendizado de máquina.

Aprendizado de máquina do inglês *Machine Learning*, consiste no processo de criação de modelos estatísticos de forma automatizada sem a necessidade de instruções diretas à máquina. Onde os modelos são inferidos utilizando algoritmos capazes de aprenderem os padrões presentes nos dados usados como entrada. As técnicas de aprendizado de máquina, fazem parte do ramo de inteligência artificial. (Pierre e Soren 2002)

Existem dois tipos de aprendizados para os algoritmos, o aprendizado supervisionado e o não supervisionado. O aprendizado supervisionado consiste em aplicar algoritmos de aprendizado em dados de entrada onde já se conhecem as saídas esperadas.

As principais técnicas de aprendizado supervisionado são a classificação e a regressão. Como exemplo de classificação, sabe-se que uma determinada base de dados contém SNPs patogênicas, logo, um algoritmo de aprendizado pode utilizar essa informação para criar um modelo que identifique algum padrão nos atributos das SNPs apresentadas como entrada, onde futuramente, pode-se apenas inserir um novo SNP e esse modelo irá prever o impacto do mesmo.

No aprendizado não supervisionado, não se têm conhecimento do resultado desejado, logo, os algoritmos apresentam as semelhanças que as entradas possuem com o objetivo de encontrar algum padrão ideal para interpretação das informações. As principais técnicas de aprendizado não supervisionado são o agrupamento e a associação.

Entre as diversas técnicas de aprendizado de máquina com diferentes algoritmos capazes de realizar as tarefas desejadas, este trabalho foca na tarefa de classificação. A classificação, técnica de aprendizado supervisionado, consiste em atribuir uma classe à um determinado conjunto de dados. Por exemplo, temos uma substituição de bases G por A no cromossomo 1 na posição 19199400, ao inserir esses dados em um modelo classificador, obteremos como resposta uma classe, neste caso, existem duas, patogênico ou não patogênico. O sistema nos responde qual das duas classes essa substituição pertence e um score representando o grau de pertinência dessa mutação à classe em que se encontra.

Alguns termos são essenciais para o entendimento de aprendizado de máquina e são mencionados em diversos momentos deste texto. As seções a seguir definem esses conceitos, onde exemplos são apresentados no contexto deste trabalho.

2.2.1 Atributos

O primeiro conceito importante é o de atributo. Um atributo de uma base de dados, é representado por uma coluna, eles são as características que cada entrada no banco de dados possui. Por exemplo, a figura 4 apresenta exemplos de atributos utilizados pela ferramenta *iFish*. Nela são mostrados alguns dos atributos utilizados no treinamento juntamente com o atributo alvo *impact*.

2.2.2 Instâncias ou entradas

Instâncias ou entradas, representam cada linha no banco, ou seja, cada vez que um novo dado é inserido no banco de dados, temos uma nova entrada com os valores de seus atributos. A Figura 4, mostra um exemplo de entrada para o classificador da ferramenta *iFish*. Nela é possível visualizar os valores de 9 atributos para uma entrada.

CpG	IdpMax	IdQmin	Weight	Bulkiness	impact
1	14.4970	86.5000	16.0500	3.9700	pathogenic

Figura 4: Exemplo de uma entrada de um classificador

2.2.3 Atributo alvo

Para a tarefa de classificação, temos o conceito de atributo alvo, que consiste no atributo que deseja-se prever o valor, com base na combinação dos valores dos outros atributos presentes na base de dados. É neste ponto que os algoritmos de classificação montam seus modelos e treinam com informações dos conjuntos de treinamento.

A figura 4, o atributo alvo é a coluna *impact*. Neste trabalho, utiliza-se técnicas de classificação, onde o atributo alvo é sempre conhecido no conjunto de treinamento.

2.2.4 Dados de treinamento e teste

Por fim, temos os dados de treinamento e os dados de teste. Os dados de treinamento são utilizados para a montagem do modelo de classificação, onde a saída necessária (valor do atributo alvo) é conhecida. Já os dados de teste, são utilizados para avaliação dos modelos, suas saídas também são conhecidas, porém, na etapa dos testes o modelo tenta inferir qual a classe correta para uma determinada entrada.

2.3 Pré-processamento

A etapa do pré-processamento dos dados, consiste em aplicar técnicas para que se possam filtrar os dados de maneira que elimine-se dados inconsistentes ou redundantes. Neste trabalho, foram utilizadas duas técnicas de pré-processamento para que fosse possível obter os resultados desejados. A primeira delas é a seleção de atributos, seguida pela imputação de dados.

2.3.1 Seleção de atributos

A seleção de atributos consiste em aplicar algoritmos capazes de combinar os atributos de uma base de dados de maneira otimizada. Ou seja, os algoritmos selecionam um grupo específico de atributos que tem uma maior relevância quando agrupados entre si.

Existem diferentes categorias de métodos de seleção de atributos. Dentre elas, existem os métodos chamados de *Wrappers*. Neles, é executado um algoritmo de aprendizado de máquina em um subconjunto de atributos selecionados pelo métodos e avalia-se a taxa de erro do modelo gerado montando-se uma pontuação para comparar esse modelo com outros gerados em outros subconjuntos. Ao final da execução, o conjunto de atributos com a melhor pontuação é selecionado para o treinamento.

Neste projeto, utilizou-se a metodologia *Rfe* (*Recursive Feature Elimination*) (Ambroise e McLachlan 2002), a qual é uma forma de implementação de um *Wrapper*.

2.3.1.1 RFE

O *RFE* é um *Wrapper* que testa diversas combinações de atributos e treina um conjunto de dados com esses atributos ranqueando-os com base na sua importância.

A técnica é definida pelo algoritmo 1

Primeiramente, o algoritmo realiza a predição dos dados com todos os atributos do conjunto. Em seguida, é avaliada a performance do modelo juntamente com o rank do mesmo. Esse rank pode ser calculado baseado em diferentes métricas, durante esse trabalho, foram utilizados dois métodos de seleção de modelos AIC e BIC.

Com as informações computadas para o conjunto original, são executadas iterações sobre o conjunto de atributos, começando em S_i e terminando em S , onde S é o parâmetro de entrada que delimita o tamanho do subconjunto máximo a ser calculado e i é a iteração atual do algoritmo. São mantidos os i atributos mais importantes (que possuem a maior pontuação no rank) e o modelo é treinado usando-os. O ranqueamento é recalculado para os atributos selecionados, continuando o processo até que sejam testados todos os subconjuntos S de melhores atributos (Ambroise e McLachlan 2002) (Svetnik et al. 2004).

Por fim, é comparada a performance de todos os modelos gerados em cada iteração, resultando no número apropriado de atributos e qual é o conjunto ótimo de atributos.

Algoritmo 1: *Recursive Feature Elimination*

Entrada: S

Saída: Subconjunto ótimo de atributos do conjunto de dados

```

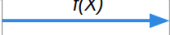
1  início
2  Treina o modelo usando todos os atributos
3  Calcula a performance do modelo
4  Calcula o ranking dos atributos
5  para Cada subconjunto de tamanho  $S_i, i = 1...S$  faça
6      Mantenha os  $S_i$  atributos mais importantes
7      Treina o modelo utilizando os  $S_i$  atributos
8      Calcula a performance do modelo
9      Recalcula o ranking de atributos
10 fim
11 Compara performance de todos  $S_i$  subconjuntos
12 Determina o número apropriado de atributos
13 Retorna o conjunto ótimo de atributos
14 fim
```

2.3.2 Imputação de dados

Outra técnica de pré-processamento utilizada é a imputação de dados. Este método consiste em utilizar técnicas para inserir dados no conjunto de treinamento onde há informações faltantes. Neste trabalho, foi necessária a utilização desta técnica, devido à necessidade de os dados estarem completos para que os algoritmos de aprendizado pudessem ser executados. Utilizando o pacote *Mice* da linguagem R, utilizou-se o método *cart* para realizar a imputação.

A imputação é utilizada quando há dados faltantes no conjunto de dados de treinamento. Esses dados são identificados e aplicam-se técnicas matemáticas para inferir os valores faltantes. A figura 5, mostra uma tabela com dados genéricos, onde existem dados faltantes. Esses dados são representados pela sigla NaN (*Not a Number*). Ao aplicar-se a equação matemática $f(x)$ nos dados, obtêm-se uma estimativa dos valores que devem

estar presentes nessas colunas, baseando-se nos valores já existentes das colunas que não possuem dados faltantes.



	Col 1	Col 2	Col 3
0	NaN	2	1.6
1	0	NaN	2.7

	Col 1	Col 2	Col 3
0	1	2	1.6
1	0	1.6	2.7

Figura 5: Exemplo de imputação de dados

2.4 Aprendizado supervisionado

Esta seção explica os algoritmos de classificação utilizados por algumas das ferramentas estudadas neste trabalho e também algoritmos usados na metodologia proposta.

2.4.1 SVM

Support Vector Machines ou SVM (Ray 2017) (Cortes e Vapnik 1995) é um algoritmo utilizado para classificação e regressão. Seu funcionamento baseia-se na utilização de gráficos para plotar os dados. Cada entrada no banco de dados, é plotada em n dimensões, onde n é igual ao número de atributos. A ideia principal desta técnica é encontrar um plano que divida os dados em classes da melhor maneira possível.

Na figura 6, é possível visualizar essa divisão de planos. Cada ponto no gráfico representa uma entrada na base de dados com dois atributos x e y e existem três possíveis planos para a divisão desses dados, os planos A, B e C. Pela imagem é possível identificar que o plano B seria o mais correto para a tarefa de classificação, pois divide os dados completamente em duas classes. Em uma situação realista, possivelmente essa divisão perfeita não ocorre, pois existem *outliers* nos dados. Os *outliers* são entradas fora do padrão seguido pela maioria das instâncias, levando a algumas inconsistências na hora de classificar. Existem técnicas de remoção de *outliers* para melhorar o desempenho dos algoritmos de classificação, porém elas não estão dentro do escopo deste estudo.

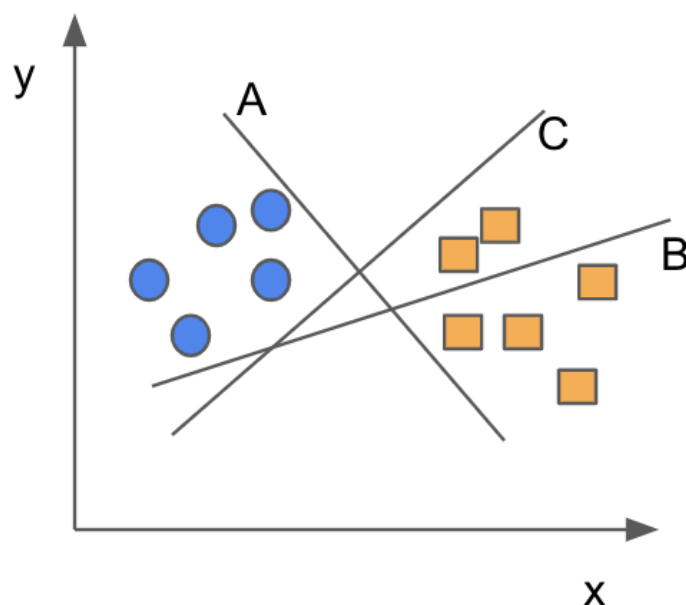


Figura 6: Exemplo de possíveis divisões de planos para o algoritmo SVM (Ray 2017)

O algoritmo SVM é utilizado pela ferramenta *iFish*, descrita na seção 3.2.1.

2.4.2 Random Forest

O algoritmo *Random Forest*, é utilizado tanto para classificação quanto para regressão. O seu funcionamento consiste em uma combinação de outros algoritmos para classificar os dados.

Utiliza-se as técnicas de treinamento de árvores de decisão e *bagging* para a montagem do modelo. Primeiramente, o algoritmo utiliza a técnica de *bagging* para selecionar uma amostra aleatória dos dados de treinamento, treinando árvores de decisão para cada uma dessas amostras. É calculada a média de todas as predições realizadas por cada uma das árvores geradas em cada amostra para geração do modelo final, onde novos dados podem ser testados (Ho 1998).

2.5 Seleção de modelos

Esta seção faz uma breve descrição dos algoritmos utilizados pela ferramenta *iFish* para seleção de atributos. A ideia principal por trás destas técnicas é de avaliar a relação que os atributos têm entre si, e quantificar isso de uma maneira que atributos que não influenciam de maneira impactante no resultado da classificação, sejam descartados no momento do treinamento do modelo para uma determinada instância.

Neste trabalho, utilizou-se as técnicas de seleção de modelos para realizar a etapa de

ranqueamento do algoritmo RFE, ou seja, o RFE é responsável por selecionar os subconjuntos de atributos de cada iteração de sua execução, e os dois métodos descritos abaixo são responsáveis por calcular o ranking dos atributos.

Isso acontece no momento em que o algoritmo do RFE faz o ranqueamento e passa como entrada o subconjunto de atributos atual de tamanho n . Quando essa entrada é submetida, a customização feita para o RFE itera por todos esses atributos e calcula o score BIC ou AIC para ele em relação a todos os outros atributos que estão na entrada. Após isso, um ranking de atributos é montado nessa iteração baseado no valor do score BIC ou AIC e é retornado para o algoritmo dar procedimento ao restante de sua execução.

2.5.1 AIC

A metodologia AIC (*Akaike information criterion*) (Akaike 1973) apresenta uma estimativa de informações perdidas quando um certo modelo é utilizado em relação a outros modelos. Neste estudo, o AIC pode-se aplicar para prever o quanto um atributo influencia nos resultados de uma predição, tornando possível a seleção dos atributos com maior impacto nos resultados. O modelo do AIC é representado pela equação 1

$$AIC = 2k - 2\ln(\hat{L}) \quad (1)$$

Onde:

- $\hat{L}$: função de semelhança máxima de um modelo;
- k : número de parâmetros no modelo;

Na metodologia proposta, a função de semelhança é construída por um modelo linear dos atributos selecionados para a iteração corrente do RFE. Sendo que para cada atributos presente na iteração, o seu score AIC é calculado individualmente para validar a sua influência nos resultados da predição. O mesmo é válido para a sessão a seguir, onde o método BIC é descrito.

2.5.2 BIC

O algoritmo BIC (*Bayesian Information criterion*) (Schwarz 1978) implementa uma função de semelhança para identificar a relação entre modelos distintos. A equação do modelo BIC é baseada no modelo AIC (*Akaike information criterion*). Em ambos os casos, os modelos melhores serão aqueles que possuem o menor valor BIC ou AIC. A equação 2 representa a forma de se encontrar o BIC para cada modelo.

$$BIC = \ln(n)k - 2\ln(\hat{L}). \quad (2)$$

Onde:

- $\hat{L}$: função de semelhança máxima de um modelo;
- k : número de parâmetros no modelo;
- n : tamanho dos dados de entrada.

2.6 Avaliação de modelos

Para a comparação de resultados de classificadores, são necessárias algumas métricas estatísticas para a comparação dos mesmos. As seções seguintes descrevem algumas dessas medidas. Todas as definições a seguir foram extraídas de (Glossary of Terms 1998), exceto nos casos onde a referências está especificada.

2.6.1 Validação cruzada

Dentro das tarefas de aprendizado de máquina, pode-se avaliar melhor a acurácia de um modelo usando a técnica de validação cruzada. Divide-se os dados em n subconjuntos de tamanho aproximados, onde a cada iteração da etapa de treinamento do modelo, é utilizada $n - 1$ partes do conjunto de dados para o treino e o subconjunto restante serve o propósito de testes do modelo. Ao fim das iterações de treinamento, é feita uma média dos valores das medidas de avaliação dos modelos, resultando em todas as métricas desejadas pelo usuário (Acurácia, Recall, Especificidade).

2.6.2 Treinamento de um modelo

A etapa de treinamento em um sistema de aprendizado de máquinas, consiste em executar o algoritmo de aprendizado, seja ele de classificação ou de alguma outra técnica, em uma base de dados de entrada para que ele possa inferir o modelo que melhor representa os valores fornecidos.

2.6.3 Teste de um modelo

Após a etapa de treinamento, podem ser realizados testes para validar se o modelo resultante da etapa de treinamento é consistente. Dados de entrada são fornecidos para o sistema, que produz um resultado baseado nos mesmos padrões identificados nos dados de teste.

2.6.4 Matriz de confusão

A matriz de confusão de um modelo, é a representação dos valores previstos pelo modelo em relação aos valores que deveriam ter sido previstos, mais especificamente, a matriz mostra quantas instâncias foram classificadas corretamente e incorretamente para cada uma das classes existentes. A matriz possui tamanho $N \times N$, onde N é a quantidade de classes presentes no resultado do modelo. A tabela 1 mostra um exemplo de matriz

de confusão. Neste exemplo, interpreta-se que de 12 instâncias que deveriam ser classificadas como patogênicas, 2 foram classificadas incorretamente. O mesmo vale para as instâncias não patogênicas, onde 4 delas foram classificadas incorretamente.

Tabela 1: Matriz de confusão para um modelo de predição de SNPs

	Patogênico	Não patogênico
Patogênico	10	2
Não patogênico	4	15

2.6.5 Acurácia

A acurácia de um modelo representa a proporção de acertos (classificações corretas) sobre o número total de instâncias presentes na base de dados. Utilizando-se a matriz da tabela 1, podemos obter a acurácia do classificador que produziu estes resultados aplicando a seguinte equação.

$$Acc = \frac{TP + FN}{TP + TN + FN + FP} = \frac{(10 + 15)}{(10 + 2 + 4 + 15)} = 0,806 \quad (3)$$

Onde:

- TP: Verdadeiros positivos.
- TN: Verdadeiros negativos.
- FP: Falsos positivos.
- FN: Falsos negativos.

2.6.6 Recall ou Sensibilidade

Esta medida indica quantas classificações corretas foram feitas sobre o total de classificações para uma determinada classe. Também conhecido como medidas de verdadeiros positivos de um sistema. A equação abaixo mostra o cálculo do Recall para a matriz da tabela 1.

$$Recall = \frac{TP}{TN + TP} = \frac{(10)}{(10 + 2)} = 0,833 \quad (4)$$

2.6.7 Especificidade

Ao contrário da sensibilidade, a especificidade mede a capacidade do sistema identificar, corretamente, que uma instância realmente não deve ser classificado em determinada classe. Também conhecida como medidas de verdadeiros negativos de um sistema. A equação abaixo mostra o cálculo da especificidade para a matriz da tabela 1.

$$Especificidade = \frac{TN}{FN + TN} = \frac{(15)}{(4 + 15)} = 0,789 \quad (5)$$

2.6.8 Curva ROC

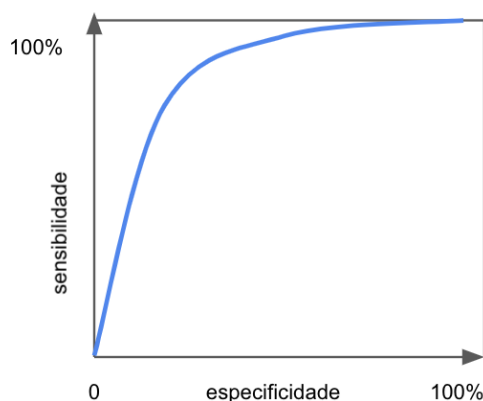


Figura 7: Exemplo de uma curva ROC.

A curva ROC (*Receiver operating characteristic*) (6), é a representação gráfica da sensibilidade pela especificidade. Sua função é avaliar a capacidade de diagnóstico de um classificador. A Figura 7 mostra um exemplo de uma curva ROC². Quanto mais perto do canto superior esquerdo a curva estiver, maior é a acurácia do modelo em questão.

2.7 Amostragem

O processo de amostragem na estatística, consiste em extrair partes (uma amostra) de um conjunto de dados para análise. Podendo ocorrer de maneira aleatória.

No campo da ciência de dados, a amostragem pode acontecer de duas maneiras quando necessária para resolver problemas de desbalanceamento de classes. O *oversampling* e o *undersampling* (16), consistem em métodos para ajustar a distribuição de uma classe quando há um desbalanceamento muito grande. Por exemplo, em uma base de 1000 entradas, possuímos 99% das instâncias de uma classe A, enquanto que apenas 1% das instâncias representam a outra classe B. Neste caso, poderia ser aplicado o *undersampling* para reduzir o número de instâncias da classe A para chegar ao número de instâncias da classe B ou poderia ser feito o processo reverso, o *oversampling* nas instâncias da classe B.

Neste trabalho, utiliza-se o *undersampling* para balancear as classes no treinamento dos modelos. É utilizado o método aleatório disponibilizado no pacote *caret* do R.

²<https://www.medcalc.org/manual/roc-curves.php>

3 TRABALHOS RELACIONADOS

Neste capítulo serão descritas algumas das principais ferramentas disponíveis para predição de SNPs, com duas subseções, uma para as ferramentas chamadas de genéricas (onde os atributos escolhidos para treinamento não são específicos para cada gene) e uma seção para as específicas, onde há o processo de especialização do classificador para genes ou grupos de genes.

3.1 Ferramentas de predição genéricas

3.1.1 PolyPhen2

O PolyPhen2 (Adzhubei et al. 2010) (2) é uma software desenvolvido na Universidade de Harvard com o objetivo de prever o impacto de SNPs na funcionalidade das proteínas humanas. A aplicação é disponibilizada via web e também é possível baixar a versão para desktop. O programa se baseia em um classificador *Naïve Bayes* para a predição de SNPs patogênicas,

3.1.1.1 Metodologia

O PolyPhen utiliza uma série de atributos obtidos de bancos de dados externos e alguns calculados pela própria ferramenta. Para treinar o seu classificador, o PolyPhen2 utiliza duas bases de dados distintas: HumDiv e HumVar. A primeira base contém dados sobre todos alelos conhecidos, os quais são causadores de doenças hereditárias em humanos, enquanto a segunda fonte de dados apresenta mutações causadoras de doenças conhecidas pela base de dados UniProtKB. Durante utilização da ferramenta, é possível selecionar qual classificador deseja-se utilizar, baseando-se em qual dado de treinamento foi utilizado. Todos os atributos são buscados e calculados para a aplicação do algoritmo de classificação. A ferramenta então calcula a probabilidade da mutação ser prejudicial ao ser humano, juntamente com uma classificação da mesma em "benigna", "possivelmente patogênica" e "provavelmente patogênica".

3.1.2 Mutation Assessor

O Mutation Assessor, é uma ferramenta online construída para prever o efeito das mutações, especialmente para mutações causadoras de câncer. O seu algoritmo funciona baseando-se na conservação evolucionária dos aminoácidos afetados pela variação.

A ferramenta busca os dados do bancos de dados COSMIC e constrói um modelo de clusterização, explorando a conservação evolucionária das famílias de proteína.

3.1.3 Sift e Provean

As ferramentas Sift e Provean preveem o efeito causado na função de uma proteína por uma substituição em um aminoácido. Ambas as ferramentas utilizam algoritmos próprios para o cálculo do score de predição (algoritmos Sift e Provean).

3.1.3.1 Algoritmo Sift

O algoritmo Sift (15) calcula a probabilidade de uma variação na sequência de uma proteína afetar o seu funcionamento. Eles baseia seu cálculo na suposição de que posições importantes da proteína, são conservadas durante a evolução, interpretando que qualquer substituição nessas posições conservadas, afetam a funcionalidade da proteína.

Primeiramente, a ferramenta busca dados de proteínas relacionadas às variantes de entrada na base de dados. Em seguida, o algoritmo constrói um alinhamento nas sequências e calcula o score de conservação para cada posição da sequência. Com base nesses valores, a ferramenta indica se a SNP é patogênica ou não. .

3.1.3.2 Algoritmo Provean

O algoritmo Provean (Choi et al. 2012), utiliza métodos de clusterização para prever a patogenicidade de uma SNP. A ferramenta para sua execução na web, executa em paralelo o algoritmo Sift, mostrando os resultados de ambas metodologias.

Para treinar o seu classificador, o algoritmo Provean coleta informações sobre as sequências no banco de dados de proteínas NCBI. Em seguida, as sequências obtidas na base de dados são agrupadas em clusters com semelhança de 80%. São formados um total de 45 clusters. A partir daí, são calculados scores para cada sequência presente nos clusters utilizando a matriz BLOSUM62. O score final da ferramenta, é a média dos scores para cada sequência, predizendo se uma variação é maléfica ou não.

3.2 Ferramentas de predição específicas

3.2.1 iFish

O *iFish* (Meng e Liping 2016) é uma ferramenta desenvolvida na Universidade de Pequim. Seu objetivo é a predição do impacto de SNPs no genoma humano utilizando um

classificador treinado especificamente para um determinado gene ou família de genes. O programa foi desenvolvido em uma plataforma inteiramente web, e está disponível para uso em ¹.

O *iFish* possui 40 atributos candidatos para cada gene e família de genes, esses atributos, descritos na seção 4.2 são selecionados a partir de algoritmos de seleção, no caso desta ferramenta, são utilizados o AIC, BIC e o LASSO. Com os atributos selecionados para cada um dos genes, um classificador SVM é treinado para cada um dos modelos. A tabela 2 mostra um exemplo de atributos selecionados para o gene A1BG (anotação baseada nos dados da base de sequenciamento hg19).

Atributo
Polyphen2_HDIV
Polyphen2_HVAR
MutationAssessor
GERP_RS
PhyloP
dScore
Score1
CpG
Flexibility
TurnTend
BLOSUM62

Tabela 2: Atributos utilizados para o treinamento do classificador para o gene A1BG

Atualmente, a ferramenta treinou modelos para 254 genes diferentes e para 152 famílias de genes, totalizando 5731 genes. Também é possível utilizar a ferramenta para treinamento de modelos customizados, para que isso seja possível, é necessário fornecer os dados de treinamento para a ferramenta.

¹<http://ifish.cbi.pku.edu.cn/submit.html>

4 BASES DE DADOS, DADOS E ATRIBUTOS

Diversas bases de dados distintas são consultadas pelas ferramentas de predição de variantes para a extração dos valores dos atributos utilizados pelos seus algoritmos de classificação de SNPs. Esta seção está dividida em duas partes, a primeira, descreve todas as bases utilizadas nessas consultas, enquanto que a segunda parte, descreve os atributos conhecidos presentes nas ferramentas e quais atributos estão presentes em cada uma delas.

4.1 Bases de dados

Diversos bancos de dados são consultados pelas ferramentas de predição em diferentes momentos. Para que seja possível construir o modelo de classificação, as bases de dados são consultadas para a extração dos atributos necessários para o treinamento do modelo. Após obter o modelo resultante, extrai-se novos dados que não foram utilizados pelo treinamento, e testa-se o modelo.

Um exemplo que ilustra esse comportamento de busca dos atributos tanto para o treinamento quanto para a execução do modelo, é a ferramenta *iFish*. Nela, há a busca de informações nas bases de dados *HGMD*, *1000 Genome* e *dbSNP* para montar o seu conjunto de treinamento. Após o classificador estar treinado, as mesmas bases de dados são consultadas no momento em que o usuário fornece suas entradas para teste.

A subseções a seguir, descrevem as bases de dados utilizadas pelas ferramentas mostradas no capítulo 3. Sendo elas:

- *UCSC Genome Browser*
- *Cosmic*
- *UniProt*
- *HGMD*
- *1000 Genome Project*
- *NCBI*

Até o presente momento, este trabalho utiliza os dados fornecidos pelo *NCBI*, mais especificamente o banco de dados *clinVar*.

4.1.1 UCSC Genome Browser

O *UCSC Genome Browser* (Kent et al. 2002) é um banco de dados público para fins acadêmicos disponibilizado pela Universidade da Califórnia Santa Cruz. Seus dados consistem de informações sobre o genoma, com informações sobre variantes e alinhamentos de sequências. Os dados estão disponíveis em ¹.

Os dados podem ser acessados via *ftp*, acessando o banco de dados MySQL ou baixando as versões do genoma desejadas.

Também é possível acessar as informações de anotação das variantes utilizando a ferramenta *Variant Annotation Integrator* presente no *UCSC Genome Browser*. Nela podem ser encontradas variantes de diversas espécies, com anotações dos resultados das ferramentas de predição.

4.1.2 Cosmic

A base de dados Cosmic (Forbes et al. 2017) contém informações sobre mutações relacionadas ao câncer. As buscas podem ser feitas por gene, tipo de câncer e pela mutação desejada.

Os dados estão disponíveis em ² de forma pública para fins acadêmicos. Após registro na plataforma, é possível baixar os arquivos de dados aplicando filtros de nome do gene, nome da amostra ou tipo de tecido, possibilitando o *download* de arquivos únicos. Caso seja necessário baixar a informação de múltiplos genes, pode-se acessar os dados através do sistema *SFTP* disponibilizado pela ferramenta.

4.1.3 UniProt

O *UniProt* (2017) consiste em um banco de dados com informações sobre as sequências de proteínas e suas funcionalidades. É possível acessar os dados em ³. As ferramentas de predição estudadas nesse trabalho, utilizam a base *Swiss-Prot* para obtenção de informações sobre as proteínas codificadas pelos genes em que deseja-se analisar a variante. Essa base possui 556.196 entradas e está disponível para *download* nos formatos *xml*, *fasta* e *txt* no site da ferramenta.

Também é possível acessar os dados utilizando a ferramenta de busca do próprio site do *UniProt*, onde pode-se filtrar as proteínas por espécie. É possível acessar os dados via *Web Service*, onde é disponibilizada uma *REST API* para aceitar requisições externas às diversas funcionalidades presentes na ferramenta ⁴.

¹<https://genome.ucsc.edu/index.html>

²<http://cancer.sanger.ac.uk/cosmic>

³<http://www.uniprot.org/>

⁴<http://www.uniprot.org/help/api>

4.1.4 HGMD

A *HGMD* (Human Gene Mutation Database) (Stenson et al. 2014) é um banco de dados que reúne todas as versões publicadas de alterações genômicas responsáveis por doenças hereditárias em humanos. Por tratar-se de um banco de dados privado, algumas informações não estão disponíveis abertamente, é necessária a compra da licença de uso da ferramenta. Parte dessas informações é disponibilizada para uso acadêmico, contabilizando um total de 148.399 entradas públicas e um total de 214.158 entradas para a versão com licenciamento.

Acessando ⁵, é possível fazer o download dos dados ou realizar a busca do gene desejado utilizando a ferramenta de busca disponível na plataforma, sendo necessário o cadastro no *site* para ter o acesso às informações.

4.1.5 1000 Genome Project

O *1000 Genome Project* (Consortium 2015) é o maior catálogo público de variações genéticas em humanos. Os seus dados incluem variações com no mínimo 1% da população na qual as coletas foram feitas. Os dados podem ser baixado via *ftp* e também podem ser acessados utilizando a ferramenta *Genome Browser* presente no *site*. Pode-se acessar as bases de dados em ⁶

4.1.6 NCBI

O *NCBI* (*The National Center for Biotechnology*), é responsável por fornecer acesso a diversas bases de dados distintas, com informações genômicas de diferentes espécies. Todos os dados estão disponíveis para acesso, via o *website* da ferramenta e também através de *download* via *ftp* das bases de dados. As ferramentas deste trabalho utilizam duas bases específicas do *NCBI*, a *dbSNP*, onde são buscadas variantes com resultados conhecidos para treinamento dos classificadores e a base *RefSeq* para informações de anotações de genoma, identificação e caracterização de genes.

Durante a aplicação da metodologia proposta neste trabalho, utilizou-se o banco de dados *clinVar*, disponibilizado pelo *NCBI*. O principal motivo da escolha deste banco, é o fato de o mesmo possuir a informação da patogenicidade da SNP, que é o atributo alvo necessário para o treinamento dos modelos, e por ser uma base pública.

4.2 Atributos

Embora as ferramentas de predição produzam resultados semelhantes, elas trabalham com atributos distintos para o treinamento e obtenção de seus modelos. Esta seção descreve atributos apresentados nas ferramentas *iFish* e *PolyPhen2*.

⁵<http://www.hgmd.cf.ac.uk/ac/index.php>

⁶<http://www.internationalgenome.org/>

4.2.1 Scores de conservação

Esses scores representam a conservação da sequência durante a evolução. Essas sequências conservadas são representadas por sequências similares ou idênticas entre espécies ou moléculas diferentes produzidas pela mesma espécie durante a evolução. Quanto maior o *score* de conservação, mais tempo determinada sequência se manteve intacta no processo evolutivo. Dois atributos são utilizados pelas ferramentas para avaliar o score de conservação: o *GERP* e o *PhyloP*. A tabela 3 demonstra os dois atributos de conservação e quais ferramentas utilizam os mesmos.

Tabela 3: Scores de conservação utilizados como atributos pelas ferramentas de predição

Score de Conservação	Descrição	iFish	PolyPhen2
GERP	Score de conservação GERP (Cooper et al. 2005) para a mutação	✓	✓
PhyloP	Score de conservação PhyloP (Pollard et al. 2010)	✓	✓

4.2.2 Atributos relacionados a propriedades das sequências

Estes atributos descrevem as regiões de interesses na sequência da proteína. São representadas pelas proteínas codificadas pelos genes utilizados como entrada no sistema. O atributo *Region* descreve características da região da substituição, como por exemplo, o papel da região em mediar interações entre proteínas ou descrever características da região contendo repetições. O atributo *SITE*, representa qualquer local de aminoácido único interessantes para a substituição, como por exemplo, áreas de clivagem ⁷. Alguns atributos mostrados na tabela 4 utilizam os *scores* PSIC (Sunyaev et al. 1999) para determinação de seus valores. Esses *scores* representam o peso da sequência em análise balanceando as informações observadas com sequências similares em alinhamentos múltiplos.

⁷<http://www.uniprot.org/help/site>

Tabela 4: Atributos relacionados a propriedades das sequências de aminoácidos

Atributo	Descrição	iFish	PolyPhen2
SITE	Anotação SITE para a sequência	✓	✓
Region	Região em que ocorre a substituição		✓
Coiled	Indica se as substituições realizadas na proteína, são realizadas em regiões de alças e voltas	✓	✓
Compbias	Indica se a substituição ocorre na região {compositional bias region} da proteína	✓	✓
Intramem	Indica que a substituição ocorre na região intramembrana	✓	✓
Propep	Refere-se à região da proteína clivada após a maturação ou ativação.	✓	✓
Repeat	Substituição nas regiões em que ocorrem repetições nos {motifs} de sequências ou domínios proteicos.	✓	✓
Signal	Região de {signal peptides} destinados ao retículo endoplasmático.	✓	✓
Transmem	Indica se a substituição ocorre na região transmembrana.	✓	✓
CpG	Indica se a substituição altera o contexto CpG, os quais são regiões do DNA onde uma Guanina é precedida por uma Citosina.	✓	✓
dScore	Representa a diferença entre o score <i>PSIC(position-specific-independent counts)</i> (Sunyaev et al. 1999) para a substituição de entrada	✓	✓
Score1	Informa o PSIC de aminoácidos selvagens.	✓	✓
dScore2	Representa o score PSIC para aminoácidos mutantes.		✓
Nobs	É o número de aminoácidos observados nos locais de substituição em que há um alinhamento de múltiplas sequências	✓	✓
IdPMax	Congruência máxima do aminoácido mutante em alinhamento de múltiplas sequências.	✓	✓
IdQMin	Representa a identidade da sequência com o homólogo mais próximo.	✓	✓
PHAT	Elemento da matriz PHAT para substituições na região TRANSMEM.		✓

4.2.3 Atributos da estrutura 3D da proteína

Os atributos representados na Tabela 5 seguir descrevem algumas propriedades da estrutura 3D da proteína em que ocorre a substituição.

Tabela 5: Atributos da estrutura 3D da proteína

Atributo	Descrição	iFish	PolyPhen2
Helix	indica se a substituição acontece na estrutura secundária de hélice	✓	
Sheet	a substituição acontece na estrutura secundária de folha da proteína	✓	
Turn	substituição ocorre na estrutura secundária de voltas da proteína	✓	
dVol	Indica se há mudanças no volume da cadeia de resíduos	✓	
B-Fact	representa o B-Fact normalizado para a posição da substituição	✓	
NormASA	contabiliza a área acessível ao solvente normalizada	✓	
MapReg	região do mapa phi-psi (21), que representa os ângulos diédricos de aminoácidos que compõem a proteína.	✓	

4.2.4 Mudanças físico-químicas dos aminoácidos

A tabela 6 demonstra os atributos que representam as mudanças físico-químicas dos aminoácidos.

Tabela 6: Mudanças físico-químicas dos aminoácidos

Atributo	Descrição	iFish	PolyPhen2
Weight	representa o quanto foi mudado o peso do resíduo molecular no momento da substituição	✓	
Volume	Alteração no volume de resíduo na substituição	✓	
Polarity	Mudança na polaridade do resíduo na substituição	✓	
Hydrophathicity	Mudança na hidrofília do resíduo na substituição	✓	
Hydrophobicity	Alteração na hidrofobia do resíduo molecular após a substituição	✓	
TransmemTend	Alteração na tendência transmembrantar do resíduo na substituição	✓	
Flexibility	Representa alterações na flexibilidade do resíduo na substituição	✓	
Helix tendency	Representa mudanças na tendência helicoidal da proteína	✓	
Sheet tendency	É a mudança na tendência ao formato folha da estrutura secundária da proteína	✓	
Turn tendency	Representa mudanças na tendência da proteína alterar a direção da cadeia polipeptídica	✓	
Coil tendency	Tendência do resíduo estar no formato de mola na substituição	✓	
BLOSUM62	<i>Blocks of Amino Acid Substitution Matrix</i> é a matriz de substituição utilizada para alinhamento de sequências de proteínas	✓	

4.2.5 Anotações das ligações da proteína

Esses atributos correspondem aos tipos de ligações que a proteína codificada pelo gene em análise pode realizar. A Tabela 7, descreve esses atributos.

Tabela 7: Atributos relacionados às ligações da proteína

Atributo	Descrição	iFish	PolyPhen2
DISULFID	Ponte dissulfeto		✓
CROSSLN	Ligação covalente entre as proteínas		✓

4.2.6 Anotações das regiões funcionais

Campos referentes as anotações de regiões de funcionalidade das proteínas, são representados por 7 campos distintos: BINDING (região de ligação para qualquer grupo químico), ACT_SITE(região ativa da enzima), METAL(região de ligação para um íon metálico), LIPID (resíduo lipídico), CARBOHYD (resíduo glicosilado), MOD_RES (outras

modificações covalentes) e NON_STD(aminoácido não padrão). A Tabela 8 demonstra esses atributos e quais ferramentas os utiliza.

Tabela 8: Atributos referentes às anotações das regiões funcionais da proteína

Atributo	Descrição	iFish	PolyPhen2
BINDING	Região de ligação para qualquer grupo químico		✓
ACT_SITE	Região ativa da enzima		✓
METAL	Região de ligação para um íon metálico		✓
LIPID	Resíduo lipídico		✓
CARBOHYD	Resíduo glicosilado		✓
MOD_RES	Outras modificações covalentes		✓
NON_STD	Aminoácido não padrão		✓

4.2.7 Atributos especiais

Algumas das ferramentas estudadas neste trabalho, apresentam atributos especiais para o treinamento de seus classificadores. O primeiro caso, é o da ferramenta iFish, nela, 4 atributos representam scores dos classificadores de outras ferramentas. Com a utilização desses scores na fase de treinamento, a ferramenta aprimora os resultados do seu classificador. Os scores utilizados são das ferramentas SIFT, MutationAssessor e PolyPhen2 onde, na ferramenta PolyPhen2, foram comparados dois classificadores distintos HDIV e HVAR. A Tabela 9

Tabela 9: Atributos referentes aos resultados de outras ferramentas

Atributo	Descrição	iFish	PolyPhen2
PolyPhen2_HDIV	Predição da ferramenta PolyPhen2 treinado com HDIV	✓	
PolyPhen2_HVAR	Predição da ferramenta PolyPhen2 treinado com HDIV	✓	
SIFT	Predição da ferramenta SIFT	✓	
MUTATION ASSESSOR	Predição da ferramenta Mutation Assessor	✓	

4.2.8 Dados de entrada

Em todos os testes feitos nesse trabalho, foram utilizadas sequências genômicas *GRCh37* e a versão do sequenciamento do genoma humano *hg19*. A entrada das ferramentas segue uma ordem padrão de entrada com algumas diferenças na formatação. A sequência básica de entrada é: **cromossomo, posição, base de referencia, base alternada**.

Um exemplo para essa entrada, é o da ferramenta *PolyPhen2*, onde o padrão para analisar-se uma mutação em determinado gene é: **cromossomo:posição referência/alternada**.

4.2.9 Dados de saída

As ferramentas produzem *scores* com base nos seus classificadores previamente treinados, para cada uma das entradas fornecidas. Por exemplo, executando a ferramenta *iFish*, obtêm-se uma probabilidade da SNP informada na entrada ser patogênica e uma classe qualificando este valor. Cada uma das ferramentas possui o seu próprio *score* e suas classes, que informam de diferentes maneiras se uma SNP é patogênica ou não.

5 METODOLOGIA

Nesta seção são detalhados os métodos utilizados para atingir os objetivos deste trabalho. Primeiramente, o contexto de toda a execução do trabalho é explicado, mostrando cada etapa do desenvolvimento.

Em seguida, essas etapas são divididas em subseções, detalhando ainda mais os processos referentes a cada uma das seções.

O processo inteiro baseia-se na metodologia empregada pela ferramenta *iFish* (Meng e Liping 2016), diferenciando em alguns pontos, os quais são descritos em detalhes nas seções seguintes.

5.1 Workflow

Para que fosse possível criar os modelos de predição de SNPs, foram necessárias uma série de etapas, desde a coleta dos dados em bancos públicos, ao processamento desses dados e treinamento dos modelos. A figura 8, demonstra uma visão geral da metodologia adotada.

Primeiramente, foi feita a busca em bancos de dados públicos de SNPs conhecidas, para que pudessem ser avaliadas pelos modelos. Foi escolhida a base de dados *clinVar*, onde a patogenicidade da mutação é conhecida, permitindo o uso dessa informação como atributo alvo no treinamento. Entre as instâncias coletadas no *clinVar*, foram consideradas duas possíveis classificações, SNPs patogênicas ou não patogênicas.

Filtros foram criados pré processar os dados coletados preparando entradas para o treinamento dos modelos. Para esse trabalho, o intuito é validar três filtros distintos: um gene, um painel de genes, ou para todos os genes disponíveis no banco de dados usado para coleta.

Após a coleta dos dados ser concluída, separa-se as entradas em dois subconjuntos, um para treinamento e outro para validação dos modelos obtidos. O conjunto de treinamento é submetido aos filtros.

A partir do subconjunto de treinamento, partiu-se para a utilização de algoritmos de seleção de atributos para customizar os modelos conforme necessário pois segundo

(Meng e Liping 2016), ao utilizar-se técnicas de seleção de atributos, é possível especializar o modelo para o subconjunto de dados específico. Após a seleção, é realizado o treinamento do modelo para predição das SNPs. O subconjunto de testes, é utilizado para testar o modelo final.

A figura 8 ilustra o processo em um workflow.

Todo o trabalho foi construído utilizando-se a linguagem *R*, pois possui uma série pacotes e ferramentas prontas que facilitam o desenvolvimento do projeto.

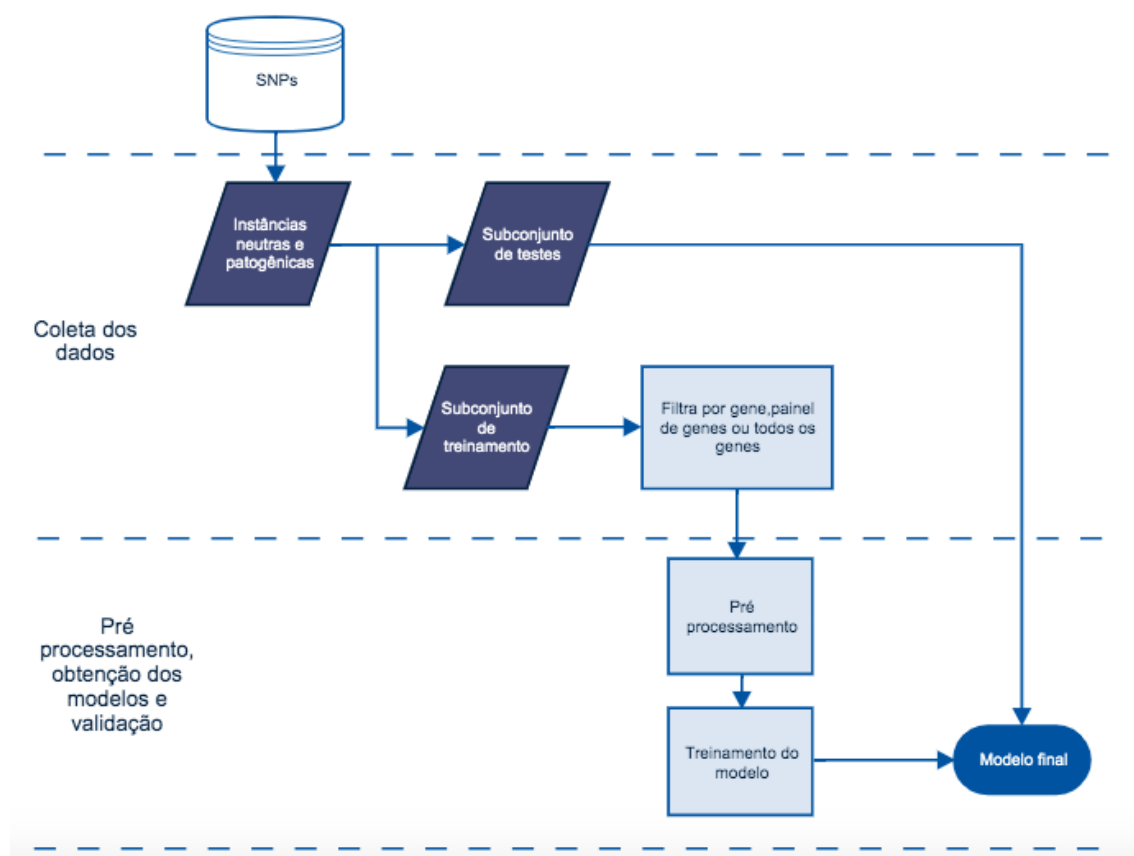


Figura 8: Workflow de execução de treinamento dos modelos

5.2 Coleta de dados

Esta seção detalha como foi feito o processo de coleta de informações para os modelos obtidos nesse trabalho.

5.2.1 Base de dados

Optou-se pelo uso da base de dados *clinVar* (MJ1 et al. 1999) pela sua facilidade de obtenção e pelo fato de possuir as informações de patogenicidade da variante. Como descrito anteriormente, a metodologia proposta por esse trabalho baseia-se na proposta pela ferramenta *iFish*, a qual utiliza outras fontes para a obtenção das informações. Na

etapa atual do projeto, não possuímos acesso aos dados do HGMD, impossibilitando a reprodução dos mesmos dados de treinamento utilizados pelo *iFish* (Meng e Liping 2016).

5.2.1.1 Coleta das informações

Para que fosse possível trabalhar com as informações fornecidas pelo *clinVar*, foram necessárias diversas etapas de processamento de informação até coletarmos todas as informações necessárias para os treinamento dos modelos.

No capítulo 4, foram descritos os atributos que são necessários para esse treinamento, incluindo também, quais ferramentas utilizam os atributos, no entanto, esses atributos não estão disponíveis no *clinVar*, tornando necessárias buscas em ferramentas externas para a coleta dessas informações. Por exemplo, o *polyphen2* (Adzhubei et al. 2010) e o *AnnoVar* (27) fornecem informações sobre a proteína a qual aquela variante é codificada. Essas ferramentas são utilizadas extensivamente pelo *iFish* para obtenção dos dados necessários para os seus cálculos.

Com o intuito de facilitar a coleta para este trabalho, utilizou-se o sistema de submissão do *iFish*, o qual possibilita solicitar as informações dos atributos para cada variante que se deseja realizar a predição da patogenicidade, integrando a partir de diferentes fontes os dados necessários para o treinamento dos modelos, com exceção da patogenicidade.

5.2.1.2 Processo de organização do banco de dados

Por ser um banco de dados amplamente utilizado no mercado e possuir uma versão gratuita, optou-se pela utilização do *MySQL*. Uma tabela foi criada para salvar os dados provenientes do *iFish*. Esta tabela, consiste em colunas representando todos os atributos retornados pela ferramenta, e está representada na figura 9, onde omitiu-se parte dos atributos presentes na tabela.



Figura 9: Diagrama entidade relacionamento do banco de dados

Em seguida, o arquivo VCF com todas as variantes presentes no *clinVar* foi transferido e carregado no R. O *Variant Call Format*(VCF), é um formato de arquivo de texto desenvolvido para o *1000 Genome Project*, utilizado para armazenar informações de polimorfismos. Esse formato, possui uma série de *headers* com diferentes informações sobre a variante. Neste trabalho, são utilizados os *headers* CHR, POS, ALT, REF e CLINSIG, que representam o cromossomo que ocorre a variante, a posição, a base de referência, a base alternada e a patogenicidade da variantes, respectivamente.

Nesse arquivo, foram aplicados uma série de processamentos para adaptar as suas variantes para o formato aceito pelo iFish *chr pos ref alt* ¹.

Após esse processamento, é gerado um arquivo de texto que é submetido no *iFish* para que a coleta dos atributos seja realizada.

Com os atributos devidamente preenchidos, é gerado um arquivo no formato TSV com todas as variantes e seus atributos. O formato TSV (*Tab separated values*) consiste em um arquivo de texto representando uma tabela, onde um *Tab*(Geralmente 4 espaços em branco), representa uma coluna, e toda nova linha no arquivo, representa uma nova linha na tabela. Esse arquivo é então importado para o banco de dados.

Por fim, é feito outro processamento no arquivo VCF original do *clinVar*, com o intuito de buscar a informação de patogenicidade de cada uma das variantes. Essa informação é atualizada na tabela do banco de dados, completando o processo de obtenção de

¹ A ferramenta aceita arquivos no formato VCF, porém no momento em que esse trabalho foi realizado, não estava funcionando corretamente, por isso optou-se pela criação da entrada manualmente

dados. Após esse procedimento, foram encontradas diversas entradas com informações inconsistentes em relação a patogenicidade da variante. Variantes de significado incerto, ou desconhecidos foram desconsideradas no treinamento, restando apenas àquelas que possuem uma significância. Também foram agrupadas as instâncias que possuíam significados com a nomenclatura de *provavelmente_patogênico* e *provavelmente_benigno*, em suas referidas classes patogênico e benigno, respectivamente, a fim de obtermos uma base de dados com um atributo alvo de apenas duas classes.

A figura 10, demonstra esse procedimento em um fluxograma. Esse fluxograma descreve todo o processo de obtenção de dados. Começando pela coleta dos SNVs da base de dados *clinVar*. Após esta etapa, aplica-se um filtro para a frequência populacional da variante, visto que, neste trabalho considera-se apenas SNPs. Após a aplicação do filtro, os dados resultantes são transformados para o formato aceito pela ferramenta *iFish*, para então serem submetidos e computados os atributos das variantes. O resultado do cálculo dos atributos é então importado no banco de dados local para a execução do processo de treinamento e validação dos modelos gerados.

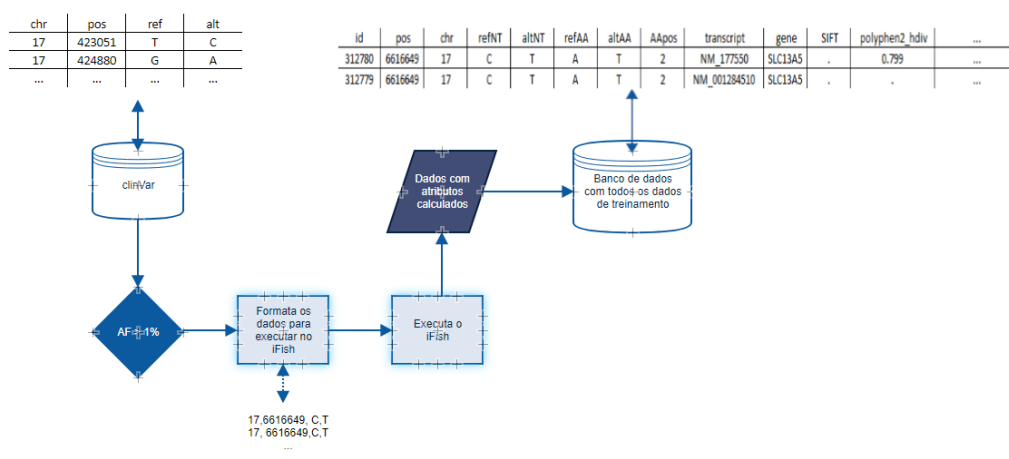


Figura 10: *Workflow* do processo de obtenção dos dados

5.3 Pré-processamento e obtenção dos modelos

Com as variantes coletadas e os atributos obtidos, partimos para a etapa de processar a informação e obter os modelos desejados.

5.3.1 Filtros

Foram criadas três opções de filtrações nos dados. A primeira, filtra somente por um gene específico, como por exemplo, o BRCA1. O segundo filtro, pode ser especificado um painel de genes desejado para o treinamento. Por último, pode ser treinado a base de dados completa.

Além de possuir estes filtros específicos, as consultas pelos dados no banco Mysql ainda desconsideram todas as instâncias que possuem classes diferentes de benigno e patogênico nos atributos alvo. Conforme, descrito na seção anterior, algumas variantes presentes no *clinVar* possuem significado incerto, não sendo utilizadas para o treinamento. Isso deve-se a natureza do treinamento proposto nesta metodologia, onde deseja-se obter apenas duas classes como resultado, patogênico ou benigno.

5.3.2 Dados faltantes

Durante o processo, foi detectada uma grande quantidade de dados faltantes para certos atributos da base de dados. Para tratar esses dados, optou-se por delimitar um limiar de número de instâncias mínimo que um atributo deve estar preenchido para que seja incluído no treinamento.

Quando um filtro é aplicado, calcula-se o percentual de instâncias em que é encontrado um valor para determinado atributo, caso o percentual de instâncias seja maior que 50% do total de entradas, o atributo passa a ser considerado como um possível candidato para o conjunto de treinamento.

Esse método é aplicado após a filtração devido à possibilidade de um atributo possuir uma diferente representatividade em certas filtrações, por exemplo, caso seja filtrado todas as instâncias para o gene BRCA1, é possível detectar que o atributo x possui um valor em 70% das instâncias, enquanto que, se aplicarmos o mesmo cálculo filtrando por todos os genes da base de dados, obtemos um valor de 50% de entradas. Esse fato, possibilita que o modelo final seja ainda mais específico para o grupo de genes filtrados.

Além de serem considerados apenas atributos com o percentual citado acima, também são consideradas as instâncias que tem um mínimo de atributos computados. Utilizou-se o limite inferior de 50% e essa eliminação de instâncias ocorre após a seleção das colunas mais relevantes.

5.3.3 Imputação de dados

Devido à limitação dos algoritmos utilizados no projeto, tornou-se necessária a utilização de técnicas de imputação de dados para o preenchimento dos dados faltantes das instâncias que restaram após a aplicação dos filtros. Inicialmente detecta-se quais colunas possuem um desvio padrão 0 para que, em seguida, os algoritmos do pacote *mice* possam ser executados, concluindo a imputação. A figura 11 demonstra o processo de remoção de colunas e entradas do banco e a aplicação da imputação de dados.

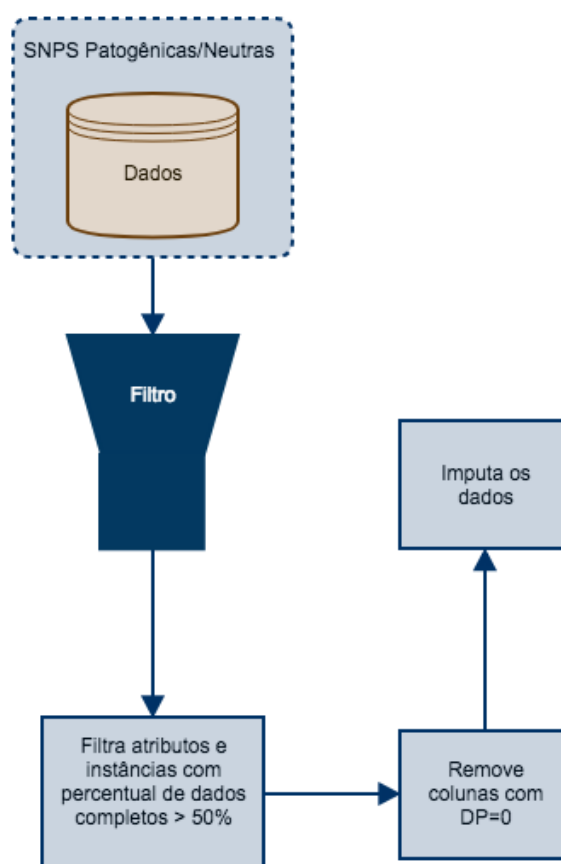


Figura 11: Pipeline de execução da imputação de dados

5.3.4 Seleção de atributos

Nesta etapa, os dados já estão pré processados para a execução dos algoritmos de seleção de atributos. Conforme descrito anteriormente, utiliza-se o algoritmo RFE disponibilizado pelo pacote *Caret* do R, customizado em duas formas distintas, utilizando algoritmos de seleção de modelo BIC e AIC para o ranqueamento de cada atributo e a técnica de classificação SVM para a avaliação de cada combinação de atributos testada pelo algoritmo. É feita uma validação cruzada de 10 *folds* repetida cinco vezes² para cada

²Sugestão da documentação do algoritmo

um dos modelos de ranqueamento.

Esse processo, resulta em 2 conjuntos de atributos selecionados, incluindo o número ideal de atributos para aquela seleção e o ranking de atributos resultante. O modelo resultante, é avaliado e é selecionado o que apresenta uma melhor performance em relação aos outros.

Os atributos do modelo resultante de melhor performance serão considerados para o treinamento do modelo final

5.3.5 Treinamento dos modelos

Com o conjunto de atributos selecionados pelo RFE, filtra-se a base de dados de treinamento com as colunas escolhidas pelo algoritmo.

É realizado um treinamento com o algoritmo de aprendizado supervisionado SVM e *randomForest* afim de comparar o resultado de ambos para escolher qual performa melhor no caso atual. Após ser escolhido o algoritmo que melhor performa, o modelo é submetido ao conjunto de testes separado anteriormente para validar a sua performance.

A figura 12, demonstra o processo de treinamento dos modelos, desde a etapa em que os dados são imputados, ao momento em que os algoritmos são executados.

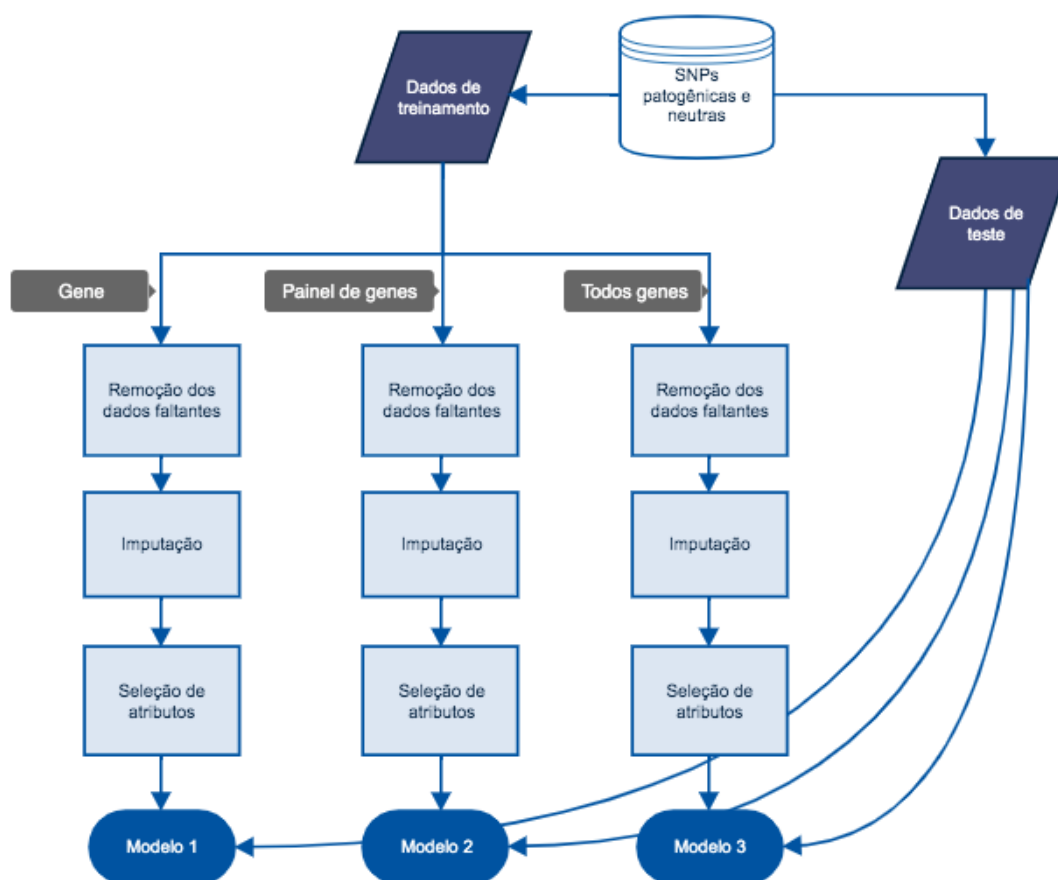


Figura 12: Pipeline de execução de treinamento dos modelos

5.3.6 Validação dos modelos

Para avaliar a qualidade dos modelos, separa-se um subconjunto de testes referentes a apenas um gene específico. O objetivo desta separação é para que seja possível comparar a performance do modelo específico com os mais genéricos, uma vez que, se forem utilizadas instâncias de teste que não estão relacionadas ao gene, seu modelo pode não performar como esperado. Por exemplo, no capítulo 6, utilizam-se dados de testes referentes ao gene BRCA1, uma vez que, são utilizados o filtro gene específico para o mesmo e também o filtro de painel de genes contém o BRCA1.

Para comparação entre os modelos gerados, é utilizada a acurácia de cada uma das etapas de validação. Primeiramente avalia-se qual método de seleção de atributos apresentou um melhor desempenho na acurácia e seleciona o modelo mais adequado. Em seguida, são comparadas as acurácias dos modelos gerados.

6 RESULTADOS

Esta seção, apresenta a aplicação da metodologia proposta nesse trabalho. Primeiramente, é discutido a presença de dados faltantes no *dataset* de treinamento e mostrado o seu resultado após a aplicação dos métodos citados no capítulo 5.

Em seguida, são comparados os resultados de cada um dos métodos de seleção de atributos para escolher qual possui uma melhor performance baseado na acurácia. Em seguida, o modelo é treinado com os atributos selecionados pelo *RFE* e também sem a seleção, para que seja possível comparar se há ganho de performance ao se aplicar a seleção.

Este processo é executado para os três filtros mencionados no capítulo 5.

O capítulo será dividido em três seções, sendo uma para cada um dos filtros.

6.1 Modelos treinados com todos os genes disponíveis

Esta seção apresenta os resultados obtidos ao selecionar-se todos os genes disponíveis na base de dados.

6.1.1 Pré-processamento

6.1.1.1 Dados faltantes

O processo de execução da aplicação é iniciado coletando os dados do banco com o filtro especificado. Neste caso, selecionam-se todos os genes disponíveis na base de dados. O resultado desta busca, trouxe um total de 361.843 instâncias, contendo 75 atributos. Destes 75 atributos, 64 são considerados como possíveis candidatos para o treinamento, visto que, os atributos restantes são referentes a identificação da SNP. Essa identificação inclui qual o cromossomo que ocorre a variante, as bases de referência e alternadas, informações dos alelos, qual o transcrito dessa variante e qual o gene da mesma. Também exclui-se da lista de atributos candidatos o atributo alvo *impact*.

É iniciado o processo de tratamento dos dados faltantes, começando pela identificação do percentual de instâncias e atributos incompletos. Conforme descrito anteriormente, são considerados todos os atributos que estejam presentes em uma frequência maior que 50%

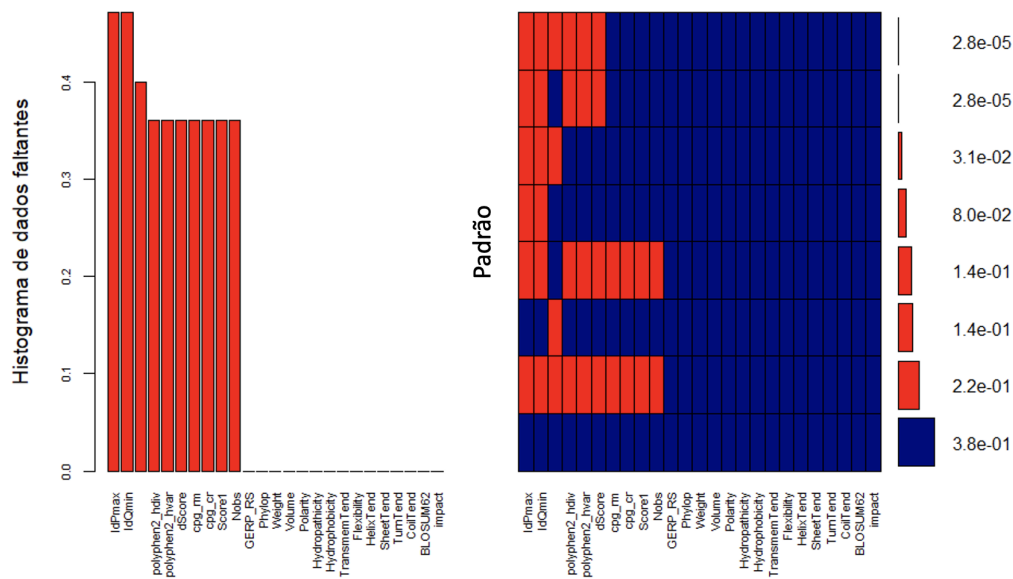


Figura 14: Histograma e distribuição dos dados faltantes para o filtro com todos os genes após a filtragem de variáveis e instâncias

Após a filtragem dos dados, analisa-se a distribuição das classes alvo nos dados restantes. Das 361.843 instâncias, 357.373 são da classe benigna, enquanto que as 3877 restantes, representam a classe patogênica. Essa distribuição representa um percentual de 98,92% das entradas na classe benigna e 1,08% na classe patogênica. Para que os algoritmos de classificação pudessem ser executados com menor viés no processo de treinamento, utilizou-se a amostragem aleatória para reduzir o número de entradas da classe benigna. A amostragem resultou em um total de 8456 instâncias, distribuídas em 50% para cada classe, considerando todas as instâncias patogênicas existentes.

Com as colunas filtradas e a amostragem realizada, o algoritmo de imputação é executado. O próximo passo, consiste em utilizar os dados resultantes desta imputação para a seleção de atributos.

6.1.1.2 Seleção de atributos com RFE

Neste experimento, foi executado o algoritmo RFE com validação cruzada de 10 *folds*. Os subconjuntos de atributos com as possíveis combinações foram baseadas nos dados de entrada, neste caso, o algoritmo testa combinações de 1 atributo à 24 atributos com dois métodos distintos de ranqueamento, utilizando as métricas de avaliação de modelo BIC e AIC.

Para que fosse possível utilizar essas métricas, em cada iteração da etapa de ranqueamento do algoritmo RFE, o score de cada um dos atributos selecionados naquela iteração é calculado e ordenado para que os melhores atributos fossem computados no final da

execução da rotina do RFE.

O resultado desse método, é demonstrado na tabela 10. O score AIC apresentou uma acurácia maior em relação ao BIC, selecionando 24 atributos contra os 20 selecionados pelo BIC. É importante observar que ambos os métodos de seleção classificaram os mesmos atributos como melhores. Como o AIC selecionou todos os atributos disponíveis, optou-se por testar este filtro com os atributos selecionados pelo BIC, afim de validar a eficácia do algoritmo de seleção de atributos.

Método	Acurácia	Número de atributos selecionados	Top 5 atributos
AIC	0.7937	24	MutationAssessor, Phylop, polyphen2_hvar, BLOSUM62, dScore
BIC	0.7927	20	MutationAssessor, Phylop, polyphen2_hvar, BLOSUM62, dScore

Tabela 10: Resumo do resultado da execução do RFE

Assim, é finalizado o processo de seleção de atributos. O próximo passo é executar o treinamento dos modelos e comparar o resultado do *dataset* filtrado com as colunas selecionadas pelo *BIC* com os dados completos sem a seleção de atributos.

6.1.2 Treinamento dos modelos

Foram utilizados dois algoritmos de classificação neste trabalho, o *SVM* e o *Random Forest*. Os dados são separados em dois grupos, um para testes e outro para treinamento. Os dados de teste são referentes ao gene *BRCA1*, para que seja possível a comparação dos resultados de cada um dos filtros com o mesmos dados.

O atributo alvo possui duas classes *Benign* e *Pathogenic* e sua distribuição nas 8456 instâncias de treinamento é mostrada na tabela 11.

	Frequência	Percentual
Benign	4158	50,00
Pathogenic	4158	50,00

Tabela 11: Distribuição das classes do atributo alvo nos dados de treinamento

	Frequência	Percentual
Benign	77	52,38095
Pathogenic	70	47,61905

Tabela 12: Distribuição das classes do atributo alvo nos dados de teste

6.1.2.1 Treinamento dos dados com atributos selecionados pelo RFE

Conforme descrito na seção 6.3.1.2, foram selecionados 20 atributos entre os 24 disponíveis para o treinamento.

Os atributos selecionados nessa etapa, são ilustrados na figura 15. Nela, é mostrado um gráfico da densidade de cada atributo nas classes existentes no alvo. Onde cada linha representa uma das duas classificações, a linha azul representa a classe benigna, enquanto que a linha magenta representa a classe patogênica. Neste contexto, o gráfico de densidade representa o intervalo de valores de um determinado atributo para a classe predita.

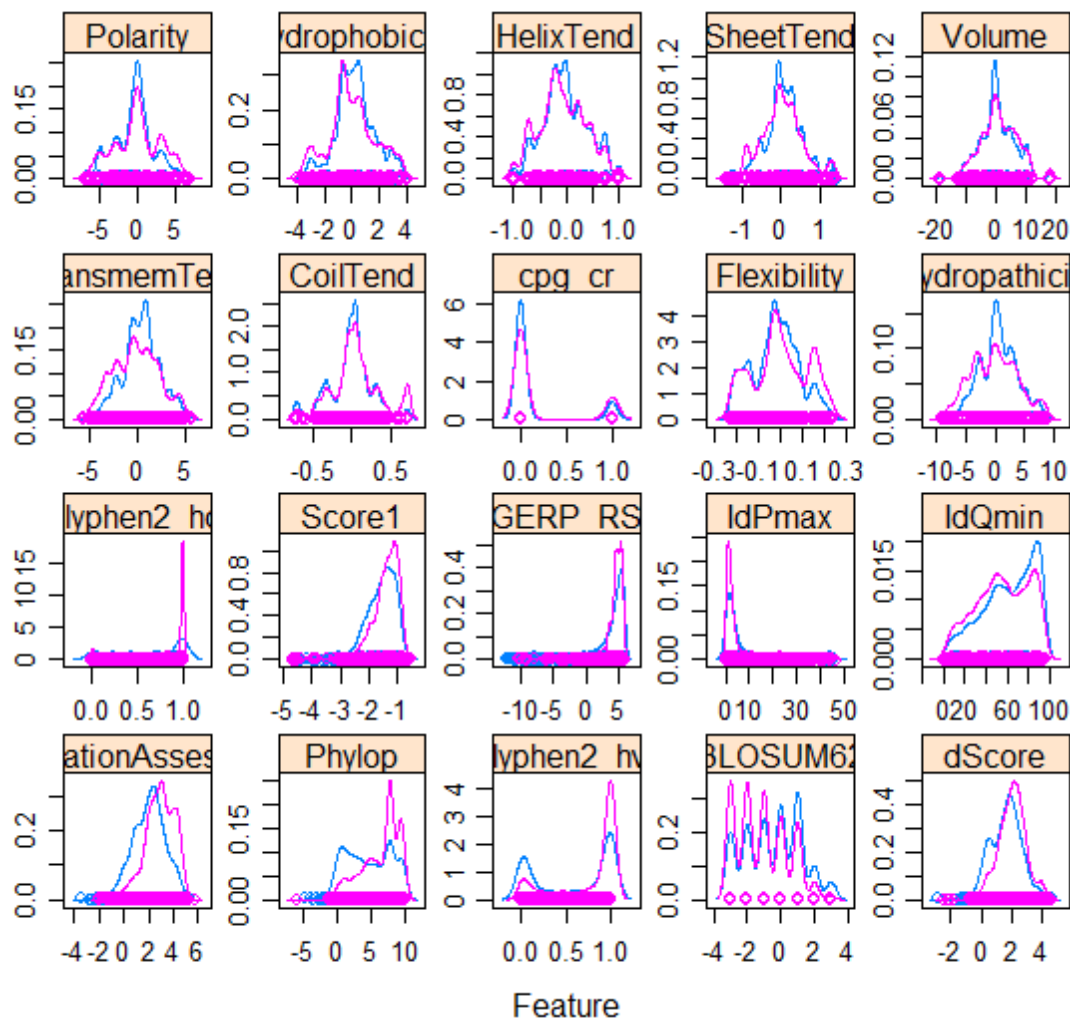


Figura 15: Densidade dos atributos selecionados pelo BIC

O resultado do treinamento, utilizando validação cruzada com 10 folds, é mostrado na figura 16. Nela é possível visualizar o intervalo de acurácia de cada um dos algoritmos, demonstrando o menor valor obtido, o valor médio e o valor máximo de acurácia. O resultado produzido pelo *Random Forest* apresenta uma acurácia média superior ao do SVM, 0.7955509 e 0.7019231 respectivamente. Como a acurácia vem sendo a medida utilizada como referência para as comparações entre modelos, é selecionado o resultado produzido pelo *Random Forest* como modelo para executar os dados de teste.

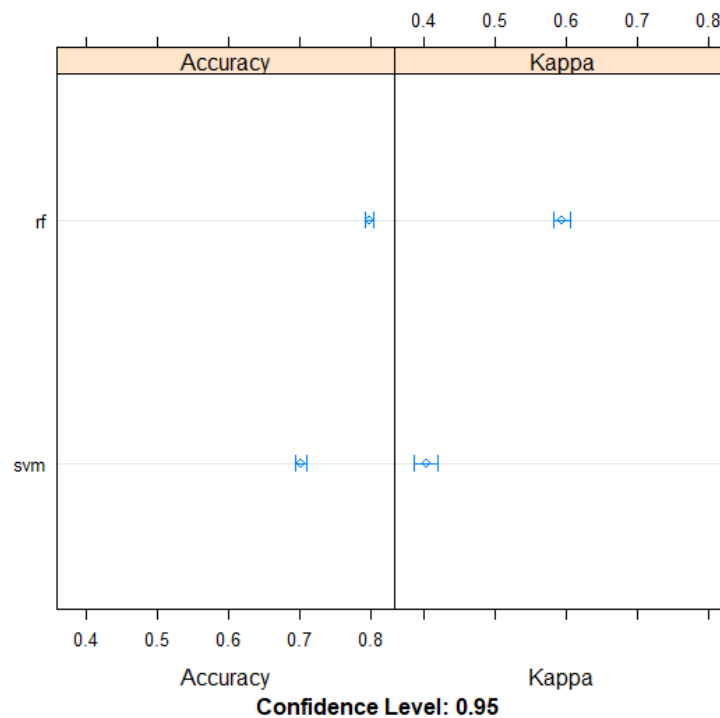


Figura 16: Acurácia dos modelos testados no dataset completo com seleção de atributos

O modelo resultante, faz a predição sobre as 147 instâncias de validação e apresenta a matriz de confusão representada pela tabela 13

Prediction	Benign	Pathogenic
Benign	38	5
Pathogenic	1	8

Tabela 13: Matriz de confusão resultante dos testes. As linhas representam os valores reais e as colunas os valores preditos

O resultado apresentou uma acurácia de 0.8846, uma sensibilidade de 0.6154 e uma especificidade de 0.9744, considerando como classe positiva o alvo *Pathogenic*.

6.1.2.2 Treinamento dos dados com todos os atributos candidatos

Seguindo os mesmos passos da seção anterior, são gerados os resultados para os 24 atributos selecionados inicialmente como possíveis candidatos. A figura 17 ilustra a densidade dos atributos selecionados.

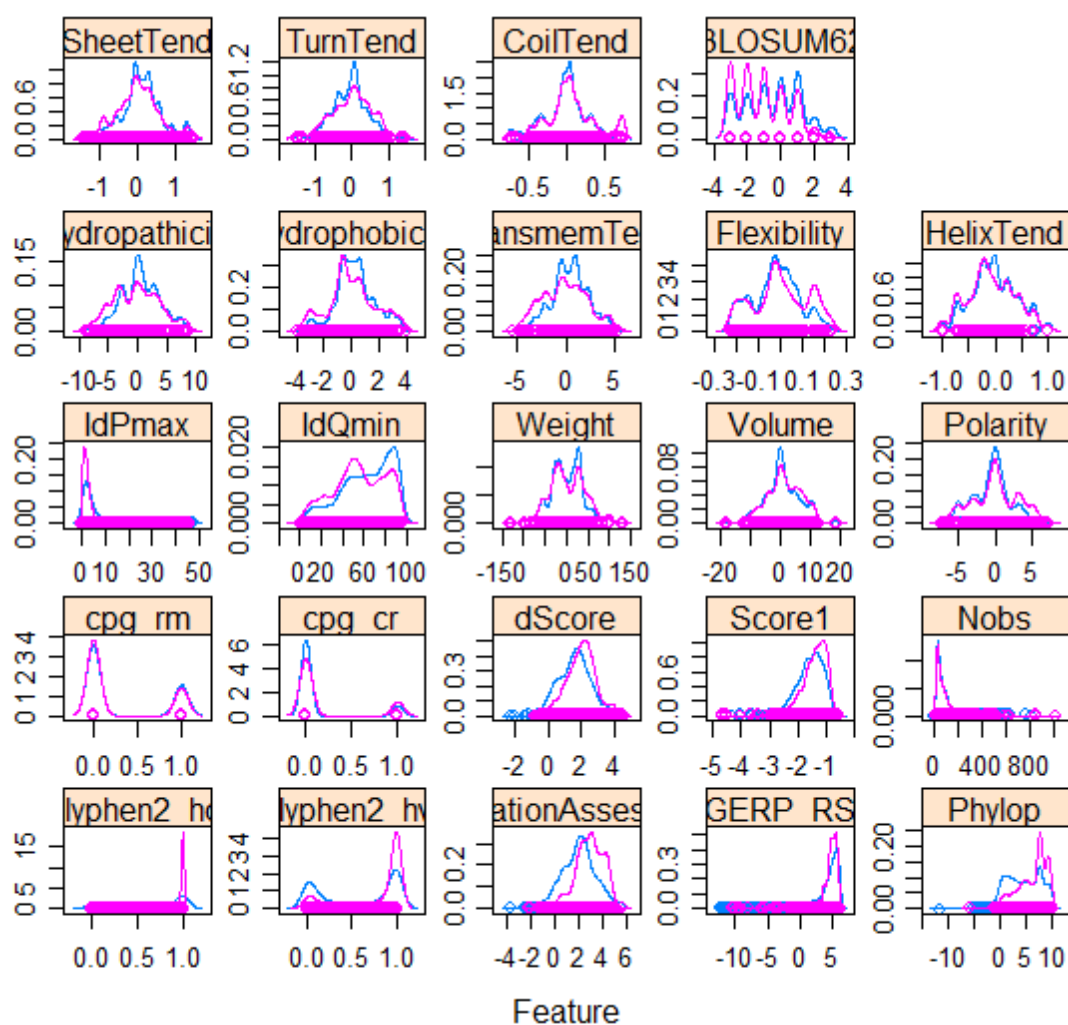


Figura 17: Densidade de todos os atributos candidatos

Após a execução do treinamento, foi obtido uma acurácia de 0.7979569 para o *Random Forest* e de 0.6981365 para o SVM. Novamente, o modelo escolhido foi o *Random Forest*. A figura 18 mostra a acurácia média de cada um dos modelos.

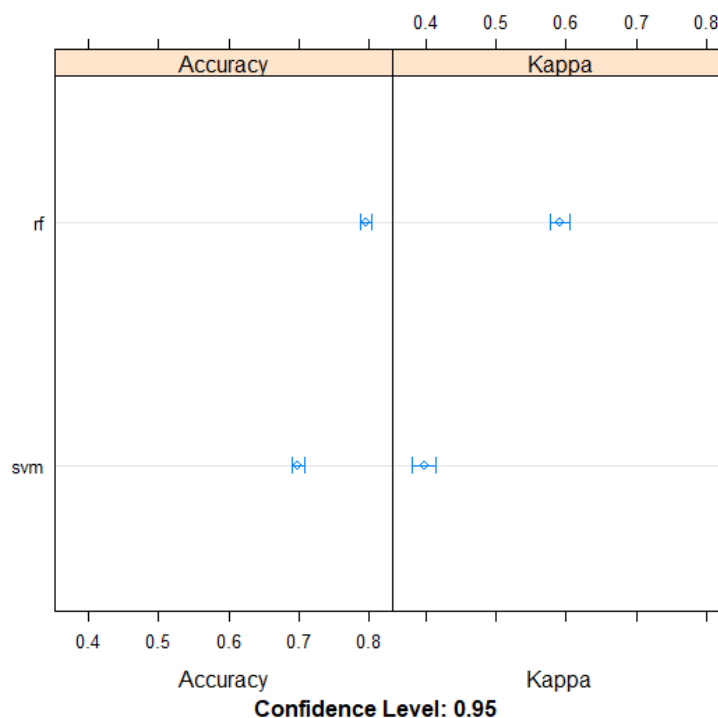


Figura 18: Acurácia dos modelos testados no dataset com todos os candidatos sem a seleção de atributos

Os dados de validação são executados no modelo produzindo a matriz de confusão representada na tabela 14.

Prediction	Benign	Pathogenic
Benign	39	3
Pathogenic	0	10

Tabela 14: Matriz de confusão resultante dos testes com todos os candidatos

O resultado produzido pelo pela execução do modelo nos dados de teste, apresentou uma acurácia de 0.9615, uma sensibilidade de 0.7692 e uma especificidade de 1, considerando como classe positiva o alvo *Pathogenic*.

6.1.2.3 Comparação dos resultados

Com os treinamentos executados, é possível visualizarmos uma diferença nas medidas de avaliação descritas na seção anterior. O modelo com todos os atributos candidatos, apresentou uma acurácia maior que a resultante do modelo com a seleção de atributos aplicada. A diferença no número de atributos entre os dois casos é pequena, apenas 4 atributos.

É importante ressaltar que durante a execução dos testes, nem todas as instâncias

são utilizadas na validação. Isto ocorre pois a biblioteca utilizada neste trabalho possui limitação para dados faltantes e, como durante a separação entre dados de teste e dados de treinamento, a imputação ainda não havia sido executada, existem dados faltantes nas 147 instâncias de teste. O modelo consegue prever somente instâncias que possuam as colunas selecionadas completas, neste teste com todos os genes, sobraram 52 instâncias das 147 para validação.

6.2 Modelos treinados com dados do painel de genes

Esta seção apresenta os resultados obtidos ao selecionar-se o painel de genes relacionados ao câncer de mama. Neste painel estão incluídos os genes BRCA1, BRCA2, PALB2, TP53, CDH1, STK11 e PTEN ¹.

6.2.1 Pré-processamento

6.2.1.1 *Dados faltantes*

A coleta de dados para este filtro retornou um total de 9361 instâncias. A figura 19, mostra a distribuição de cada um dos atributos baseado no total de instâncias. É possível notar que para o filtro escolhido neste experimento, existem atributos muito próximos a 100% das entradas sem nenhum valor. Como o agrupamento apresentado na seção anterior abrange todos os genes disponíveis, é esperado que a distribuição dos dados faltantes seja semelhante no agrupamento por painel de genes. No entanto, ainda existe a diferença na distribuição, pois cada agrupamento pode conter um número de instâncias que tenham um determinado atributo com mais valores. A figura 19, demonstra esse comportamento.

¹<https://www.genomika.com.br/exames/MAMATEST>

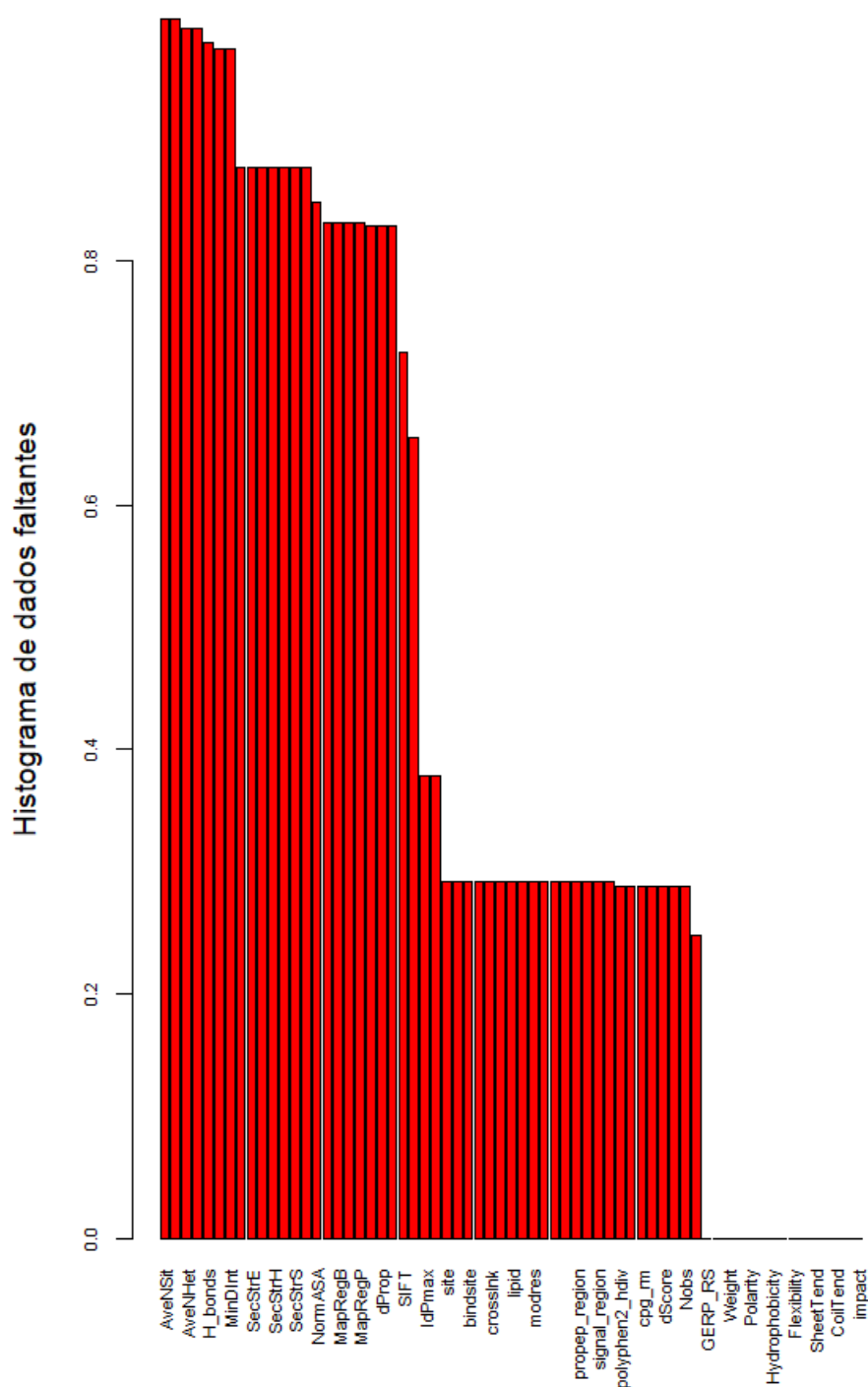


Figura 19: Histograma dos dados faltantes para o filtro com o painel de genes

Os dados são filtrados, resultando em 6660 instâncias com 40 variáveis. A figura 20, mostra novamente um histograma dos dados faltantes por atributo após a filtragem e também a distribuição nas instâncias desses atributos. É possível observar que 86% das instâncias estão completas e que existem 19 atributos com dados incompletos.

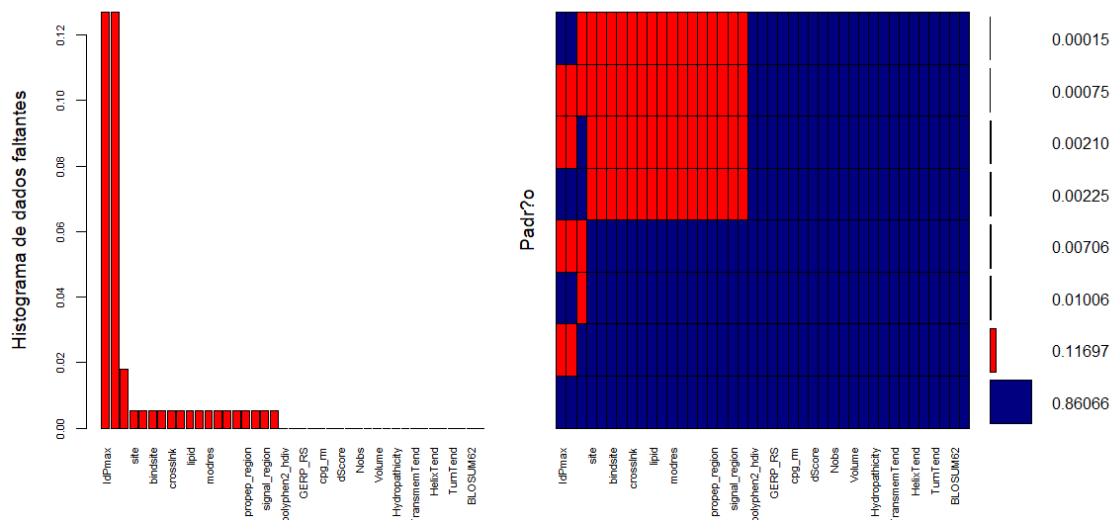


Figura 20: Histograma e distribuição dos dados faltantes para o filtro com o painel de genes após a filtragem de variáveis e instâncias

Após a filtragem dos dados, analisa-se a distribuição das classes alvo nos dados restantes. Das 6660 instâncias, 6422 são da classe benigna, enquanto que as 238 restantes, representam a classe patogênica. Essa distribuição representa um percentual de 96.42% das entradas na classe benigna e 3.57% na classe patogênica. Para que os algoritmos de classificação pudessem ser executados sem viés em relação à classe majoritária, utilizou-se a amostragem aleatória para reduzir o número de entradas da classe benigna. A amostragem resultou em um total de 476 instâncias, distribuídas em 50% para cada classe, considerando todas as instâncias patogênicas existentes.

A imputação inferiu dados de 8 atributos: MutationAssessor, site, actsite, modres, compbias_region, repeat_region, IdPmax e IdQmin. O restante dos atributos com dados faltantes, possuíam dados com desvio padrão 0, portanto foram removidos também. O resultado dessa operação, resultou em um dataset com 30 atributos.

O próximo passo, consiste em utilizar os dados resultantes desta imputação para a seleção de atributos.

6.2.1.2 Seleção de atributos com RFE

Neste experimento, foi executado o algoritmo RFE com validação cruzada de 10 *folds*. Os subconjuntos de atributos com as possíveis combinações foram baseadas nos dados de entrada, neste caso, o algoritmo testa combinações de 1 atributo à 30 atributos.

O resultado desse método, é demonstrado na tabela 10. O score BIC apresentou uma acurácia maior em relação ao AIC, com número de atributos distintos, 30 para o AIC e 29 para o BIC. Ambos os métodos de seleção classificaram os mesmos atributos como os melhores. Sendo assim, o BIC foi o score selecionado para a etapa de treinamento.

Método	Acurácia	Número de atributos selecionados	Top 5 atributos
AIC	0.8656	30	polyphen2_hvar, polyphen2_hdiv, IdPmax, MutationAssessor, dScore
BIC	0.8823	29	polyphen2_hvar, polyphen2_hdiv, IdPmax, MutationAssessor, dScore

Tabela 15: Resumo do resultado da execução do RFE para o painel de genes

O próximo passo é executar o treinamento dos modelos e comparar o resultado do *dataset* filtrado com as colunas selecionadas pelo *BIC* com os dados completos sem a seleção de atributos. Neste filtro, teremos duas bases de entrada, uma com os 29 atributos selecionados pelo *BIC* e a outra com os 30 atributos que restaram do pré-processamento inicial.

6.2.2 Treinamento dos modelos

Os dados de teste são referentes ao gene *BRCA1*, os mesmos utilizados para o teste do classificador com todos os genes.

O atributo alvo possui duas classes *Benign* e *Pathogenic* e sua distribuição das 476 instâncias de treinamento é mostrada na tabela 16. A distribuição dos dados de teste é demonstrada na tabela 17

	Frequência	Percentual
Benign	238	50,00
Pathogenic	238	50,00

Tabela 16: Distribuição das classes do atributo alvo nos dados de treinamento

	Frequência	Percentual
Benign	77	52,38095
Pathogenic	70	47,61905

Tabela 17: Distribuição das classes do atributo alvo nos dados de teste

6.2.2.1 Treinamento dos dados com atributos selecionados pelo RFE

Conforme descrito na seção 6.3.1.2, foram selecionados 29 atributos entre os 30 disponíveis para o treinamento.

Os atributos selecionados nessa etapa, são ilustrados na figura 21. Nela, é mostrado um gráfico da densidade de cada atributo nas classes existentes no alvo, onde cada linha representa uma das duas classificações, novamente, a linha azul representa a classe benigna e a linha magenta representa a classe patogênica. Neste contexto, o gráfico de densidade representa o intervalo de valores de um determinado atributo para a classe predita.

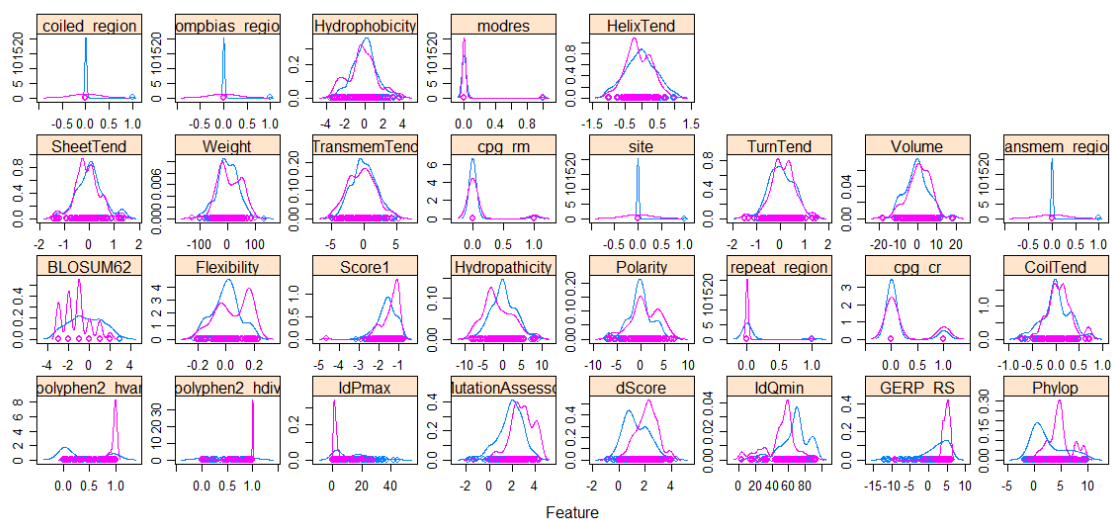


Figura 21: Densidade dos atributos selecionados pelo BIC

O resultado do treinamento, utilizando validação cruzada com 10 folds, é mostrado na figura 22. Nela é possível visualizar o intervalo de acurácia de cada um dos algoritmos, demonstrando o menor valor obtido, o valor médio e o valor máximo de acurácia. O resultado produzido pelo *Random Forest* apresenta uma acurácia média superior ao do SVM, 0.8843085 e 0.7998670 respectivamente. Como a acurácia vem sendo a medida utilizada como referência para as comparações entre modelos, é selecionado o resultado produzido pelo *Random Forest* como modelo para executar os dados de teste.

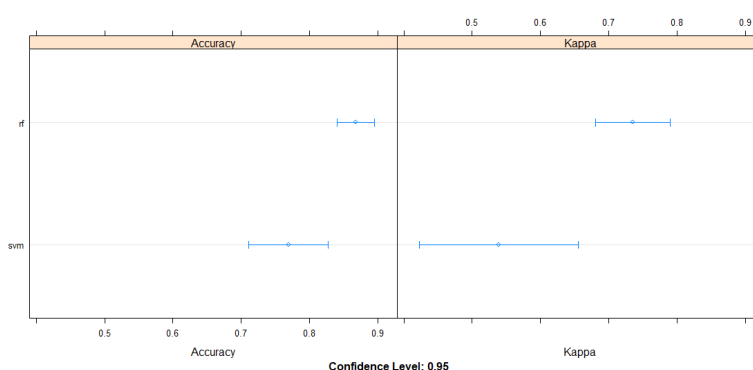


Figura 22: Acurácia dos modelos testados no dataset com a seleção de atributos no painel de genes

O modelo resultante, faz a predição sobre as 147 instâncias de validação e apresenta a matriz de confusão representada pela tabela 18

Prediction	Benign	Pathogenic
Benign	31	2
Pathogenic	7	8

Tabela 18: Matriz de confusão resultante dos testes. As linhas representam os valores reais e as colunas os valores preditos

O resultado apresentou uma acurácia de 0.8125, uma sensibilidade de 0.8 e uma especificidade de 0.8158, considerando como classe positiva o alvo *Pathogenic*.

6.2.2.2 Treinamento dos dados com todos os atributos candidatos

Seguindo os mesmos passos da seção anterior, são gerados os resultados para os 30 atributos selecionados inicialmente como possíveis candidatos. A figura 23 ilustra a densidade dos atributos selecionados.

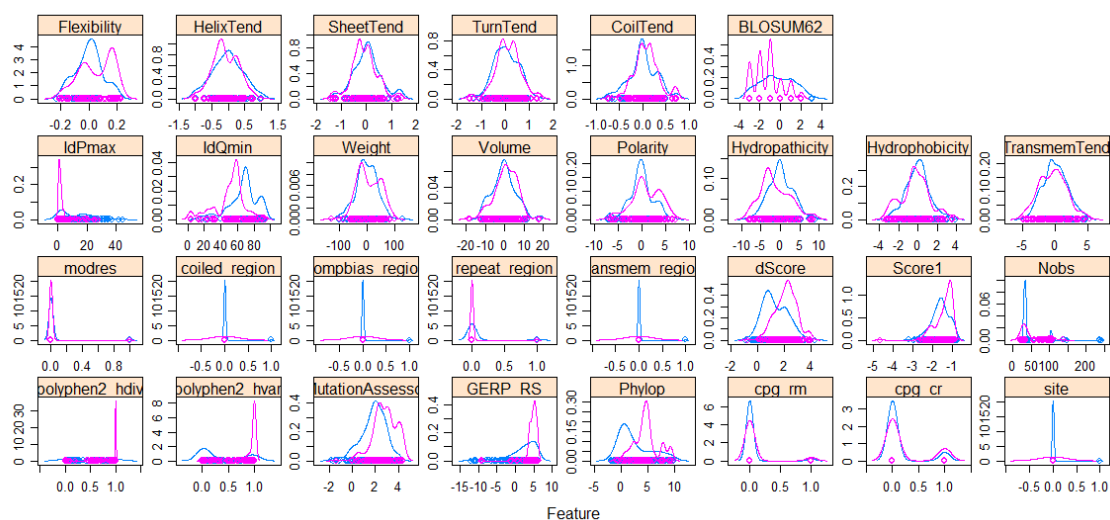


Figura 23: Densidade de todos os atributos candidatos

Após a execução do treinamento, foi obtido uma acurácia de 0.8958333 para o *Random Forest* e de 0.7894504 para o SVM. Novamente, o modelo escolhido foi o *Random Forest*. A figura 24 mostra a acurácia média de cada um dos modelos.

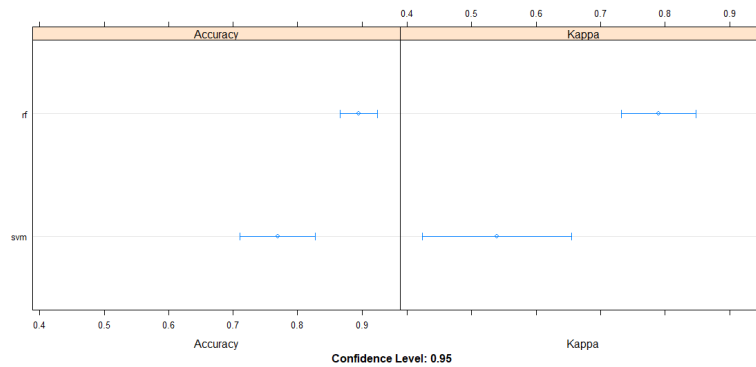


Figura 24: Acurácia dos modelos testados no dataset com todos os atributos candidatos para o o painel de genes

Os dados de validação são executados no modelo produzindo a matriz de confusão representada na tabela 19.

Prediction	Benign	Pathogenic
Benign	32	2
Pathogenic	6	8

Tabela 19: Matriz de confusão resultante dos testes com todos os candidatos

O resultado produzido pelo pela execução do modelo nos dados de teste, apresentou uma acurácia de 0.8333, uma sensibilidade de 0.8 e uma especificidade de 0.8421, considerando como classe positiva o alvo *Pathogenic*.

6.2.2.3 Comparação dos resultados

O modelo com todos os atributos candidatos, apresentou uma acurácia maior que a resultante do modelo com a seleção de atributos aplicada. A diferença no número de atributos entre ambos os modelos é de apenas 1 atributo.

6.3 Modelos treinados com dados de um gene específico

Esta seção apresenta os resultados obtidos ao selecionar-se o gene BRCA1, para o qual o coletou-se dados para treinar e testar os modelos.

6.3.1 Pré-processamento

6.3.1.1 Dados faltantes

A coleta de dados para este filtro retornou um total de 593 instâncias. A figura 25, mostra a distribuição de cada um dos atributos baseado no total de instâncias. Devido ao grande número de atributos faltantes, não é possível plotar todos os nomes de atributos

em um gráfico de barras, portanto, embora as figuras não demonstrem todos os nomes de atributos, eles estão representados pelas barras dentro do gráfico. É possível notar que para o filtro escolhido neste experimento, existem muitos atributos muito próximos a 100% das entradas sem nenhum valor. Grande parte destes atributos também estão presentes na distribuição das seções anteriores.

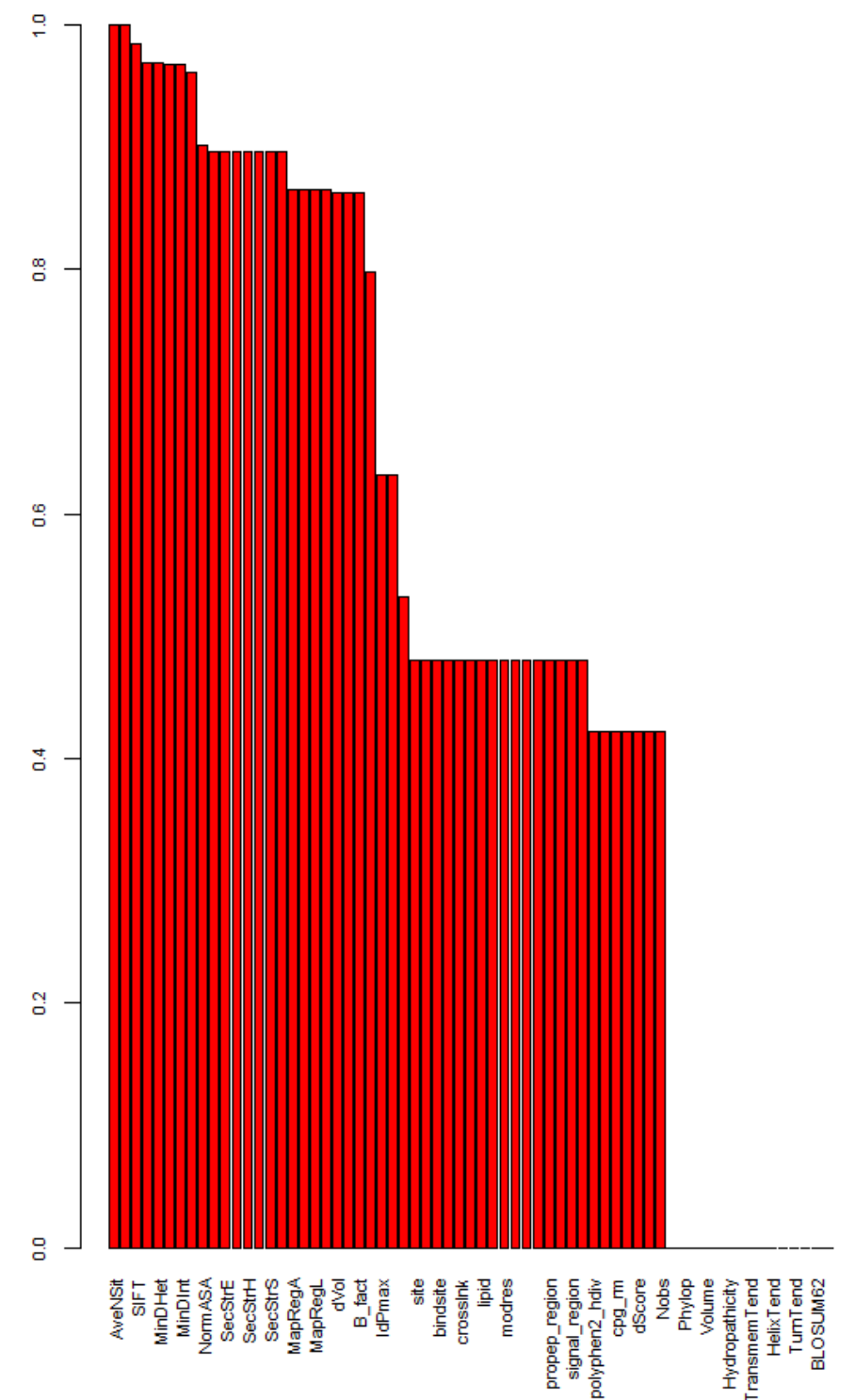


Figura 25: Histograma dos dados faltantes para o filtro pelo gene BRCA1

Os dados são filtrados, resultando em 343 instâncias com 37 variáveis. A figura 26, mostra novamente um histograma dos dados faltantes por atributo após a filtragem e também a distribuição nas instâncias desses atributos. É possível observar que 90% das instâncias estão completas e que existem 16 atributos com dados incompletos.

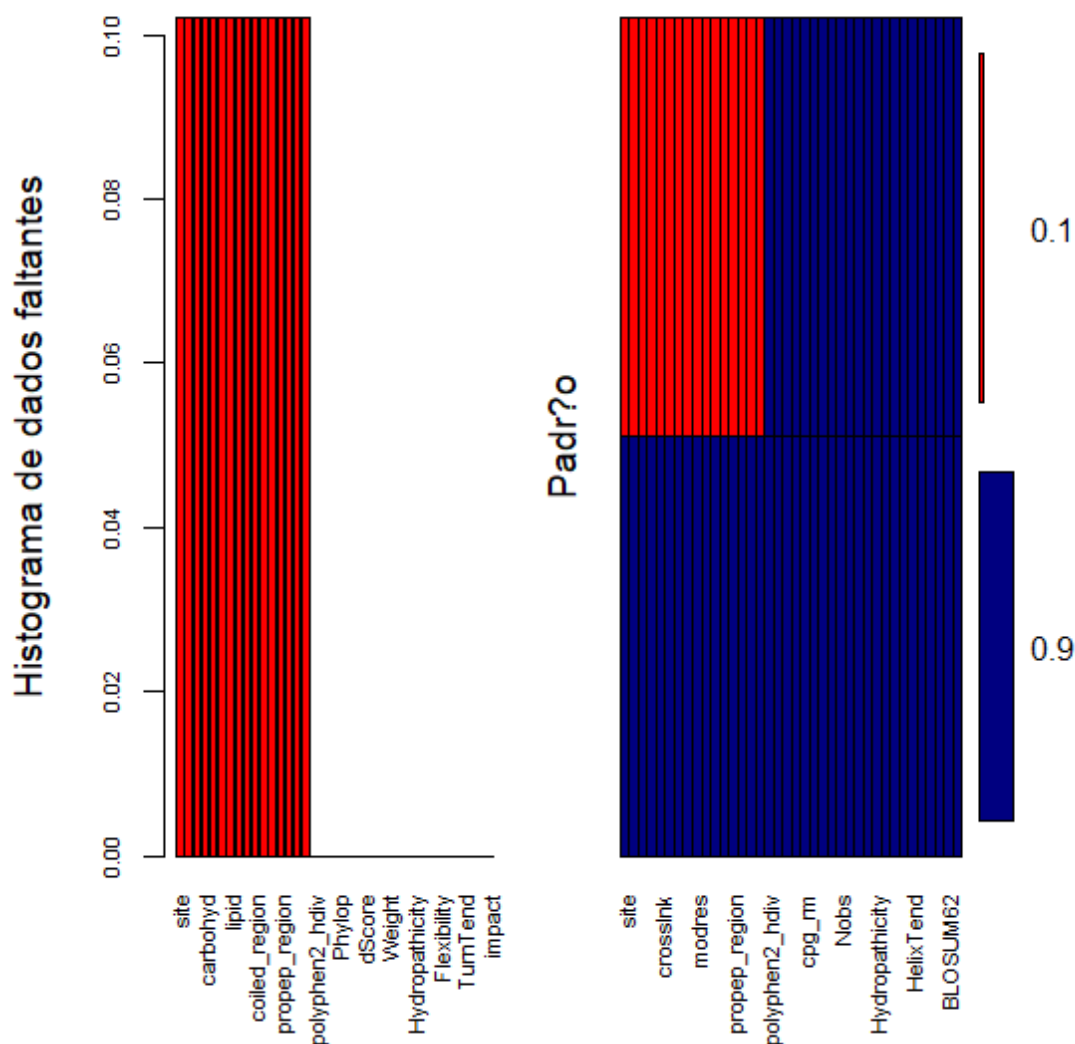


Figura 26: Histograma e distribuição dos dados faltantes para o filtro com o painel de genes após a filtragem de variáveis e instâncias

Das 343 instâncias, 182 são da classe benigna, enquanto que as 161 restantes, representam a classe patogênica. Essa distribuição representa um percentual de 53.06% das entradas na classe benigna e 46.93% na classe patogênica. A distribuição das classes estava balanceada, portanto, não foi necessária a utilização da amostragem.

A imputação inferiu dados de 1 atributo: modres. O restante dos atributos com dados faltantes, possuíam dados com desvio padrão 0, portanto foram removidos. O resultado dessa operação, resultou em um *dataset* com 22 atributos.

O próximo passo, consiste em utilizar os dados resultantes desta imputação para a

seleção de atributos.

6.3.1.2 Seleção de atributos com RFE

Neste experimento, foi executado o algoritmo RFE com validação cruzada de 10 *folds*. Os subconjuntos de atributos com as possíveis combinações foram baseadas nos dados de entrada, neste caso, o algoritmo testa combinações de 1 atributo à 22 atributos.

O resultado desse método, é demonstrado na tabela 20. O score AIC apresentou uma acurácia maior em relação ao BIC, com número de atributos distintos, 12 para o AIC e 9 para o BIC. Ambos os métodos de seleção classificaram os mesmos atributos como os melhores. Sendo assim, o AIC foi o score selecionado para a etapa de treinamento.

Método	Acurácia	Número de atributos selecionados	Top 5 atributos
AIC	0.9534	12	Phylop, polyphen2_hvar, GERP_RS, polyphen2_hdiv, dScore
BIC	0.9330	9	polyphen2_hvar, GERP_RS, polyphen2_hdiv, dScore

Tabela 20: Resumo do resultado da execução do RFE para o painel de genes

O próximo passo é executar o treinamento dos modelos e comparar o resultado do *dataset* filtrado com as colunas selecionadas pelo AIC com os dados completos sem a seleção de atributos. Neste filtro, teremos duas bases de entrada, uma com os 12 atributos selecionados pelo AIC e a outra com os 22 atributos que restaram do pré-processamento inicial.

6.3.2 Treinamento dos modelos

Os dados de teste são referentes ao gene BRCA1, os mesmos utilizados para o teste do classificador com todos os genes e o do painel de genes.

O atributo alvo possui duas classes *Benign* e *Pathogenic* e sua distribuição das 343 instâncias de treinamento é mostrada na tabela 21. A distribuição dos dados de teste é demonstrada na tabela 22

	Frequência	Percentual
Benign	182	53.06
Pathogenic	161	46.93

Tabela 21: Distribuição das classes do atributo alvo nos dados de treinamento

	Frequência	Percentual
Benign	77	52,38095
Pathogenic	70	47,61905

Tabela 22: Distribuição das classes do atributo alvo nos dados de teste

6.3.2.1 Treinamento dos dados com atributos selecionados pelo RFE

Foram selecionados 12 atributos entre os 22 disponíveis para o treinamento.

Os atributos selecionados nessa etapa, são ilustrados na figura 27. Nela, é mostrado um gráfico da densidade de cada atributo nas classes existentes no alvo. Onde cada linha representa uma das duas classificações. Neste contexto, o gráfico de densidade representa o intervalo de valores de um determinado atributo para a classe predita.

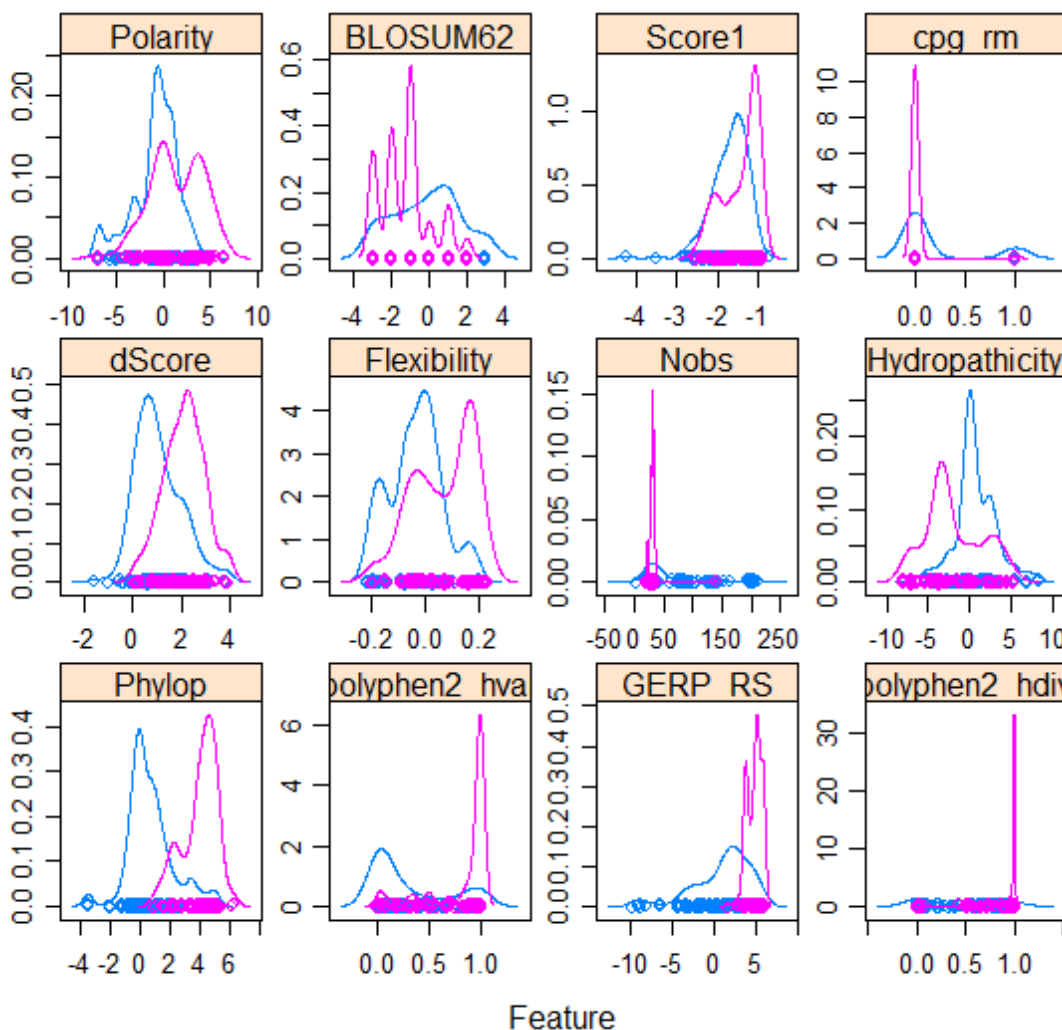


Figura 27: Densidade dos atributos selecionados pelo AIC

O resultado do treinamento, utilizando validação cruzada com 10 folds, é mostrado na figura 28. Nela é possível visualizar o intervalo de acurácia de cada um dos algoritmos, demonstrando o menor valor obtido, o valor médio e o valor máximo de acurácia. O resultado produzido pelo *Random Forest* apresenta uma acurácia média superior ao do SVM, 0.9558824 e 0.9411765 respectivamente. Como a acurácia vem sendo a medida utilizada como referência para as comparações entre modelos, é selecionado o resultado produzido pelo *Random Forest* como modelo para executar os dados de teste.

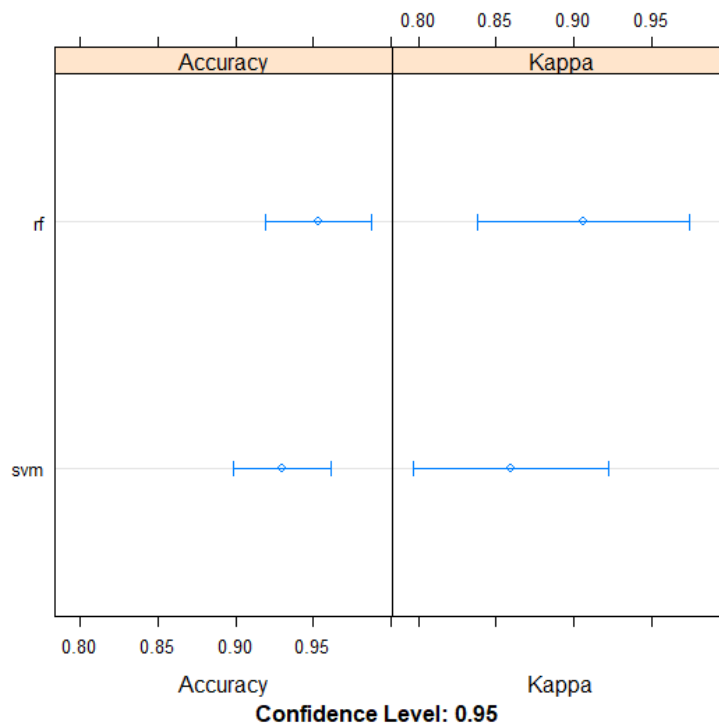


Figura 28: Acurácia dos modelos testados no dataset com a seleção de atributos e o filtro para um gene

O modelo resultante, faz a predição sobre as 147 instâncias de validação e apresenta a matriz de confusão representada pela tabela 23

Prediction	Benign	Pathogenic
Benign	48	0
Pathogenic	2	39

Tabela 23: Matriz de confusão resultante dos testes. As linhas representam os valores reais e as colunas os valores preditos

O resultado apresentou uma acurácia de 0.9775, uma sensibilidade de 1 e uma especificidade de 0.96, considerando como classe positiva o alvo *Pathogenic*.

6.3.2.2 Treinamento dos dados com todos os atributos candidatos

Seguindo os mesmos passos da seção anterior, são gerados os resultados para os 30 atributos selecionados inicialmente como possíveis candidatos. A figura 29 ilustra a densidade dos atributos selecionados.

Após a execução do treinamento, foi obtido uma acurácia de 0.9705882 para o *Random Forest* e de 0.9411765 para o SVM. Novamente, o modelo escolhido foi o *Random Forest*. A figura 30 mostra a acurácia média de cada um dos modelos.

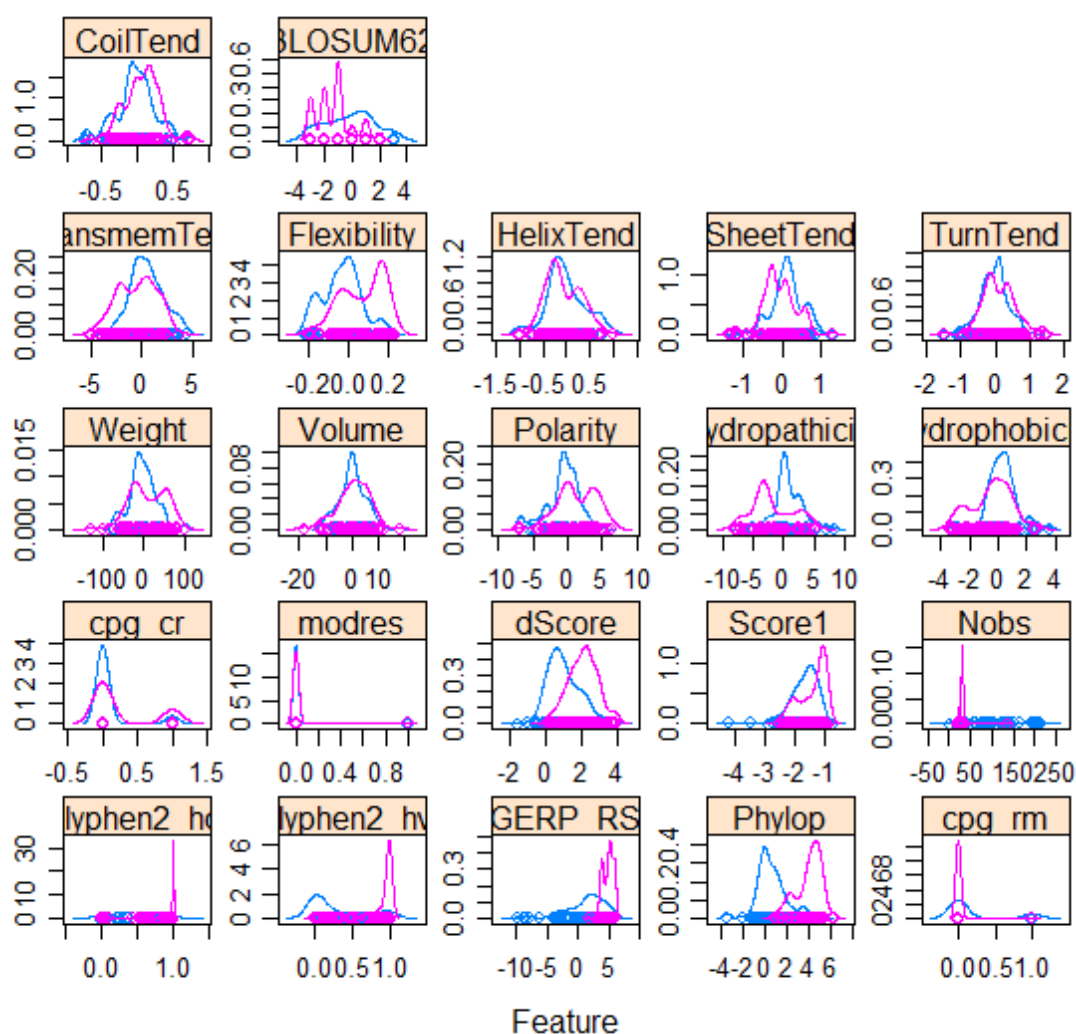


Figura 29: Densidade de todos os atributos candidatos

Os dados de validação são executados no modelo produzindo a matriz de confusão representada na tabela 24.

Prediction	Benign	Pathogenic
Benign	48	0
Pathogenic	0	33

Tabela 24: Matriz de confusão resultante dos testes com todos os candidatos

O resultado produzido pelo pela execução do modelo nos dados de teste, apresentou uma acurácia de 1, uma sensibilidade de 1 e uma especificidade de 1, considerando como classe positiva o alvo *Pathogenic*.

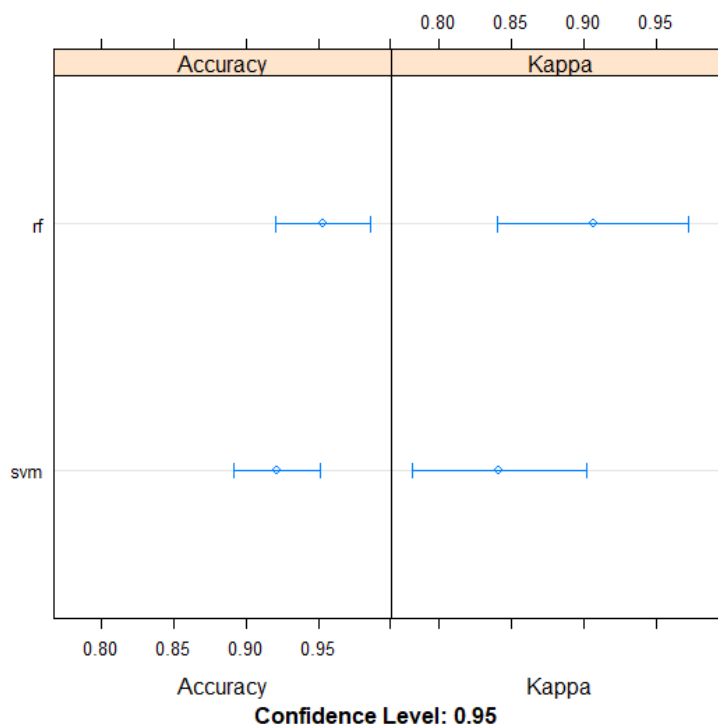


Figura 30: Acurácia dos modelos testados no dataset com todos os atributos candidatos com o filtro de um gene

6.3.2.3 Comparação dos resultados

O modelo com todos os atributos candidatos, apresentou uma acurácia maior que a resultante do modelo com a seleção de atributos aplicada, onde acertou todas a predição de todas as instâncias. Neste caso pode estar ocorrendo um *overfitting* no modelo, visto que ele está especializado somente para esse conjunto de dados. No capítulo seguinte, será feita a discussão sobre os resultados apresentados até o momento e quais são os possíveis trabalhos futuros para o aprimoramento da metodologia proposta neste trabalho.

6.4 Treinamento do genes restantes no painel

Esta seção apresenta os resultados do treinamento realizado no restante dos genes disponíveis no painel de genes escolhido, o BRCA2, PALB2, TP53, CDH1, STK11 e PTEN.

Para comparar o resultado obtido pelo modelo gene específico, implementou-se a mesma metodologia para todos os genes disponíveis no painel. Os genes BRCA2, PALB2, CDH1, STK11 e PTEN, não apresentaram quantidade dados suficientes para que a análise pudesse ser feita. Este fato ocorre no momento em que o pré processamento é aplicado, resultando em nenhum dado disponível para testar o método. Ao validar os dados no banco, é possível identificar um grande número de dados faltante para os mesmos,

refletindo no resultado.

O gene restante TP53, possui dados suficientes para realização dos testes. A distribuição de instâncias de teste está representada pela tabela 25. Nota-se que os dados do atributo alvo estão desbalanceados, se comparados às outras instâncias de teste.

	Frequência	Percentual
Benign	131	15.26807
Pathogenic	727	84.73193

Tabela 25: Distribuição das classes do atributo alvo nos dados de teste do gene TP53

Novamente, aplicou-se a seleção de atributos, onde todos os atributos disponíveis foram selecionados. Em seguida, foram executados os treinamentos. O resultado é demonstrado na figura 31. Novamente o algoritmo random forest performou melhor, apresentando uma acurácia de 1.

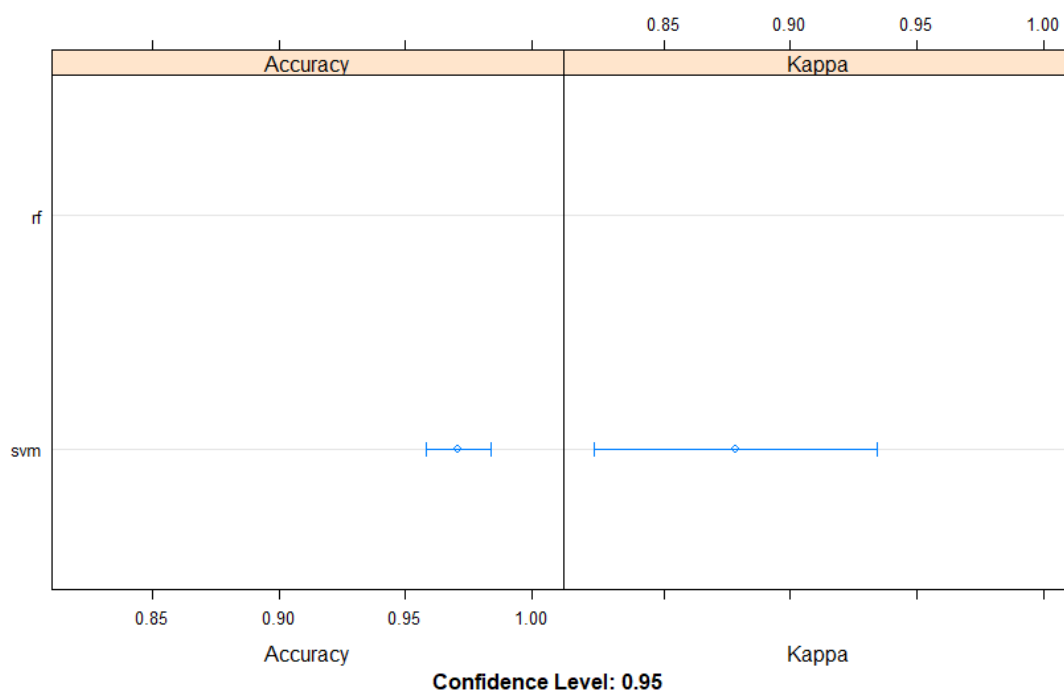


Figura 31: Acurácia dos modelos testados no dataset com todos os atributos candidatos com o filtro do gene TP53

Este resultado leva à interpretação de que quanto mais especializamos o modelo, menor impacto a seleção de atributos causa e maior é o *overfitting*. A capítulo 7 apresentará os resultados de forma comparativa e irá discutir o significado dos mesmos.

7 DISCUSSÃO

Nesta seção, são discutidos os resultados obtidos pela metodologia proposta neste trabalho.

As figuras 32 e 33 demonstram dois gráficos comparativos. Nelas, são apresentadas as acurácias de treino e de teste para cada um dos filtros utilizados na metodologia. A figura 32, apresenta os resultados obtidos quando a seleção de atributos foi realizada, enquanto que, a figura 33, mostra os resultados quando não se executou a seleção de atributos.

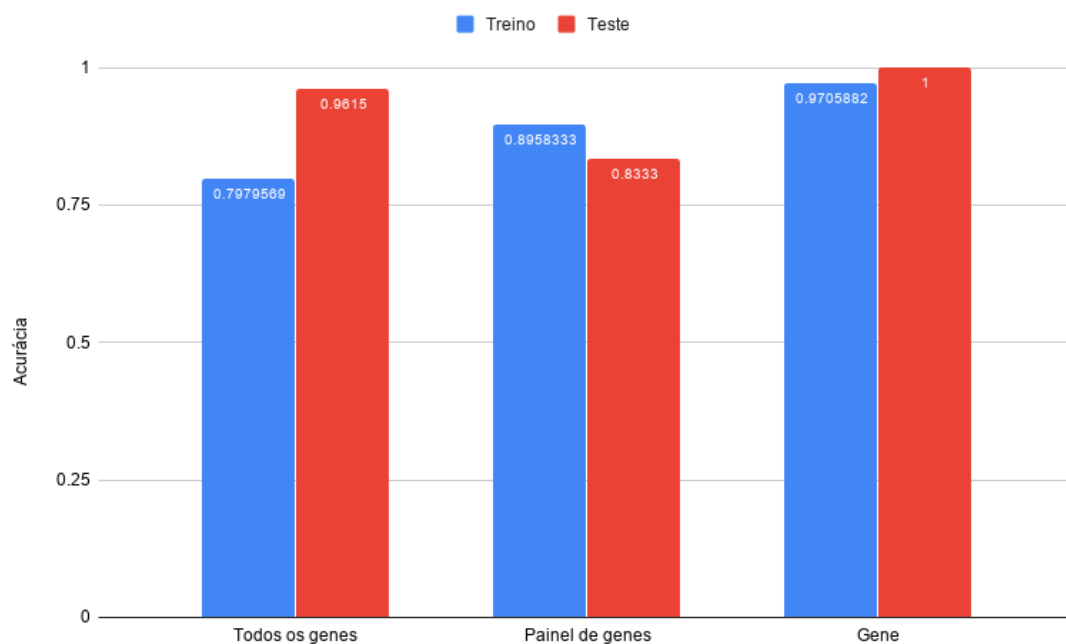


Figura 32: Gráfico de comparação dos resultados com a seleção de atributos

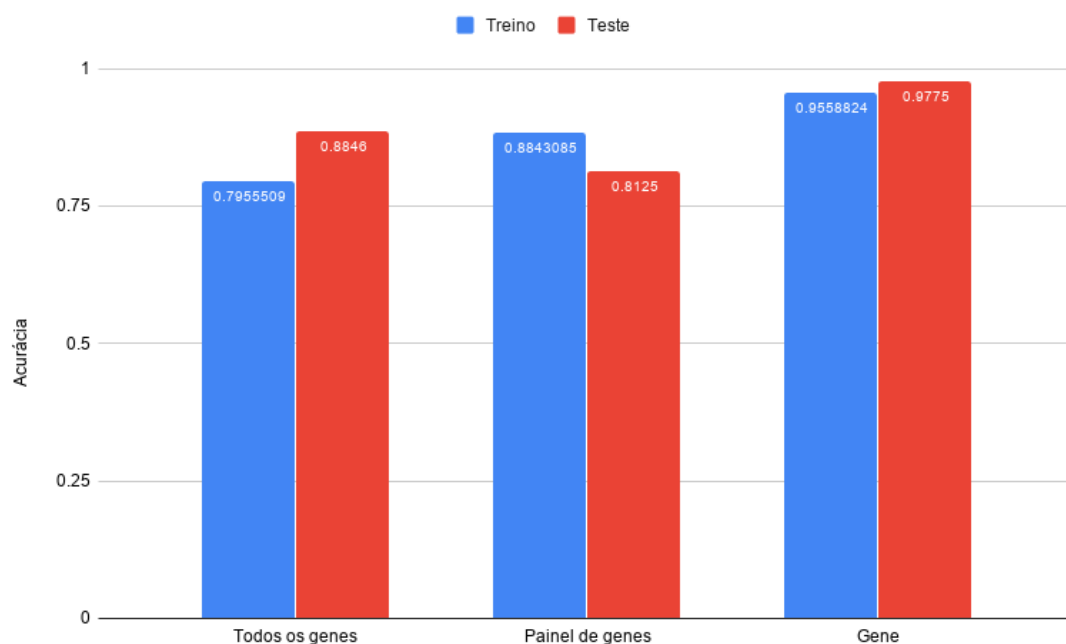


Figura 33: Gráfico de comparação dos resultados sem a seleção de atributos

Em ambos os casos, é possível visualizar que há um crescimento na acurácia dos testes conforme o modelo é treinado em bases especializadas. Por exemplo, na figura 32, conforme os filtros foram sendo aplicados, o modelo gerado teve um aumento do valor de sua acurácia.

No entanto, ao aplicar os dados de testes, nota-se que quando os dados são filtrados pelo painel de genes, o modelo não segue o mesmo padrão de crescimento. Diversos fatores podem estar gerando essa divergência, o principal deles está relacionado aos dados que são utilizados como teste. Devido às limitações das bibliotecas utilizadas durante o desenvolvimento deste trabalho, não é possível realizar previsões com dados faltantes na base de testes. Isto significa que, quando é feita a validação do modelo gerado pela metodologia no caso de teste, somente as instâncias de teste que possuem todas as colunas selecionadas pelo modelo completas são consideradas o que, consequentemente, acaba gerando diferentes grupos de testes. Para demonstrar este caso, utilizando como exemplo a execução da metodologia para todos os genes e para o painel de genes, obteve-se as seguintes matrizes de confusão 26 e 27 ao utilizar-se os casos de teste. A primeira matriz possui um total de 39 instâncias de testes enquanto que a segunda matriz apresenta 38 casos. Além de existir a diferença no número total de instâncias, há também a diferença nas entradas que estão presentes nestes resultados, onde na tabela 26, 38 instâncias são da classe benigna contra apenas 1 da classe patogênica. Já na tabela 27, existem 31 instâncias benignas e 7 instâncias patogênicas. Para tentar evitar que houvesse esta diferença entre os conjuntos de teste para cada um dos filtros, foi executado o pré-processamento utilizado na metodologia para que os dados de teste estivessem padronizados em todos os

filtros. No entanto, cada um dos filtros apresenta diferentes combinações de atributos e para que o modelo possa validar um conjunto de testes, o mesmo deve obrigatoriamente possuir todos os atributos aos quais foi submetido durante o seu treinamento. Quando aplicado o pré-processamento, não foi possível utilizar o teste para todos os modelos gerados, pois o mesmo não incluía todas as colunas selecionadas pelos filtros. Com isso, faz-se necessária a utilização de outras bases para validar os modelos gerados pela metodologia, o que será discutido na seção de trabalhos futuros.

Prediction	Benign	Pathogenic
Benign	38	5
Pathogenic	1	8

Tabela 26: Matriz de confusão resultante dos testes para o filtro de todos os genes

Prediction	Benign	Pathogenic
Benign	31	2
Pathogenic	7	8

Tabela 27: Matriz de confusão resultante dos testes para o filtro do painel de genes

Apesar de possuir a divergência nos conjuntos de testes, é possível notar que o padrão de crescimento da acurácia dos modelos repete-se quando há ou não a seleção de atributos. Os gráficos 32 e 33 indicam também que a aplicação da seleção de atributos na base de dados do *clinVar*, não surtiu um efeito positivo em relação a acurácia. Tal fato, indica que quanto mais atributos selecionamos, melhor é a qualidade do modelo gerado. Porém, é importante ressaltar que a especialização dos modelos, ou seja, o treinamento de um modelo com os filtros, apresentou melhor desempenho no treinamento dos modelos, embora o filtro pelo painel de genes não tenha apresentado um bom desempenho na base de testes.

Outro fato importante é a escolha dos melhores atributos por cada um dos modelos de seleção. No capítulo anterior foram demonstrados quais foram os 5 melhores atributos escolhidos por cada modelo de seleção. Entre eles, os únicos que apareceram nos três filtros foram polyphen2_hvar e dScore. Isso demonstra a diferença na execução de cada seleção de atributos para os diferentes filtros.

Por fim, temos os testes realizados em outros genes disponíveis no painel. Dentre a seleção disponível, apenas foi possível executar em um deles, o TP53, devido à falta de dados suficientes para o treinamento dos outros genes disponíveis.

Este fato, reforça o argumento de que os dados não são completos o suficiente para compararmos os resultados e faz-se necessária a obtenção de uma base com mais infor-

mações disponíveis. No entanto, o resultado desta execução, seguiu o mesmo padrão apresentado pelo modelo do gene BRCA1, onde a acurácia aumenta conforme especializamos o modelo.

8 CONCLUSÃO

A metodologia desenvolvida neste trabalho é capaz de criar modelos especializados para painéis de genes e genes específicos. Conforme comprovado em (Meng e Liping 2016), os modelos específicos apresentam um desempenho melhor em relação aos modelos genéricos. Neste trabalho, é possível validar esta hipótese para os treinamentos demonstrados nos capítulos anteriores, no entanto, o comportamento apresentado no caso de testes validado durante os experimentos não demonstrou o mesmo comportamento para o painel de genes, somente para os filtros de genes específicos.

Para que se possa ter uma visão mais ampla de como essa metodologia pode ser aplicada em outros casos, é necessária a validação da mesma com outros dados de treinamento e testes. A base de dados utilizada durante o desenvolvimento deste trabalho, apresentou muitas informações faltantes e dados desbalanceados, o que pode afetar o resultado que deseja-se obter com esta metodologia.

A principal contribuição deste trabalho está na metodologia em si, a qual foi desenvolvida com intuito de validar os agrupamentos por painel de genes e como eles performam em relação aos outros tipos de filtros que já são implementados pelo trabalho de referência.

No presente momento, esta metodologia permite o treinamento com diferentes bases de dados para a sua execução, o que difere das ferramentas atuais, principalmente o iFish. Embora ainda seja necessário a continuidade deste trabalho, é possível utilizá-lo para testes com qualquer fonte de dados.

8.1 Trabalhos Futuros

Como principal trabalho para a continuação desta metodologia, deve-se obter novas fontes de dados para validar o processo como um todo. A intenção é obter informações mais balanceadas e com menos dados faltantes, a fim de possibilitar um treinamento com um maior número de atributos, o que também pode influenciar na hipótese de que a seleção de atributos melhora o resultado do treinamento.

Além de novas fontes de dados, também é necessário a utilização de diferentes painéis

de genes, a fim de validar se também ocorre o overfitting.

Há também a possibilidade de alteração dos algoritmos utilizados para cada uma das etapas da metodologia.

REFERÊNCIAS

- [2017]Uniprot: the universal protein knowledgebase. *Nucleic Acids Research*, v. 45, n. D1, p. D158–D169, 2017. Disponível em: <<http://dx.doi.org/10.1093/nar/gkw1099>>.
- [2]ADZHUBEI, I.; JORDAN, D. M.; SUNYAEV, S. R. Predicting functional effect of human missense mutations using polyphen-2. *Curr Protoc Hum Genet*, v. 7, Jan 2013.
- [Adzhubei et al. 2010]ADZHUBEI, I. A. et al. A method and server for predicting damaging missense mutations. *Nat. Methods*, v. 7, n. 4, p. 248–249, Apr 2010.
- [Akaike 1973]AKAIKE, H. 2nd international symposium on information theory. Akadémiai Kiadó, Budapest, Hungary, p. 267–281, 1973.
- [Ambroise e McLachlan 2002]AMBROISE, C.; MCLACHLAN, G. J. *Selection bias in gene extraction on the basis of microarray gene-expression data*. National Academy of Sciences, 2002. Disponível em: <<https://www.pnas.org/content/99/10/6562.short>>.
- [6]BEWICK, V.; CHEEK, L.; BALL, J. Statistics review 13: receiver operating characteristic curves. *Crit Care*, v. 8, n. 6, p. 508–512, Dec 2004.
- [Choi et al. 2012]CHOI, Y. et al. Predicting the functional effect of amino acid substitutions and indels. *PLOS ONE*, Public Library of Science, v. 7, n. 10, p. 1–13, 10 2012. Disponível em: <<https://doi.org/10.1371/journal.pone.0046688>>.
- [Consortium 2015]CONSORTIUM, T. . G. P. A global reference for human genetic variation. *Nature*, The Author(s) SN -, v. 526, p. 68 EP –, Sep 2015. Article. Disponível em: <<http://dx.doi.org/10.1038/nature15393>>.
- [Cooper et al. 2005]COOPER, G. M. et al. Distribution and intensity of constraint in mammalian genomic sequence. *Genome Res.*, v. 15, n. 7, p. 901–913, Jul 2005.
- [Cortes e Vapnik 1995]CORTES, C.; VAPNIK, V. Support-vector networks. *Machine Learning*, v. 20, p. 273–297, 1995.

- [Forbes et al. 2017]FORBES, S. A. et al. Cosmic: somatic cancer genetics at high-resolution. *Nucleic Acids Research*, v. 45, n. D1, p. D777–D783, 2017. Disponível em: <+ <http://dx.doi.org/10.1093/nar/gkw1121>>.
- [Glossary of Terms 1998]GLOSSARY of Terms. *Mach. Learn.*, Kluwer Academic Publishers, Hingham, MA, USA, v. 30, n. 2-3, p. 271–274, fev. 1998. ISSN 0885-6125. Disponível em: <<http://dl.acm.org/citation.cfm?id=288808.288815>>.
- [Ho 1998]HO, T. K. The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 20, n. 8, p. 832–844, Aug 1998. ISSN 0162-8828.
- [Kent et al. 2002]KENT, W. J. et al. The human genome browser at ucsc. *Genome Research*, Cold Spring Harbor Laboratory Press, v. 12, p. 996–1006, Jun 2002. Disponível em: <<https://doi.org/10.1371/journal.pone.0046688>>.
- [15]KUMAR, P.; HENIKOFF, S.; NG, P. C. Predicting the effects of coding non-synonymous variants on protein function using the sift algorithm. *Nature Protocols*, Nature Publishing Group SN -, v. 4, p. 1073 EP –, Jun 2009. Disponível em: <<http://dx.doi.org/10.1038/nprot.2009.86>>.
- [16]Liu, X.; Wu, J.; Zhou, Z. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, v. 39, n. 2, p. 539–550, April 2009.
- [Meng e Liping 2016]MENG, W.; LIPING, W. ifish: predicting the pathogenicity of human nonsynonymous variants using gene-specific/family-specific attributes and classifiers. *Scientific Reports*, v. 6, n. 31321, 2016.
- [MJ1 et al. 1999]MJ1, L. et al. PSIC: profile extraction from sequence alignments with position-specific counts of independent observations. *Protein Eng.*, v. 12, n. 5, p. 387–394, May 1999.
- [Pierre e Soren 2002]PIERRE, B.; SOREN, B. *Bioinformatics The Machine Learning Approach*. [S.l.]: The MIT Press, 2002.
- [Pollard et al. 2010]POLLARD, K. S. et al. Detection of nonneutral substitution rates on mammalian phylogenies. *Genome Res.*, v. 20, n. 1, p. 110–121, Jan 2010.
- [21]RAMACHANDRAN, G. N.; RAMAKRISHNAN, C.; SASISEKHARAN, V. Stereochemistry of polypeptide chain configurations. *J. Mol. Biol.*, v. 7, p. 95–99, Jul 1963.
- [Ray 2017]RAY, S. *Understanding Support Vector Machine algorithm from examples (along with code)*. Analytics Vidhya, 2017. Disponível em:

<<https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/>>.

[Schwarz 1978]SCHWARZ, G. Estimating the dimension of a model. *Ann. Statist.*, The Institute of Mathematical Statistics, v. 6, n. 2, p. 461–464, 03 1978. Disponível em: <<https://doi.org/10.1214/aos/1176344136>>.

[Stenson et al. 2014]STENSON, P. D. et al. The Human Gene Mutation Database: building a comprehensive mutation repository for clinical and molecular genetics, diagnostic testing and personalized genomic medicine. *Hum. Genet.*, v. 133, n. 1, p. 1–9, Jan 2014.

[Sunyaev et al. 1999]SUNYAEV, S. R. et al. PSIC: profile extraction from sequence alignments with position-specific counts of independent observations. *Protein Eng.*, v. 12, n. 5, p. 387–394, May 1999.

[Svetnik et al. 2004]SVETNIK, V. et al. *Application of Breiman's Random Forest to Modeling Structure-Activity Relationships of Pharmaceutical Molecules*. Springer, Berlin, Heidelberg, 2004. Disponível em: <https://rd.springer.com/chapter/10.1007/978-3-540-25966-4_3>.

[27]WANG, K.; LI, M.; HAKONARSON, H. ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data. *Nucleic Acids Research*, v. 38, n. 16, p. e164–e164, 07 2010. ISSN 0305-1048. Disponível em: <<https://doi.org/10.1093/nar/gkq603>>.

[Yates e Sternberg 2013]YATES, C. M.; STERNBERG, M. J. The effects of non-synonymous single nucleotide polymorphisms (nssnps) on protein–protein interactions. *Journal of Molecular Biology*, v. 425, n. 21, p. 3949 – 3963, 2013. ISSN 0022-2836. Understanding Molecular Effects of Naturally Occurring Genetic Differences. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0022283613004440>>.