CEFET/RJ
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO

MÉTODOS BASEADOS EM HOMOLOGIA E APRENDIZADO DE MÁQUINA PARA IDENTIFICAÇÃO DE PROTEÍNAS ESSENCIAIS

Jéssica da Silva Costa

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Ciência da Computação, Centro Federal de Educação Tecnológica Celso Suckow da Fonseca CEFET/RJ, como parte dos requisitos necessários à obtenção do grau de mestre.

Orientadora:
Prof. Dra. Kele Teixeira Belloze

Rio de Janeiro,

Outubro de 2023

Métodos Baseados em Homologia e Aprendizado de Máquina para Identificação de Proteínas Essenciais

Dissertação de Mestrado em Ciência da Computação, Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ.

Jéssica da Silva Costa

Aprovada por:

Presidente, Prof. Kele Teixeira Belloze, D.Sc. (orientador)

Prof. Eduardo Bezerra da Silva, D.Sc.

Prof. Diogo Antonio Tschoeke, D.Sc.

Prof. Victor Ströele de Andrade Menezes, D.Sc.

Rio de Janeiro,

Outubro de 2023

Ficha catalográfica elaborada pela Biblioteca Central do CEFET/RJ

C837 Costa, Jéssica da Silva
Métodos baseados em homologia e aprendizado de máquina para identificação de proteínas essenciais / Jéssica da Silva Costa. — 2023.
53f. : il. color. , enc.

Dissertação (Mestrado) Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, 2023.
Bibliografia : f. 46-53
Orientadora: Kele Teixeira Belloze

1. Integração de dados (Computação). 2. Proteínas – Identificação. 3. Aprendizado do computador. 4. Farmacologia. 5. Schistosoma mansoni. I. Belloze, Kele Teixeira (Orient.). II. Título.

CDD 005.74

Elaborada pela bibliotecária Tania Mello – CRB/7 nº 5507/04

DEDICATÓRIA

Dedico este trabalho a todas as pessoas que em um algum nível acreditam que ainda é possível fazer Ciência no Brasil.

AGRADECIMENTOS

Agradeço a minha orientadora a Profa. Kele Teixeira Belloze por todo apoio, conselhos orientação e amizade nesses anos de mestrado.

Agradeço aos professores e colegas do Cefet/RJ que me ajudaram com conselhos e conhecimentos que eu precisei durante estes anos de mestrado.

E finalmente quero agradecer a todos que em algum momento da minha trajetória acreditaram em mim e me deram uma oportunidade.

RESUMO

Métodos Baseados em Homologia e Aprendizado de Máquina para Identificação de Proteínas Essenciais

Jéssica da Silva Costa

Orientadora:
Kele Teixeira Belloze

O desenvolvimento de um fármaco costuma ser um processo complexo e demorado. Principalmente na fase inicial, a seleção de um alvo para desenvolvimento de fármacos pode demorar muitos anos. Genes e proteínas essenciais são entidades biológicas responsáveis por processos biológicos de sobrevivência e reprodução dos organismos. Genes e proteínas com relação de ancestralidade, em organismos de espécies diferentes, costumam conservar a função. Além disso, estudos indicam que genes essenciais tendem a ter maior expressão e codificam proteínas que se envolvem em mais interações proteína-proteína. Todas essas características tornam proteínas essenciais potenciais alvos de fármacos. Muitos trabalhos na literatura propõem abordagens biológicas e computacionais para identificação de essencialidade. Diante disso, este trabalho apresenta dois *workflows* para identificação de características de essencialidade em proteínas para alvos de fármacos do organismo alvo *S. mansoni*. Para isso, foram abordados um método baseado em homologia e outro método baseado em aprendizado de máquina com os organismos modelo *S. cerevisiae*, *C. elegans* e *D. melanogaster*. O método baseado em homologia identificou 11 proteínas candidatas a essenciais com o grupo de organismos modelo e o organismo *S. mansoni*. Entre os pares, a maior quantidade de candidatas foi com *S. cerevisiae* onde foram identificadas 726 proteínas candidatas a essenciais. No método baseado em aprendizado de máquina, cerca de 4000 proteínas foram preditas Para esse método foram realizados experimentos com três algoritmos baseados em árvore, XGBoost e Gradient Boosting e Random Forest, e características baseadas em contexto (interação proteína-proteína) e baseadas em sequência. Os algoritmos apontaram melhores valores de recall com o uso da técnica de *Undersampling*. Ainda, 3290 proteínas foram preditas como essenciais pelos três algoritmos trabalhados, o que demonstrou certa semelhança entre os resultados dos algoritmos.

Resumo da Dissertação submetida ao Programa de Pós-graduação em Ciência da Computação do Centro Federal de Educação Tecnológica Celso Suckow da Fonseca CEFET/RJ como parte dos requisitos necessários à obtenção do tıtulo de mestre.

Palavras-chave: Alvos de Fármacos, Homologia, Aprendizado de Máquina, Redes de Interação Proteína-Proteína, Sequências, Essencialidade

Rio de Janeiro,

Outubro de 2023

ABSTRACT

Methods Based on Homology and Machine Learning for Identification of Essential Proteins

Jéssica da Silva Costa

Advisor:
Kele Teixeira Belloze

Drug development is often a complex and time-consuming process. Especially in the initial phase, the selection of a target for drug development can take many years. Essential genes and proteins are biological entities responsible for biological processes of survival and reproduction of organisms. Genes and proteins related to ancestry, in organisms of different species, usually retain their function. Furthermore, studies indicate that essential genes tend to have higher expression and encode proteins that engage in more protein-protein. All these characteristics make proteins potential drug targets. Many works in the literature propose biological and computational approaches for essentiality identification. Therefore, this work presents two workflows for identifying essentiality characteristics in proteins for drug targets of the target organism *S. mansoni*. For this, a method based on homology and another method based on machine learning with Model organisms *S. cerevisiae*, *C. elegans* and *D. melanogaster*. The homology-based method identified 11 essential candidate proteins with the group of model organisms and the organism *S. mansoni*. Among peers, the largest number of candidates it was with *S. cerevisiae* where 726 candidate essential proteins were identified. Full name based on machine learning, around 4000 proteins were predicted for this method experiments were carried out with three tree-based algorithms, XGBoost and Gradient Boosting and Random Forest, and context-based features (protection-protection interaction) and sequence-based. The algorithms showed better recall values using the technique of Undersampling. Furthermore, 3290 proteins were predicted as essential by the three algorithms worked on, which demonstrated a certain similarity between the results of the algorithms.

Abstract of dissertation submitted to Programa de Pós-graduação em Ciência da Computação - Centro Federal de Educação Tecnológica Celso Suckow da Fonseca CEFET/RJ as partial fulfillment of the requirements for the degree of master.

Key-words: Drug Targets, Homology, Machine Learning, Protein-Protein Interaction Networks, Sequences, Essentiality

Rio de Janeiro,

Outubro de 2023

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo I Introdução

I.1 Contextualização e Problema

O desenvolvimento de um novo fármaco desde a ideia original até o lançamento de um produto final é um processo complexo que pode levar de 12 a 15 anos [Hughes et al., 2010]. O processo é demorado porque existem várias fases de testes, chamadas de ensaios de Fase 1, Fase 2 e Fase 3. A Fase 1 compreende a fase de pesquisa por novos alvos de fármacos e compostos moleculares que podem ser potenciais fármacos; a Fase 2 ou pesquisa pré-clinica compreende os primeiros testes em laboratório, podendo ser *in vitro* ou *in vivo*; e a Fase 3 ou clínica é relacionada a testes em humanos para verificação de efetividade. Na maioria das vezes, o fármaco falha nos testes nas Fases 2 e 3, assim, em ambas essas fases são realizados testes de segurança [FDA, 2022; Biswas et al., 2020].

Ainda na Fase 1 pode se levar muitos anos para construir um corpo de evidências de apoio antes da seleção de um alvo para um dispendioso programa de descoberta de fármacos [Hughes et al., 2010; Biswas et al., 2020]. Dada essa complexidade, se faz importante encontrar formas que tornem esse processo, principalmente da Fase 1, menos custoso. Em contrapartida, atualmente há uma vasta coleção de bancos de dados biológicos disponíveis publicamente [Rigden and Fernández, 2023] que auxiliam bastante nas pesquisas *in silico* para o levantamento de alvos para o desenvolvimento de fármacos. Neste contexto, genes e proteínas essenciais se destacam como potenciais alvos.

Os genes e proteínas que são considerados necessários para a sobrevivência ou reprodução de um organismo são classificados como genes essenciais [Zhang and Ren, 2015; Peng et al., 2017; Belloze et al., 2020]. Os genes essenciais também tendem a ter maior expressão e codificam proteínas que se envolvem em mais interações proteína-proteína (PPI) [Wang et al., 2015]. Diante de todas estas características, detectar essencialidade é uma das primeiras tarefas para descoberta de alvos de fármacos [Xie et al., 2020].

A essencialidade dos genes e proteínas costuma ser detectada por meio de experimentos biológicos como *single-gene knockout*, *transposon*, *CRISPR*, entre outros [Nagai et al., 2018; Belloze et al., 2020]. Muitos desses experimentos geram um grande volume de dados, os quais são depositados em bancos de dados públicos como o *Database of Essential Genes* (DEG) [Luo et al., 2020] e o *Online Gene Essentiality database* (OGEE) [Gurumayum et al., 2020].

A esquistossomose é uma doença parasitária aguda e crônica causada por um verme sanguíneo

(platelminto trematódeo) do gênero *Schistosoma*. Mais comumente encontrados em regiões tropicais e subtropicais, a esquistossomose afeta principalmente comunidades pobres e rurais, particularmente populações agrícolas e pesqueiras. No Brasil a espécie mais comum é o *Schistosoma mansoni*. Por ser uma doença com medicamentos antigos e com disponibilidade limitada, faz-se necessária o desenvolvimento de novos fármacos para combater a doença.

Diversos trabalhos na literatura apresentam métodos computacionais (ou pesquisas *in silico*) relacionados à identificação de essencialidade em genes e proteínas. Alguns trabalhos adotam a análise genômica comparativa com base na similaridade e homologia de sequências, são os chamados métodos baseados em homologia [Garcia and Belloze, 2018; Hadizadeh et al., 2018]. Outros trabalhos adotam técnicas híbridas, agregando conceitos biológicos e técnicas computacionais, como aprendizado de máquina, são os chamados métodos baseados em aprendizado de máquina [Garcia et al., 2020; Peng et al., 2017; Zhang et al., 2020]. Diante de diversas pesquisas relacionadas à identificação de essencialidade em genes e proteínas, torna-se relevante estudar diferentes métodos de modo a conhecer com detalhes algumas abordagens para a identificação de essencialidade, principalmente como alvos de fármacos.

I.2 Objetivo

O objetivo deste trabalho é construir e apresentar dois métodos de identificação de essencialidade de genes e proteínas: um método baseado em homologia e um método baseado em aprendizado de máquina. Para o método baseado em aprendizado de máquina são utilizadas duas famílias de entradas: baseadas em contexto e baseadas em sequência. Como objetivos secundários, é considerado nesta pesquisa:

- identificar um conjunto de proteínas essenciais para o organismo alvo *Schistosoma mansoni*;
- disponibilizar workflows que possam ser utilizados com qualquer organismo para gerar dados a respeito de proteínas essenciais que possam ser testadas experimentalmente em bancada.

Esta construção foi conduzida por meio de integração de dados de diversos bancos de dados biológicos, identificação de atributos de essencialidade e a comparação dos métodos computacionais que tragam melhor acurácia na identificação de essencialidade.

I.3 Metodologia

A metodologia proposta para este trabalho apresenta dois métodos de identificação de essencialidade de proteínas: i) método baseado em homologia, utilizando o conceito de ortologia; e ii) método baseado em aprendizado de máquina, utilizando o conceito de redes de interação de

proteína-proteína (PPI) e características baseadas em sequência. Para tanto, dois *workflows* computacionais foram construídos. Além disso, os resultados de ambos os métodos são analisados com o intuito de identificar a interseção deles.

Para o método baseado em homologia, o *workflow* considera a geração de listas de proteínas candidatas a essenciais baseadas em ortologia entre pares de organismos (organismo modelo e alvo) e entre o grupo de organismos (organismos modelo e alvo). Para o método baseado em aprendizado de máquina, o *workflow* tem como tarefa a construção de modelos de aprendizado de máquina baseado em medidas de centralidade e *clustering* e característica das sequências para detecção de essencialidade em proteínas.

Para a organização dos *workflows*, as tarefas de construção de bases de dados integradas fundamentaram-se no uso de dados públicos advindos dos bancos de dados DEG (genes e proteínas essenciais), Ensembl (proteomas) e STRING (redes de PPI). Como experimento para a validação da metodologia, foram empregados dados dos organismos modelo *Saccharomyces cerevisiae*, *Caenorhabditis elegans*, *Drosophila melanogaster*, *Mus musculus* e do organismo alvo de estudo *Schistosoma mansoni*.

I.4 Resultados

O método baseado em homologia identificou cerca de 11 proteínas candidatas a essenciais com o grupo de organismos modelo e o alvo. Entre os pares, foram identificadas 726 proteínas candidatas a essenciais com *S. cerevisiae*, 35 com *C. elegans* e 223 com *D. melanogaster*.

Com relação ao método baseado em aprendizado de máquina, experimentos preliminares com características baseadas em contexto (PPI) apontaram melhores resultados para o modelo *Random Forest* com o uso de técnicas de balanceamento. Diante disso, novos experimentos foram realizados com dois modelos baseados em árvore e o acréscimo de características baseadas em sequência. Os novos experimentos apontaram melhores valores de *recall* com o uso da técnica de *Undersampling*. Cerca de 4000 proteínas foram preditas essenciais nos modelos *XGBoost*, *Random Forest* e *Gradient Boosting*. Em comum os algoritmos apresentaram 3297 proteínas como essenciais.

I.5 Organização do Trabalho

Além desta introdução, este trabalho está organizado em mais cinco capítulos. O capítulo II aborda o referencial teórico necessário para o entendimento deste trabalho. O capítulo III apresenta trabalhos relacionados ao tema de essencialidade de genes e proteínas que utilizaram ambos os métodos baseado em homologia e baseado em aprendizado de máquina, descrevendo características principais e distinções. O capítulo IV apresenta de forma detalhada a metodologia, descrevendo as atividades dos *workflows*, os dados de entrada, as bases de dados geradas e os algoritmos trabalhados

nesta pesquisa. O capítulo V apresenta os resultados obtidos para os dois métodos, especificamente relacionados às proteínas do organismo *Schistosoma mansoni*. Por fim, o capítulo VI apresenta a conclusão do trabalho e futuras oportunidades.

Capítulo II Referencial Teórico

Este capítulo apresenta conceitos e termos importantes para o entendimento dos métodos propostos nesta pesquisa. Assim, a primeira seção apresenta conceitos sobre esquistossomose, a segunda seção conceitos sobre homologia, a terceira sobre organismos-modelo, a quarta sobre redes de interação proteína-proteína, a quinta sobre características baseadas em sequência e a sexta sobre bancos de dados biológicos. A sétima e a oitava seções apresentam conceitos computacionais relacionados a medidas de centralidade e *clustering* em grafos e aprendizado de máquina, respectivamente.

II.1 Esquistossomose

A esquistossomose é uma doença parasitária aguda e crônica causada por um verme sanguíneo (platelminto trematódeo) do gênero *Schistosoma*. Mais comumente encontrados em regiões tropicais e subtropicais, a esquistossomose afeta principalmente comunidades pobres e rurais, particularmente populações agrícolas e pesqueiras. Existem duas formas principais de esquistossomose – intestinal e urogenital – causadas por cinco espécies principais de vermes sanguíneos. A espécie *Schistosoma mansoni* causa uma esquistossomose intestinal nas regiões da África, Oriente Médio, Caribe, Brasil, Venezuela e Suriname [WHO, 2022].

Os sintomas da esquistossomose são causados pela reação do corpo aos ovos dos vermes. A esquistossomose intestinal pode resultar em dor abdominal, diarreia e sangue nas fezes. O aumento do fígado é comum em casos avançados, nesses casos, também pode haver aumento do baço. Em crianças, a esquistossomose pode causar anemia, retardo de crescimento e redução da capacidade de aprendizado, embora os efeitos geralmente sejam reversíveis com o tratamento [WHO, 2022].

O controle da esquistossomose é baseado no tratamento em larga escala de grupos populacionais em risco, acesso a água potável, saneamento melhorado, educação sobre higiene e controle de caramujos (hospedeiro da espécie *S. mansoni*). O tratamento em larga escala é realizado com o medicamento Praziquantel, efetivo e de baixo custo. Uma limitação importante para o controle da esquistossomose tem sido a disponibilidade limitada de Praziquantel, particularmente para o tratamento de adultos. Dados de 2019 mostram que 44,5% das pessoas que necessitam de tratamento foram alcançadas globalmente, com uma proporção de 67,2% de crianças em idade escolar

que necessitam de quimioterapia preventiva para esquistossomose sendo tratadas [WHO, 2022].

II.2 Organismos Modelo

Organismos modelo são espécies amplamente usadas na pesquisa científica para estudar processos biológicos e que possuem seus genes e proteínas essenciais conhecidos e anotados [Palladino, 2009]. Eles são geralmente definidos como espécies não humanas que são extensivamente estudadas para entender uma série de fenômenos biológicos. Estudos genéticos permitem a análise sistemática de muitas características de um organismo, e muitos dos dados, teorias e processos estudados em detalhes podem servir como modelos para outros organismos, geralmente mais complexos, devido à homologia de genes [Ankeny and Leonelli, 2020; Irion and Nüsslein-Volhard, 2022]. Em geral organismos modelo possuem crescimento mais rápido e são facilmente cultivados em laboratório, o que facilita utilizá-los em pesquisas.

O grupo de organismos modelo mais amplamente reconhecido, compilado pelos Institutos Nacionais de Saúde dos EUA, compreende apenas 13 espécies, incluindo a levedura *Saccharomyces cerevisiae*, a mosca da fruta *Drosophila melanogaster*, o verme nematoide *Caenorhabditis elegans*, a planta *Arabidopsis thaliana*, o peixe-zebra *Danio rerio* e o rato *Mus musculus* [Ankeny and Leonelli, 2020].

II.3 Homologia

Homologia diz respeito a relação de descendência comum entre quaisquer seres, sem maiores especificações do cenário evolutivo. Assim, genes ou proteínas que possuem este tipo de relação de descendência são chamados de homólogos [Koonin, 2005]. As proteínas homólogas podem ter similaridade a nível sequencial (sequência de bases nitrogenadas) ou mesmo estrutural (estrutura tridimensional). As sequências homólogas podem ser muito similares em alguns aspectos, idênticas ou mesmo diferentes devido a mutações. Uma prática muito comum é assumir que duas sequências são homólogas se forem mais de 30% idênticas em todo o seu comprimento [Pearson, 2013].

Uma das formas para encontrar proteínas homólogas é por meio de técnicas computacionais para comparação de sequências, as quais se baseiam na similaridade entre sequências de genes e proteínas [Santos Filho and Alencastro, 2003]. O método baseado em similaridade busca por sequências similares em bancos de dados a partir de uma ou mais sequências de interesse. Por exemplo, essa comparação pode ser realizada par a par em um aminoácido de cada proteína ou por meio da maior sequência de aminoácidos de uma proteína que pode coincidir com outra [Needleman and Wunsch, 1970]. A estratégia mais usada neste caso é com o uso dos softwares do pacote Basic Local Alignment Search Tool (BLAST) [Camacho et al., 2009].

No contexto da homologia, existem duas classificações muito importantes, a ortologia e a paralogia. Na ortologia, genes ou proteínas vêm de especiação ou descendência vertical, o que ocorre em espécies diferentes, enquanto na paralogia os mesmos vêm de duplicação, ou seja, ocorrem em sua maioria em uma mesma espécie. Ortólogos tendem a conservar a função, enquanto parálogos tendem a diferentes funções [Koonin, 2005; Altenhoff and Dessimoz, 2012].

Genes ortólogos são importantes para a compreensão da biologia molecular porque se é conhecida uma determinada função do gene ortólogo, é possível inferir, ao menos de forma provisória, a função do gene recém-sequenciado e sua proteína [Moreira, 2015]. No presente trabalho, a aplicação do método baseado em homologia considera, portanto, o conceito de ortologia. Um exemplo de ortologia acontece com a hemoglobina. A hemoglobina é uma proteína presente nas hemácias de células sanguíneas de organismos vertebrados responsável principalmente pelo transporte de oxigênio. A hemoglobina presente no *Mus musculus* (camundongo) e no *Homo sapiens* (ser humano) são ortólogas e parólogas. A figura II.1 exemplifica a relação de ortologia entre o camundongo e o ser humano, pois além de possuírem um ancestral em comum, também possuem a mesma função.

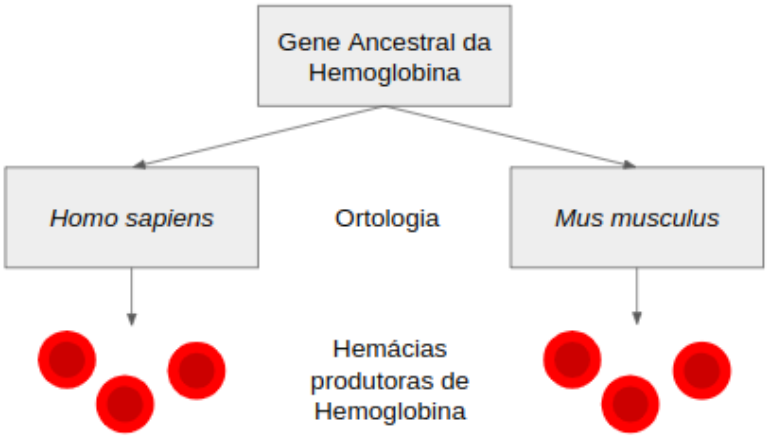

Figura II.1: Ortologia da hemoglobina

II.3.1 Ferramenta de Verificação de Ortologia

Uma forma de verificação de ortologia é por meio de ferramentas computacionais que trabalham com sequências de genes e proteínas. A OrthoMCL 2.0 e a Orthofinder, desenvolvidas nas linguagens Perl e Python respectivamente, são ferramentas para agrupar proteínas em grupos ortólogos baseados na similaridade da sequência, ou seja, determinam relações filogenéticas entre ortólogos [Fischer et al., 2011; Emms and Kelly, 2019]. Para este trabalho, foi realizada a comparação das ferramentas do ponto de vista de suas execuções e suas principais diferenças. Tal comparação apoiou a tomada de decisão sobre o uso da ferramenta Orthofinder uma vez que a literatura apresenta diversos trabalhos que utilizam tanto uma quanto outra.

Em relação a busca de similaridade de sequência, a ferramenta Orthofinder utiliza o algoritmo DIAMOnD para alinhamento de sequências. Esse algoritmo é baseado em dupla indexação, sendo 20000 vezes mais rápido que o BLASTX (modalidade do pacote BLAST que busca em bancos de dados de proteínas usando sequências de nucleotídeos traduzidas) em pequenas leituras e com grau de sensibilidade similar [Emms and Kelly, 2019; Buchfink et al., 2014]. A ferramenta OrthoMCL utiliza o pacote BLAST, o que torna esse passo da execução mais lento [Fischer et al., 2011]. Outra característica que torna a Orthofinder mais rápida é o uso de processamento paralelo por meio do uso de *threads*.

A OrthoMCL 2.0 faz uso de um banco de dados relacional, que pode ser o MySQL ou o Oracle [Fischer et al., 2011], enquanto a Orthofinder faz uso de arquivos em formato texto. Outra característica das ferramentas é em relação a usabilidade. A Orthofinder realiza todos os seus processos com a execução de um único *script*, enquanto a OrthoMCL realiza a execução por meio de vários *scripts* de forma sequencial utilizando os dados resultantes de uma etapa na próxima, ou seja, um *script* depende do sucesso da execução do anterior ou por um *script* único. Mesmo a execução do MCL (*Markov Cluster Algorithm*), um algoritmo de agrupamento muito utilizado em proteínas [Enright et al., 2002] comum às duas ferramentas, na Orthofinder é automático e na OrthoMCL precisa ser executado manualmente.

II.4 Redes de Interação Proteína-Proteína

Proteínas não agem de forma independente, elas dependem de uma complexa rede de interação com outras proteínas, ácidos nucleicos e pequenas moléculas. As interações podem mudar conforme o tempo e o estado celular, o que permite que as proteínas realizem suas funções [Lehne and Schlitt, 2009; Patil, 2019]. Essa complexa rede de interação proteína-proteína (PPI) é muito importante para compreender processos biológicos, por exemplo, das proteínas essenciais [Patil, 2019; Li et al., 2021]. Ainda sobre essencialidade, um estudo recente sobre redes PPI revela que as proteínas altamente conectadas são mais propensas a serem essenciais [Shen et al., 2022].

Existem várias estratégias experimentais para mapear PPIs, como *yeast two-hybrid assay* (Y2H), que mede interações físicas diretas em células, e, espectrometria de massa de purificação por afinidade, que mede a composição de complexos proteicos [Liu et al., 2020; Patil, 2019]. Essas técnicas geram dados de PPI que podem ser encontrados em bancos de dados públicos como o STRING [Szklarczyk et al., 2022] e o *Database of Interacting Proteins* (DIP) [Xenarios et al., 2000]. A figura II.2 mostra uma representação de uma rede de PPI ou interatoma do organismo *Saccharomyces cerevisiae*.

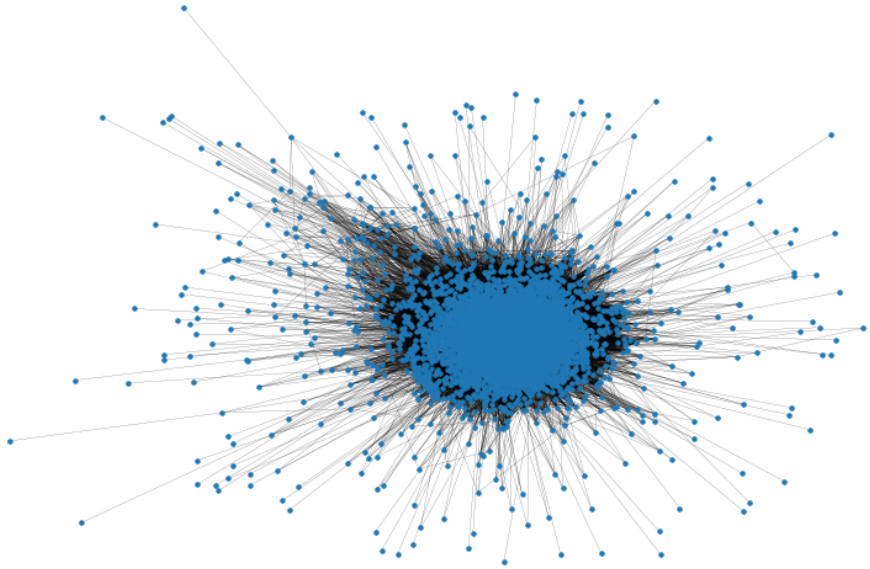

Figura II.2: Rede de PPI do organismo *S. cerevisiae*

II.5 Características Baseadas em Sequência

Características baseadas em sequência são métricas calculadas com base na sequência de um gene ou de uma proteína e trazem informações sobre elas. Elas são muito utilizadas como entradas de algoritmos de Aprendizado de Máquina [Nigatu et al., 2017; Campos et al., 2019]. Especificamente para proteínas, exemplos de tipos de métricas englobam:

- **Porcentagem de Aminoácidos**: apresenta um percentual de cada tipo de aminoácido da proteína.
- **Tamanho da Sequência**: mede a quantidade de aminoácidos da sequência.
- **Aromaticidade**: frequência relativa dos aminoácidos fenilalanina (Phe), triptofano (Trp) e tirosina (Tyr) [Lobry and Gautier, 1994].
- **Fração da Estrutura Secundária**: representa um índice de uma sequência formar uma estrutura secundária em beta-folha, alfa-hélice ou dar uma volta de transição baseada nos aminoácidos que ela possui [Shingate and Sowdhamini, 2012].

II.6 Bancos de Dados Biológicos

Bancos de dados biológicos armazenam diversos dados relativos ao sequenciamento genético de DNA, RNA, expressão gênica, proteínas e outras entidades biológicas, e assim se tornam ferramentas muito importantes para a Bioinformática. Com uma vasta coleção de bancos de dados biológicos disponíveis publicamente [Rigden and Fernández, 2023], o maior objetivo dos bancos

de dados biológicos não é apenas armazenar, organizar e compartilhar dados, mas também prover aplicações que trocam e integram dados de várias fontes de forma automatizada [Zou et al., 2015].

Os bancos de dados biológicos possuem classificação relacionada ao tipo de dado que armazenam. Os bancos primários são relativos a dados ou medidas gerados a partir de fontes biológicas originalmente armazenadas sob a forma de sequências de nucleotídeos ou de aminoácidos [Júnior et al., 2022]. Um exemplo de banco primário é o Ensembl [Cunningham et al., 2021]. A figura II.3 apresenta uma sequência em formato fasta da proteína YDL245C do organismo *Saccharomyces cerevisiae* (uma levedura), armazenada no Ensembl.

```
>YDL245C pep chromosome:R64-1-1:IV:11657:13360:-1 gene:YDL245C transcript:YDL245C_mRNA
gene_biotype:protein_coding transcript_biotype:protein_coding gene_symbol:HXT15
description:Putative transmembrane polyol transporter; supports growth on and uptake of
mannitol, sorbitol and xylitol with moderate affinity when overexpressed in a strain deleted
for hexose family members; minor hexose transport activity when overexpressed in a similar
strain; similarity to hexose transporters; expression is induced by low levels of glucose
and repressed by high levels of glucose [Source:SGD;Acc:S000002404]
MASEQSSPEINADNLNSSAADVHVQPPGEKEWSDGFYDKEVINGNTPDAPKRGFLGYLII
YLLCYPVSFGGFLPGWDSGITAGFINMDNFKMNFGSYKHSTGEYYLSNVRMGLLVAMFSV
GCSIGGVAFARLADTLGRRLAIVIVVLVYMVGAIIQISSNHKWYQYFVGKIIYGLGAGGC
SVLCPMLLSEIAPTDLRGGLVSLYQLNMTFGIFLGYCSVYGTRKYSNTAQWRIPVGLCFL
WALIIIVGMLLVPESPRYLIECERHEEACVSIAKINKVSPEDPWVLKQADEINAGVLAQR
ELGEASWKELFSVKTKVLQRLITGILVQTFLQLTGENYFFFYGTTIFKSVGLTDGFETSI
VLGTVNFFSTIIAVMVVDKIGRRKCLLFGAASMMACMVIFASIGVKCLYPHGQDGPSSKG
AGNAMIVFTCFYIFCFATTWAPVAYIVVAESFPSKVKSKAMSISTAFNWLWQFLIGFFTP
FITGSIHFYYGYVFVGCLVAMFLYVFFFLPETIGLSLEEIQLLYEEGIKPWKSASWVPPS
RRGASSRETEAKKKSWKEVLKFPKSFN
```

Figura II.3: Sequência em formato fasta da proteína YDL245C de *S. cerevisiae*. O formato fasta apresenta uma descrição da sequência (linha distinguida pelo símbolo maior-que '>' no início) seguida por linhas que contêm a sequência em si.

Os bancos de dados secundários armazenam resultados de análises feitas a partir da seleção e da escolha de dados oriundos de bancos de dados primários [Júnior et al., 2022]. Muitas vezes, esses bancos de dados fornecem referências cruzadas para interoperabilidade entre bancos de dados [Chavan et al., 2011]. Ainda, há os bancos especializados que trabalham com um interesse particular de pesquisa, como essencialidade de genes e proteínas ou redes de PPI ou um organismo em particular, como o SGD (*Saccharomyces Genome Database*), um banco especializado na levedura [Cherry et al., 2011].

II.6.1 Bancos de Dados de Genes Essenciais

No contexto de bancos de dados biológicos especializados, há aqueles que armazenam dados de essencialidade de genes e proteínas, como o *Database of Essential Genes* (DEG) [Luo et al., 2020] e o *Online Gene Essentiality database* (OGEE) [Gurumayum et al., 2020]. Para este trabalho, somente o DEG foi utilizado dada sua maior flexibilidade com identificadores que apontam para os outros bancos de dados. Por exemplo, ele possui o identificador *locus* que é muito utilizado em diversos proteomas. O DEG, criado em 2003, armazena entidades genômicas essenciais, como genes codificadores de proteínas e RNAs não codificantes, entre bactérias, arquéias e eucariotos [Luo et al., 2020].

Como comentando na seção I.1, a essencialidade dos genes e proteínas costuma ser detectada por meio de experimentos biológicos como *single-gene knockout*, *transposon*, *CRISPR* (*Clustered Regularly Interspaced Short Palindromic Repeats*), entre outros [Nagai et al., 2018; Belloze et al., 2020]. Especificamente na versão mais recente do DEG - versão 15, o *CRISPR* esteve muito presente em experimentos de organismos eucarióticos [Belloze et al., 2020; Luo et al., 2020]. O DEG disponibiliza os dados de forma segmentada por grupos de organismos, além de dividir em arquivos separados as listagens de sequências de genes e proteínas e anotações [Luo et al., 2020].

II.6.2 Bancos de Dados de Redes de Interação Proteína-Proteína

Os bancos de dados de redes de PPI são representados como grafos não direcionados, ou seja, a rede de PPI é definida como um grafo $G = (V, E)$, composta pelo nós V e arestas E, onde cada nó $v \in V$ representa uma proteína e cada aresta $(u, v) \in E$ representa uma interação da proteína u e da proteína v [Shen et al., 2022]. A teoria dos grafos é um ajuste natural para investigações biológicas de relacionamentos, padrões e complexidade, que auxilia biólogos a definir o que devem procurar entre as representações de relacionamentos [Jungck and Viswanathan, 2015].

Neste trabalho, o banco de dados utilizado para extração dos dados de PPI é o STRING. O banco de dados STRING integra associações conhecidas e previstas entre proteínas de diversos organismos, incluindo interações físicas e associações funcionais. A versão atual, 12.0, contém mais de 14.000 organismos [Szklarczyk et al., 2022] (até o momento dessa pesquisa a quantidade de organismos é igual a versão anterior, 11.5).

II.7 Medidas de Centralidade e *Clustering* em Grafos

Dado que muitas redes de PPI são complexas, uma forma possível de caracterizá-las é por meio de medidas, como as de centralidade e de *clustering*. A centralidade é um dos princípios fundamentais da análise de rede. Ela mede o quão "central" um nó está na rede. Isso é usado como uma estimativa de sua importância na rede [Golbeck, 2013]. Tal medida pode ser abstraída para redes de proteínas para compreender quais proteínas têm maior importância (central) na rede.

Freeman [1978] estudou muito sobre centralidade e abordou algumas medidas principais, são elas: centralidade de grau (*Degree Centrality*), centralidade de proximidade (*Closeness Centrality*) e centralidade de intermediação (*Betweenness Centrality*). A medida de centralidade de grau está relacionada à quantidade de ligações que o nó tem. Os valores da centralidade de grau são normalizados dividindo-se pelo grau máximo possível em um grafo simples $n-1$ onde n é o número de nós em um grafo G [Hagberg et al., 2008].

A medida de centralidade de proximidade indica o quão próximo um nó está de todos os outros nós da rede [Newman, 2010]. A centralidade de proximidade de um nó u é o recíproco da distância

média do caminho mais curto para u sobre todos os $n-1$ nós alcançáveis [Hagberg et al., 2008]. Sua fórmula é definida pela equação II.1, onde $d(v,u)$ é a distância do caminho mais curto entre v e u, e $n-1$ é o número de nós alcançáveis de u.

$$C(u) = \frac{n-1}{\sum_{v=1}^{n-1} d(v,u)} \tag{II.1}$$

A medida de centralidade de intermediação mede a importância de um nó para os caminhos mais curtos através da rede [Golbeck, 2013]. Sua fórmula é definida pela equação II.2, onde centralidade de intermediação de um nó é a soma da fração dos caminhos mais curtos de todos os pares que passam por onde V é o conjunto de nós, $\sigma(s,t)$ é o números de menores (s,t)-caminhos, e $\sigma(s,t|v)$ é o número desses caminhos passando por algum nó v que não seja s,t. Se $s=t$, $\sigma(s,t)=1$, e se $v \in s,t$, $\sigma(s,t|v)=0$ [Hagberg et al., 2008].

$$c_B(v) = \sum_{s,t \in V} \frac{\sigma(s,t|v)}{\sigma(s,t)} \tag{II.2}$$

Bonacich [1987] propôs uma medida de centralidade relacionada aos autovetores, a centralidade de autovetor (*Eigenvector Centrality*), que mede a importância de um nó enquanto considera a importância de seus vizinhos. Assim a centralidade do autovetor dá a cada vértice uma pontuação proporcional à soma das pontuações de seus vizinhos [Golbeck, 2013; Newman, 2010]. A centralidade do autovetor para o nó i é o i-ésimo elemento do vetor x definido pela equação II.3, onde A é a matriz de adjacência do grafo G com autovalor λ [Hagberg et al., 2008].

$$Ax = \lambda x \tag{II.3}$$

Além das medidas de centralidade, outra medida importante é o coeficiente de *clustering* que representa o grau em que os nós em um grafo tendem a se agrupar. Para grafos não ponderados, o agrupamento de um nó u é definido pela equação II.4 que representa a fração de possíveis triângulos através desse nó existente, onde $T(u)$ é o número de triângulos através do nó u e $deg(u)$ é o grau de u [Hagberg et al., 2008].

$$c_u = \frac{2T(u)}{deg(u)(deg(u)-1)}, \tag{II.4}$$

A análise de cluster de ligação única, por exemplo, é uma ferramenta amplamente utilizada para identificar associações em dados biológicos, como aqueles baseados em *microarrays* que identificam padrões coordenados de agrupamentos sequenciais de genes sendo ativados ou desativados de forma síncrona (ativados ou inibidos) [Jungck and Viswanathan, 2015].

II.8 Aprendizado de Máquina

O aprendizado de máquina (AM), em inglês *Machine Learning* (ML), é uma subárea da Inteligência Artificial que lida com o aprendizado automatizado de máquinas sem ser explicitamente programado. Ele se concentra na realização de previsões baseadas em dados e possui várias aplicações no campo da bioinformática [Shastry and Sanjay, 2020].

O aprendizado de máquina supervisionado é caracterizado pela ideia de supervisor, onde a principal tarefa é prover um agente com uma medida precisa de erro (comparável com valores de saída). Nos algoritmos atuais é realizado um treinamento com valores de entrada, que são previamente rotulados e valores de saída que serão comparados aos de entrada para calcular o erro [Bonaccorso, 2017]. No aprendizado não-supervisionado não existe o agente ou dados rotulados, o conjunto de elementos é agrupado de acordo com sua similaridade ou medida de distância [Bonaccorso, 2017]. Esses são algoritmos mais utilizados em problemas que envolvam agrupamento de itens semelhantes. O aprendizado por reforço é baseado no retorno do ambiente. Esse retorno pode ser positivo, chamado de recompensa, ou mesmo negativo. Esta abordagem baseia-se na ideia que o agente aprenderá com os retornos imediatos e recompensas acumuladas [Bonaccorso, 2017].

Existem muitos algoritmos de aprendizado de máquina supervisionados, como regressão logística, máquinas vetores suporte, naïve bayes, redes neurais, árvores de decisão, entre outros. Mas especificamente os algoritmos baseados em árvore são adequados para para o problema deste trabalho devido à sua popularidade na medicina e uso extensivo na computação de dados biológicos [Shazadi, 2021]. Em muitos trabalhos, os resultados com *Random Forest*, um algoritmo baseado em árvore, apresenta êxito nos resultados em problemas biológicos que possuem dados tabulares.

II.8.1 Algoritmos Baseados em Árvore

A árvore de decisão é um método de mineração de dados e algoritmo de aprendizado de máquina comumente usado para estabelecer sistemas de classificação com base em várias co-variáveis ou para desenvolver algoritmos de previsão para uma variável de destino [Kingsford and Salzberg, 2008]. A figura II.4 apresenta a estrutura de uma árvore de decisão. A partir de um nó raiz, vários nós de decisão vão se ramificando até chegarem a nós terminais. O uso de algoritmos baseados em árvore está bem presente em estudos biológicos em bases tabulares, tanto genômicos quanto da saúde de forma geral [Hafez et al., 2023; Bhardwaj et al., 2024].

Figura II.4: Estrutura de uma árvore de decisão

Existem diversos algoritmos baseados em árvores, como o CART (*Classification And Regression Tree*) e até aqueles que juntam várias árvores para combinar resultados, como as florestas de decisão. Métodos que combinam resultados são chamados de métodos de *Ensemble* e podem ser treinados de forma paralela (*Bagging*) e combinados ao final ou treinados sequencialmente (*Boosting*) com melhora iterativa, um modelo depende do anterior.

O *Random Forest* é um dos algoritmos que trabalha com *Bagging* e foi desenvolvido por Leo Breiman e Adele Cutler [Breiman, 2001; Cutler et al., 2012]. Como o próprio nome sugere é um conjunto baseado em árvores (floresta) com cada árvore dependendo de uma coleção de variáveis aleatórias [Cutler et al., 2012]. Além disso ele trabalha com um conceito chamado de *bootstrap*, uma técnica de reamostragem com substituição de uma fonte de dados para estimar um parâmetro populacional. É um algoritmo amplamente utilizado há mais de dez anos em problemas genômicos, como predição, seleção de variável, associação genética entre outros [Chen and Ishwaran, 2012].

O *Gradient Boosting* é um dos algoritmos de *Ensemble* que trabalha com *Boosting*. O nome Gradiente vem da ideia de minimizar a perda do modelo anterior usando um procedimento semelhante ao gradiente descendente. Embora outros autores (Freund e Schapire) tenham trabalhado com algoritmos de *Boosting*, a estrutura do *Gradient Boosting* foi desenvolvida por Friedman e chamada de *Gradient Boosting Machines*, tarde chamado apenas de *Gradient Boosting* [Freund and Schapire, 1997; Friedman, 2001].

O *Extreme Gradient Boosting* ou *XGBoost* é um dos algoritmos de *Boosting* baseado em árvore que usa a estrutura do *Gradient Boosting*. Desenvolvido como um projeto de pesquisa na Universidade de Washington por Tianqi Chen e Carlos Guestrin em 2011, o *XGBoost* possui melhorias algorítmicas que auxiliam na eficiência e escalabilidade. O algoritmo funciona vezes mais rápido do que as soluções populares existentes em uma única máquina e escala para bilhões de exemplos

em distribuição ou configurações limitadas de memória [Chen and Guestrin, 2016]. Embora seja mais recente, é um algoritmo que pode ser também utilizado em problemas genômicos, inclusive em predição de predição de expressão gênica [Li et al., 2019].

Existem diversas bibliotecas que implementam algoritmos de árvore. Uma biblioteca bem conhecida é a Scikit-learn, de código aberto para a linguagem de programação Python [Pedregosa et al., 2011]. Ela possui implementações de diversos algoritmos de pré-processamento de dados e aprendizado de máquina, desde modelos supervisionados a não-supervisionados.

Recentemente o Google lançou uma biblioteca do *Tensorflow* focada em algoritmos de árvore, a *Tensorflow Decision Forests*. Uma observação importante sobre a biblioteca é que ela é um *wrapper*, ela disponibiliza em Python outra biblioteca chamada *Yggdrasil Decision Forests*. Esta é uma biblioteca para o treinamento, serviço e interpretação de modelos de floresta de decisão, direcionado tanto para pesquisa quanto para trabalho de produção, implementado em C++, e disponível em C++, interface de linha de comando, Python (sob o nome *TensorFlow Decision Forests*), JavaScript , Go e Google Sheets (sob o nome Simple ML for Sheets)[Guillame-Bert et al., 2023].

A biblioteca *XGBoost* é uma biblioteca otimizada e distribuída que implementa o algoritmo *Extreme Gradient Boosting* projetada para ser altamente eficiente, flexível e portátil [Chen and Guestrin, 2016]. Ela está disponível em algumas linguagens, como Python, Julia, Ruby, R, entre outras.

II.8.2 Avaliação de Algoritmos de Classificação

Após o treinamento de algoritmos de aprendizado de máquina, é importante avaliar os modelos através de métricas para avaliar o quão bem eles classificaram as entradas. Essas métricas são chamadas de métricas de avaliação de desempenho [Vakili et al., 2020]. Especificamente em algoritmos de classificação binária são utilizadas métricas que avaliam as classes positiva e negativa. Para isso são avaliados os seguintes resultados dos modelos de classificação:

- *True Positive* ou TP, a classe positiva classificada corretamente;
- *True Negative* ou TN, a classe negativa classificada corretamente;
- *False Positive* ou FP, a classe negativa classificada como positiva incorretamente;
- *False Negative* ou FN, a classe positiva classificada como negativa incorretamente.

As métricas mais utilizadas para avaliação do desempenho desses algoritmos são: acurária, precisão, *recall* e *F1-score*. A primeira métrica é a acurácia que basicamente é a razão entre a quantidade de acertos pelo número total de observações. Embora seja muito utilizada, a acurácia

não é uma métrica adequada para *datasets* desbalanceados. A acurácia é definida pela fórmula II.5:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{II.5}$$

A precisão avalia proporção de identificações positivas correta. A precisão é definida pela fórmula II.6, representada pelo número de positivos verdadeiros dividido pela soma de positivos verdadeiros e falsos positivos [Vakili et al., 2020]:

$$Precision = \frac{TP}{TP + FP} \tag{II.6}$$

O *recall* avalia a proporção de positivos verdadeiros identificada corretamente. Na verdade, fora das observações que são realmente positivos, quantos deles foram previstos pelo algoritmo. De acordo com fórmula II.7, o recall é igual ao número de positivos verdadeiros divididos pela soma dos positivos verdadeiros e falsos negativos [Vakili et al., 2020]:

$$Recall = \frac{TP}{TP + FN} \tag{II.7}$$

Em *datasets* desbalanceados, as métricas precisão e *recall* tornam-se muito úteis para avaliar justamente a classe positiva minoritária. No entanto, em muitos casos ao aumentar uma métrica, você diminui a outra, por isso é importante avaliar o que é menos desfavorável ao modelo. Caso sejam falsos negativos, opta-se pela precisão, já os falsos positivos, opta-se pelo *recall*. Uma forma de avaliar as duas métricas em conjunto é por meio do cálculo de $F1$, apresentado na fórmula II.8, que realiza uma média harmônica entre a precisão e o *recall*.

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \tag{II.8}$$

Capítulo III Trabalhos Relacionados

Uma revisão foi realizada na concepção deste trabalho pra identificar os estudos mais relevantes relacionados a predição de essencialidade de proteínas. As bases de busca foram Scopus, IEEE e Google Scholar. Em geral, as strings de busca combinavam as palavras *homology*, *Protein-protein interaction* ou *Machine Learning* juntamente com as expressões *essential gene*, *essential protein* ou *essentiality*. Os estudos abrangeram diversas abordagens de levantamento e análise de dados relacionados a homologia, redes de interação proteína-proteína (PPI), expressão gênica e técnicas computacionais de aprendizado de máquina dos últimos 5 anos.

Em geral os alvos de estudo para detecção de essencialidade são bactérias ou organismos eucariotos, existindo especificamente abordagens aplicadas a humanos. No quesito abordagem técnica, a maioria dos trabalhos utilizou abordagens híbridas, em que se usavam diversos tipos de entradas biológicas com alguma abordagem computacional de modo a apresentar resultados mais íntegros. As seções seguintes caracterizam os trabalhos selecionados conforme a abordagem, revisões sistemáticas de métodos, homologia e técnicas híbridas (englobam computação e biologia).

III.1 Revisões de Métodos

Peng et al. [2017] apresentaram uma revisão sobre abordagens que têm sido usadas para prever genes essenciais: Pesquisa de Homologia e Métodos Baseados em Análise Evolutiva e Aprendizado de Máquina. No artigo eles descrevem os tipos de entradas em algoritmos preditivos, as baseadas em contexto como as redes de PPI, expressão gênica e Gene Ontology, e as baseadas em sequência como tamanho do gene, GC content, entre outras. Além disso, os autores avaliaram alguns os servidores de predição de genes essenciais bacterianos online disponíveis, como o CEG_Match e o Geptop, com base nos conjuntos de genes essenciais validados experimentalmente de 30 bactérias do DEG. O artigo se propõe a ser um guia de referência rápida para microbiologistas interessados em genes essenciais.

Aromolaran et al. [2021] apresentaram alguns métodos, suas vantagens e desvantagens na identificação de genes essenciais: experimental, por homologia, baseado em restrição (redes metabólicas) e aprendizado de máquina. O artigo argumenta que a vantagem do método baseado em homologia é que a sequência genômica de um organismo maior que 70% de identidade é suficiente para pre-

ver a essencialidade dos genes, no entanto, a abordagem é limitada a ortólogos conservados entre as espécies, que muitas vezes é uma pequena proporção do genoma alvo. Para aprendizado de máquina, o artigo destaca a importância e a diversidade das entradas que podem ser usadas e destaca a *Gene Ontology*. No entanto, os dados disponíveis da rede biológica de estudos experimentais e computacionais são incompletos e contêm muitos falsos positivos e falsos negativos.

Ainda na linha de revisão de abordagens, Wunderlich [2022] apresenta uma lista atualizada de possíveis alvos de fármacos para o organismo *P. falciparum*, causador da malária, através da consolidação de vários bancos de dados. A pesquisa inclui as publicações mais recentes de dados e fornece uma visão geral atualizada sobre os substratos, localização, função, classificação, essencialidade e ortólogos humanos dos transportadores do *P. falciparum*. Esses transportadores sem os ortólogos humanos podem ser novos alvos promissores para o desenvolvimento terapêutico.

III.2 Homologia

O trabalho de Hadizadeh et al. [2018] previu e caracterizou alvos de drogas putativas da família de bactérias *Enterobacteriaceae* empregando um método computacional baseado em homologia. Os alvos das drogas putativas finais foram caracterizados qualitativamente por meio de previsão de função celular, previsão de localização subcelular e análises de drogabilidade. Foram analisadas 6.327 proteínas e 35 proteínas foram selecionadas como alvos de drogas putativos para a família *Enterobacteriaceae*.

O trabalho de Garcia and Belloze [2018] adotou uma abordagem baseada em ortologia e homologia fazendo uso de proteínas essenciais de organismos modelo e proteínas já conhecidas como alvos de fármacos, e integração destes dados. Os primeiros resultados apontaram uma lista de 91 proteínas candidatas a alvos para fármacos para o *Schistosoma mansoni*, o organismo causador da esquistossomose.

O trabalho de trabalho de Wen et al. [2022] apresenta a ferramenta computacional Geptop, uma forma de identificar genes essenciais de baixo custo e eficiente em termos de tempo. Com base em conjuntos de essencialidade experimental depositados nos bancos de dados DEG e OGEE como referência, a ferramenta Geptop seleciona genes essenciais do conjunto de genes codificadores de proteínas em um genoma procariótico, que utiliza a estratégia *reciprocally best hit* para busca de homologia e distância evolutiva para atribuição de peso. A versão mais recente do Geptop é a 2.0 e prevê a essencialidade do gene com a AUC média de 0,84 em procariotos além de ser mais estável.

III.3 Técnicas Híbridas

Azhagesan et al. [2018] trabalharam na construção do banco de dados NetGenes, um banco de essencialidade de genes voltado para bactérias. No estudo, os autores utilizaram medidas de

centralidade de redes de interação proteína-proteína, extraídas do banco STRING, para treinar o algoritmo RandomForest e predizer a essencialidade de genes de mais 87000 genes. O banco NetGenes está disponível na WEB e atualmente sua nova versão possui mais de 2700 interatomas de bactérias [Senthamizhan et al., 2021].

Outro trabalho de Garcia et al. [2020] realizou um estudo dedicado a essencialidade de proteínas do *S. mansoni*, mas dessa vez com a aplicação de aprendizado de máquina. Os algoritmos propostos foram de classificação como Random Forest, J48, SMO e Logistic Regression. O modelo classificatório obteve o melhor desempenho com o algoritmo Random Forest comparado aos demais utilizados.

Zhang et al. [2020] propuseram uma plataforma online de descoberta de proteínas essenciais baseada em rede: a NetEPD. A plataforma integra dados de redes de interação proteína-proteína (PPI), expressão gênica, localização subcelular e um conjunto de redes de proteínas essenciais. Além disso, computa 22 tipos de medidas de centralidade, avalia predição de proteínas essenciais e visualiza as redes de PPI através de uma plataforma interativa.

O estudo de Xiaoqin et al. [2021] explora um novo padrão denominado EPSFLA para predizer proteínas essenciais por meio da aplicação do algoritmo de otimização *Shuffled frog-leping* (SFLA). O algoritmo baseia-se na integração de informações de localização subcelular, complexos proteicos, anotação de *Gene Ontology*, ortologia e rede PPI.

He et al. [2021] propôs um novo modelo de predição de possíveis proteínas essenciais chamado KFPM (*Key Features of Proteins in a Novel Protein-Domain Network*). Este modelo combina conhecidas interações proteína-proteína com associações de proteínas e domínios utilizando uma melhoria do algoritmo PageRank. Na mesma direção, o trabalho de Li et al. [2021] propôs um novo modelo de predição denominado TGSO que combina três tipos de redes para formar uma rede de interação proteína-proteína: a primeira adota um método baseado em medida de densidade de nó, a segunda combina informação de expressão gênica com rede de conectividade e a última baseada em dados de localização subcelular. Além disso, uma pontuação é calculada iterativamente para estimar a importância de diversas proteínas.

O trabalho de Zhu et al. [2022] propôs um novo modelo preditivo chamado LNSPF (*linear neighborhood similarity-based protein multifeatures fusion*) que foi projetado combinando características topológicas de redes PPI com uma série de características biológicas de proteínas para detectar essencialidade. No LNSPF foi adotado um método que se baseia em entropia para fusão de características e um método de similaridade linear de vizinhança para otimização.

O trabalho de Shen et al. [2022] apresenta o algoritmo LDS para prever proteínas essenciais. O LDS combina o algoritmo LFD (*Local Fractal Dimension*) com a ideia de conjunto fuzzy gerando o algoritmo LFFD (*Local Fuzzy Fractal Dimension*) combinadas com uma pontuação do compar-

timento subcelular calculada através da fórmula de Bayes. Para isso são utilizadas informações de localização subcelular de proteínas essenciais e não essenciais de *Saccharomyces cerevisiae*.

O trabalho de Yue et al. [2022] apresenta um método baseado em *Deep Learning* para identificação de essencialidade em proteínas do organismo *Saccharomyces cerevisiae*. Foram utilizados dados de PPI com o método node2vec, localização subcelular mapeadas em um longo vetor unidimensional e perfis de expressão gênica com uso de redes convolucionais. Os autores também realizaram comparações e perceberam que o node2vec apresentou resultados melhores em comparação às medidas de centralidade.

Schapke et al. [2022] propôs a EPGAT, uma abordagem para predição de essencialidade baseada em *Graph Attention Networks* (GATs), que são *Graph Neural Networks* (GNNs), que operam em dados estruturados em grafos. O modelo aprende diretamente os padrões de essencialidade de genes de redes PPI, integrando evidências adicionais de dados multiômicos codificados como atributos de nós. Comparando o EPGAT para quatro organismos, incluindo humanos, o algoritmo previu a essencialidade do gene com pontuação ROC AUC alcançada entre de 0,78 a 0,97.

O trabalho de Beder et al. [2021] empregou aprendizado de máquina em seis modelos eucariotos, usando 41635 características derivadas de sequência, informações sobre a função do gene e rede de PPI para predição de essencialidade em genes. Os algoritmos propostos foram o *Random Forest* e o *XGBoost* do Pacote R utilizando SMOTE (*Synthetic Minority Oversampling Technique*) para balanceamento de classes. Dentro de uma validação cruzada *leave-one-organism-out*, os classificadores mostraram alta generalização com uma precisão média próxima a 80% nas espécies trabalhadas.

Manzo et al. [2023] apresentaram um trabalho de identificação de essencialidade em genes humanos por meio de um abordagem tecido-específica, aplicada ao rim, usando aprendizado de máquina. Para isso utilizaram atributos biológicos específicos de tecido relacionados a níveis de expressão gênica do rim em tecido normal e tumoral, dados metabólicos e de genes de diversos bancos, além de atributos de PPI como medidas de centralidade para entradas dos algoritmos. Foram utilizados algoritmos de aprendizado de máquina clássicos como SVM e *Random Forest* e algoritmos de *Deep Learning* como node2vec com redes neurais, totalizando oito algoritmos. Observou-se resultados bons nas métricas de avaliação para o algoritmo node2vec com redes neurais.

Existem muitos outros trabalhos na literatura relacionados a essencialidade de genes e proteínas. Em geral, estes trabalhos possuem foco em patógenos, mas é notável que a detecção de essencialidade é também realizada em proteínas de órgãos humanos. Nesta seção buscou-se apresentar aqueles trabalhos mais recentes, dos últimos cinco anos, e relacionados ao escopo do tema desta pesquisa, ou seja, identificação de proteínas essenciais utilizando métodos computacionais.

Este trabalho apresenta dois métodos para identificação de proteínas essenciais, um baseado

em ortologia e outro baseado em aprendizado de máquina que podem ser utilizados para auxiliar no processo de identificação de proteínas essenciais. Os experimentos do trabalho utilizam mais organismos modelo, mais métodos de *Ensemble* e propõe dois *workflows* para que os experimentos possam ser reprodutíveis. Além disso é entregue a integração de dados de essencialidade para ser utilizada nos dois *workflows*.

Capítulo IV Metodologia

A metodologia deste trabalho se baseou nos trabalhos de Peng et al. [2017] e Garcia et al. [2020], mencionados no capítulo III, que abordaram métodos baseados em homologia, PPI, características baseadas em sequência e aprendizado de máquina. A pesquisa deste trabalho propõe uma análise experimental de algumas das abordagens mencionadas no trabalho de Peng et al. [2017] mas com foco em proteínas assim como o trabalho de Garcia et al. [2020], porém com foco em outros tipos de algoritmos preditivos e entradas.

Para este trabalho, a metodologia proposta apresenta dois *workflows*, um para cada método de identificação de possíveis proteínas essenciais: método baseado em homologia e o método baseado em aprendizado de máquina. Todo o código deste trabalho está disponível no repositório público do Github[1]. As próximas seções descrevem detalhadamente todas as etapas executadas para ambos os *workflows*.

IV.1 Método Baseado em Homologia

A figura IV.1 apresenta o *workflow* das atividades necessárias para o método baseado em homologia. Como informado na seção II.3, a presente pesquisa enfoca o conceito de ortologia. Para este método existem três etapas. A primeira é relativa à extração e preparação dos dados e construção da base integrada de essencialidade. A segunda é relacionada à execução da ferramenta Orthofinder para geração de grupos ortólogos entre as proteínas de um organismo alvo e proteínas de organismos modelo. Por fim, a terceira etapa é relacionada à comparação das proteínas do organismo alvo com as proteínas humanas para verificação de ortologia.

[1] https://github.com/JessicaIta/essentiality-protein-research

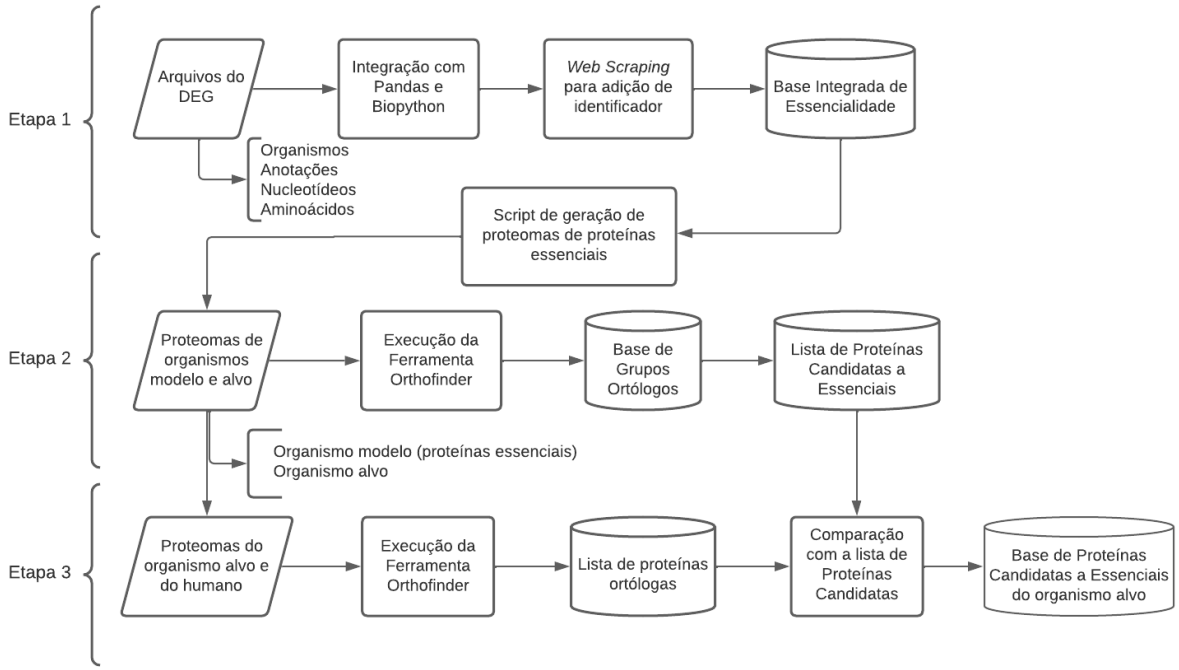

Figura IV.1: Workflow do Método Baseado em Homologia

IV.1.1 Etapa 1 - Base Integrada de Genes e Proteínas Essenciais

A Etapa 1 do *workflow* diz respeito à extração e preparação dos dados provenientes do banco de dados DEG e posterior integração destes dados para construção da base integragada de essencialidade. Para essa etapa foram utilizadas as bibliotecas Pandas e Biopython. A primeira atividade é a integração de dados dos arquivos disponibilizados pelo banco de dados: organismos, anotações, nucleotídeos e aminoácidos, todos referentes a eucariotos. O primeiro arquivo se refere aos dados de organismos e seus respectivos genes essenciais conhecidos e estudos de origem. O segundo arquivo corresponde aos dados de genes e identificadores exclusivos do DEG (prefixo DEG seguido de uma sequência de números). O terceiro e o quarto arquivos correspondem aos dados de sequências de nucleotídeos e aminoácidos, respectivamente. Os arquivos de nucleotídeos (nt) e aminoácidos (aa) foram transformados previamente em estruturas de dicionário (chave e valor), com o auxílio da biblioteca Biopython, pois encontravam-se nos formatos .nt e .aa, semelhantes ao formato fasta. Estes dicionários foram integrados aos arquivos de organismos e anotações por meio de *scripts* com o auxílio da biblioteca Pandas para realizar a primeira parte da integração dos dados.

A integração realizada com dados do DEG continha mais de 30 mapeamentos de dados de organismos eucariotos. Porém, quatro organismos modelo foram selecionados para gerar a base de essencialidade utilizada nas etapas seguintes. Como os dados possuíam associações de 1:1 (um para um), não houve problemas com duplicidade. A junção dos dados foi relacionada com os códigos disponibilizados pelo DEG do organismo, genes e proteínas. Este mesmo processo também pode

ser realizado para organismos procariotos disponíveis no DEG.

A segunda atividade diz respeito à segunda parte da integração de dados: a adição dos identificadores *locus* e *uniprot* dos genes e proteínas essenciais. Estes identificadores são muito importantes para integrar com outras bases de dados, especificamente para proteomas extraídos dos bancos de dados *Ensemble* e *Uniprot*. Para os organismos modelo é utilizado o identificador *locus* e para o humano, o identificador *uniprot*. Estes dados também se encontram no DEG, porém somente na descrição das páginas Web, sem disponibilidade de arquivos para *download*. Assim, foi realizada uma coleta automatizada de dados, por meio de *web scraping*, gerando a base integrada de essencialidade.

IV.1.2 Etapas 2 e 3 - Ortologia

A etapa 2 considera a execução da ferramenta Orthofinder para geração de grupos ortólogos de proteínas. Baseando-se no conceito de ortologia e nas características de organismos modelo, são realizadas execuções para identificação de ortologia entre um organismo modelo com proteínas essenciais já conhecidas e o organismo alvo. Assim, é possível sugerir que se ambas as proteínas são ortólogas e a do organismo modelo é essencial, elas podem ter a mesma função e consequentemente a do organismo alvo ser essencial também.

A partir da base integrada de essencialidade são gerados arquivos de proteínas essenciais para os organismos modelo que servirão como entradas para as execuções da Orthofinder, para o organismo alvo, o proteoma é completo. A execução da Orthofinder gera diversos arquivos com detalhamentos dos resultados, no entanto, para esta pesquisa é utilizado somente o arquivo que contém a base de grupos ortólogos. Quatro execuções são realizadas da Orthofinder com o organismo alvo *Schistosoma mansoni* e os organismos modelo. As três primeiras são execuções do organismo alvo com cada organismo modelo e a quarta se refere ao organismo alvo com todos os organismos modelo. No total são geradas quatro listas de proteínas.

Por fim, a etapa 3 consiste na comparação final para detectar se as proteínas candidatas a essenciais do organismo alvo são também ortólogas ao ser humano, o que inviabilizaria a indicação delas como candidatas. Isso posto, pois, caso as proteínas sejam ortólogas ao patógeno e ao humano, elas tendem a ter a mesma função nos dois organismos e consequentemente a proteína humana poderia ser alvo do fármaco assim como a proteína do patógeno, o que causaria problemas ao humano. Dada esta premissa, é realizada mais uma execução da ferramenta de ortologia entre o organismo alvo e o ser humano. O resultado da execução é uma lista de proteínas ortólogas do organismo alvo com o ser humano. Na continuidade, quatro comparações são feitas entre os resultados dos organismos modelo e o do ser humano. A interseção entre as listas de ortologia com o ser humano não é selecionada.

IV.2 Método Baseado em Aprendizado de Máquina

O método baseado em aprendizado de máquina utiliza características baseadas em contexto, neste caso *features* relacionadas às redes de PPI, e características baseadas em sequência para predição de essencialidade de proteínas. A figura IV.2 apresenta o *workflow* dos passos para o método baseado em aprendizado de máquina. Os passos são descritos nas próximas subseções.

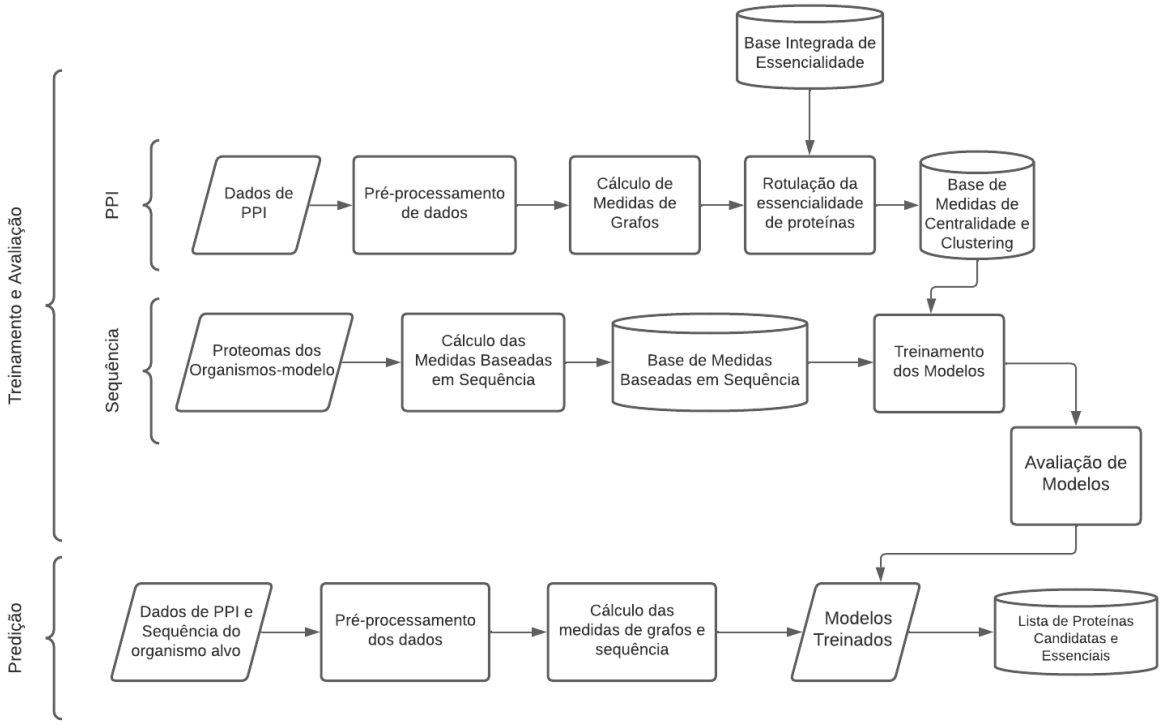

Figura IV.2: Workflow do Método Baseado em Aprendizado de Máquina

Como apresentado na figura IV.2 todo o processo é dividido em treinamento, avaliação e predição. As fases de treinamento e avaliação necessitam ser separadas em dois momentos para tratar separadamente os tipos de entrada: PPI e sequência. Após todo o treinamento e avaliação do modelo é que efetivamente é realizada uma predição de proteínas candidatas com o organismo alvo.

IV.2.1 Características dos Dados de Entrada

Inicialmente, os dados de PPI são coletados do banco de dados STRING. Estes dados são disponibilizados em formato texto e possuem diversos *scores* que apresentam graus de confiança na ligação de uma proteína A com outra proteína B. Estes valores de *score* representam valores de confiança para a origem daquela ligação: experimental, bases de dados curadas, *text mining*, entre outras. Para esta pesquisa foram consideradas as ligações que tinham valor de *score* maior do que zero para as origens experimental ou base de dados curadas (a escala variava de 0 a 1000), ou seja,

existia algum nível de confiança para as origens mencionadas. A primeira atividade do método corresponde ao pré-processamento dos dados com a enumeração de cada proteína. A segunda atividade refere-se ao cálculo de medidas de centralidade e *clustering* de todas as proteínas que formam os nós do grafo de PPI.

Para este trabalho foi utilizada a NetworkX, uma biblioteca Python para análise e exploração de grafos e seus algoritmos [Hagberg et al., 2008]. A NetworkX também interage com outros pacotes Python como NumPy, SciPy, and Seaborn, o que facilita principalmente a construção de gráficos [Hagberg et al., 2008]. Além disso, é uma biblioteca de software livre sob licença de código aberto BSD (*Berkeley Software Distribution*).

Com o auxílio da biblioteca, são calculadas cinco medidas de centralidade: *Degree Centrality* (centralidade de grau), *Closeness Centrality* (centralidade de proximidade), *Betweeness Centrality* (centralidade de intermediação), *Eingenvector Centrality* (centralidade de autovetor). Além das medidas de centralidade, é calculado também o coeficiente de *clustering* de cada nó. Embora existam outras medidas de centralidade, foram selecionadas estas cinco medidas pois foram amplamente mencionadas na literatura [Zhang et al., 2020; Aromolaran et al., 2021]. Após o cálculo das medidas de centralidade e *clustering*, a terceira atividade faz a rotulação da essencialidade de proteínas. Para tanto, a base integrada de essencialidade gerada no método baseado em homologia é utilizada. A comparação é realizada por meio de um *script* Python.

A quarta atividade é relacionada a agregação das características baseadas em sequência. Para estes dados a biblioteca Biopython foi utilizada para calcular percentuais de cada aminoácido e a aromaticidade. O tamanho da sequência e a fração da estrutura secundária foram implementadas a parte, especificamente a fração da estrutura secundária foi construída baseada na tabela do trabalho de Shingate and Sowdhamini [2012] que apresenta a tendência de cada aminoácido estar presente em determinada estrutura.

IV.2.2 Modelos de Aprendizado de Máquina

A quinta atividade tem por objetivo a aplicação de modelos preditivos de classificação para detectar essencialidade em proteínas. Testes preliminares foram realizados com o conjunto de dados relativos a PPI para identificar o algoritmo que apresentasse o melhor resultado. Foram realizados diversos experimentos com balanceamentos *Oversampling* e *Undersampling* e com os dados originais. Os algoritmos selecionados foram: KNN, SVM, Regressão Logística, Árvore de Decisão e Random Forest. Os resultados dos testes apontaram o *Random Forest* com melhores valores de *recall* em conjunto com técnicas de balanceamento [da Silva Costa et al., 2022].

A partir das evidências anteriores, algoritmos baseados em árvore (múltiplas árvores) foram o foco nos novos experimentos. Adicionalmente, houve a agregação das características baseadas em

sequência nos modelos. Para isso foi realizada uma separação estratificada em grupos de treino e teste, na porcentagem 80% para treino e 20% para teste. Foram realizados experimentos com balanceamentos *Oversampling*, *Undersampling* e uma técnica híbrida chamada SMOTEEN, que usa *Oversampling* na classe minoritária e *Undersampling* na classe majoritária. Os algoritmos selecionados foram:

- **Random Forest** da biblioteca *Scikit-learn*;
- **Gradient Boosting** da biblioteca *Tensorflow Decision Forests*;
- **XGBoost** da biblioteca *XGBoost*.

Para identificar os melhores modelos, foi aplicada a técnica de *Gridsearch* para a busca de hiperparâmetros com os algoritmos *XGBoost* e *Random Forest* em uma parte do conjunto de dados de treinamento. Para os modelos de *Gradient Boosting* não foi realizado *Gridsearch*, pois a biblioteca é mais recente, e assim foram utilizados os parâmetros padrão da própria documentação da biblioteca. A métrica utilizada para a busca foi a AUC (*Area Under Curve*) com validação cruzada, dessa forma o grupo de parâmetros com maior valor de AUC foi selecionado para as três técnicas de balanceamento. No total, nove modelos foram selecionados e treinados e, posteriormente, avaliados com as métricas precisão, *recall* e *F1-Score*.

Para complementar a validação, foi realizada uma avaliação com outro organismo modelo que não estava no grupo de treino: o *Mus musculus*. Os dados desse organismo de validação passaram pelo mesmo pré-processamento dos dados dos organismos da base de treino e teste, exceto pela medida de *betweeness centrality* que necessitou de um parâmetro k de estimativa devido a complexidade quadrática do algoritmo que calcula a medida. Além disso os dados de essencialidade do organismo *Mus musculus* foram extraídos do DEG e adicionados à base de essencialidade para que pudesse ser realizada a rotulação dos dados.

O valor de *recall* foi a métrica selecionada para definir os modelos mais adequados para predizer as proteínas essenciais do organismo alvo, pois os falsos positivos são mais aceitáveis para este problema de predição das proteínas. Foram selecionados três cenários, um cenário para cada algoritmo para predição das proteínas essenciais do *Schistosoma mansoni*. Para isso foram extraídos os dados de PPI e o proteoma do organismo alvo para que fossem calculadas suas medidas de centralidade e *clustering* e as baseadas em sequência, respectivamente. As medidas calculadas se tornam entradas para o modelo selecionado que classifica as proteínas do organismo alvo.

Capítulo V Resultados

Este capítulo apresenta os resultados encontrados nas duas abordagens propostas para identificação de candidatas a proteínas essenciais: por homologia e por aprendizado de máquina. Ambos os experimentos trabalharam com três organismos modelo, *Saccharomyces cerevisiae*, *Caenorhabditis elegans* e *Drosophila melanogaster*, além do organismo alvo *Schistosoma mansoni*.

V.1 Método Baseado em Homologia

V.1.1 Base de Essencialidade

A primeira base de dados gerada foi a base integrada de essencialidade, a partir dos dados disponíveis do DEG. A atividade de *web scraping* durou cerca de 68 minutos para 4973 proteínas em um computador Core i5 8ª geração e 8 gigabytes de memória. Esse processo possibilitou a construção da base de genes e proteínas essenciais para três organismos modelo e o ser humano: *Caenorhabditis elegans*, *Drosophila melanogaster*,*Saccharomyces cerevisiae* e *Homo sapiens*. A base foi gerada em dois formatos de dados, CSV e parquet. O formato parquet especificamente reduz de forma significativa o tamanho do arquivo [Parquet, 2022]. A Tabela V.1 apresenta os quatro organismos selecionados com seus respectivos tamanhos de proteomas (quantidade total de proteínas essenciais e não-essenciais), quantidade de proteínas essenciais e percentual de proteínas essenciais.

Tabela V.1: Quantitativo de proteínas dos organismos modelo e do ser humano.

Organismo	**Proteoma**	**Proteínas essenciais**	**Percentual proteínas essenciais**
Caenorhabditis elegans	31768	294	0,92%
Drosophila melanogaster	30719	339	1,1%
Saccharomyces cerevisiae	6600	1110	16,81%
Homo sapiens	81856	3230	3,94%

Especialmente para o *Homo sapiens* foi necessário extrair outro identificador pois o *locus*, do banco de dados DEG, não possuía correspondência com o proteoma do banco *Ensemble*. Diante desse cenário foi necessário encontrar outra forma de identificar correspondência com o proteoma humano. Observou-se que o DEG possuía o identificador do *Uniprot* o que facilitaria a integração dos dados, assim optou-se por utilizar o proteoma extraído do banco de proteínas *Uniprot* e extrair

o identifcador *Uniprot* unicamente para o ser humano. Esse processo também foi realizado via *Web Scraping* e durou cerca de 43 minutos no computador mencionado anteriormente. Quatro genes de *Saccharomyces cerevisiae* estavam com dados faltantes no identificador *locus* extraído do DEG e para isso foram necessárias consultas no banco SGD (*Saccharomyces Genome Database*) para preenchimento desses dados, o que mantém a integridade de todas as proteínas do organismo. A tabela V.2 apresenta os campos e as descrições que compõem a base integrada de essencialidade.

Tabela V.2: Descrição dos campos da base integrada de essencialidade

Campo	**Descrição**
Organism	Nome científico do organismo
Reference	Estudo original
Name	Nome convencional
Essential_Genes	Quantidade de genes essenciais
Method	Experimento biológico
Code_Organism	Código do DEG do organismo
Date	Data Inserção no DEG
Code_Gene_DEG	Código do DEG do gene
Gene	Nome do gene
Function	Função do gene
RefSeq	Identificação no Genbank
Seq_Gene	Sequência de nucleotídeos
Seq_Prot	Sequência de aminoácidos
Locus	Identificador Locus
Uniprot	Identificador Uniprot (para humanos)

V.1.2 Verificação de Ortologia

Todos os proteomas dos organismos modelo foram reduzidos para conter somente as proteínas essenciais, o que facilitou a análise de ortologia com o organismo alvo para gerar proteínas candidatas a essenciais. Dessa forma as execuções da ferramenta Orthofinder foram rápidas, levando cerca de três minutos para as execuções realizadas entre dois organismos. Já a execução realizada com todos os organismos modelos para identificar grupos ortólogos em comum durou cerca de quatro minutos.

Nos resultados da etapa 2, a verificação da ortologia entre os pares de organismos (um modelo e o alvo) e entre todos os organismos modelo e o alvo geraram arquivos individuais em formato texto com as proteínas ortólogas. A tabela V.3 apresenta a quantidade de proteínas ortólogas encontradas nos resultados das execuções. Para o grupo de 'todos os organismos modelo e o organismo alvo' foram selecionadas proteínas do *Schistosoma mansoni* que tivessem ortologia com todos os organismos modelo.

Tabela V.3: Quantidade de proteínas ortólogas

Organismos	**Qtd. Proteínas Ortólogas**
Saccharomyces cerevisiae x alvo	1100
Caenorhabditis elegans x alvo	44
Drosophila melanogaster x alvo	415
Todos organismos modelos x alvo	18

A etapa 3 foi a verificação de ortologia com o ser humano para retirar essas proteínas dos grupos gerados na etapa 2. Por isso foi necessário verificar em cada grupo se haviam proteínas ortólogas nos grupos simultâneos com o ser humano. Especificamente com o ser humano, o *Schistosoma mansoni* apresentou 2553 proteínas ortólogas, o que indicava que essas proteínas não poderiam ser candidatas a essenciais. A tabela V.4 apresenta o resultado quantitativo das comparações realizadas. Já a tabela V.5 apresenta a lista com a descrição das proteínas encontradas no grupo Todos organismos modelo [Vasconcelos et al., 2018].

Tabela V.4: Quantidade de proteínas candidatas a essenciais

Organismo	**Candidatas**
(*Saccharomyces cerevisiae* x alvo) - ser humano	726
(*Caenorhabditis elegans* x alvo) - ser humano	35
(*Drosophila melanogaster* x alvo) - ser humano	223
(Todos organismos modelo x alvo) - ser humano	11

Tabela V.5: Descrição das 11 proteínas encontradas no grupo de '(todos os organismos modelos x alvo) - ser humano'

Identificador	Proteína
Smp_157750.1	Rna-binding protein musashi-related
Smp_017280.1	Putative ubiquitin; Ubiquitin and ribosomal protein S27-like protein
Smp_046690.1	Putative ubiquitin (Ribosomal protein L40)
Smp_046690.2	Putative ubiquitin (Ribosomal protein L40)
Smp_046690.3	Putative ubiquitin (Ribosomal protein L40)
Smp_089430.1	Ubiquitin (Ribosomal protein L40), putative
Smp_123260.1	Putative ubiquitin 1
Smp_130170.1	NEDD8 — Putative ubiquitin 1
Smp_335990.1	ubiquitin B
Smp_007170.1	Calcium binding protein
Smp_079050.1	DNA primase large subunit, putative

Nota-se que o organismo *Saccharomyces cerevisiae* apresentou mais proteínas ortólogas com o *Schistosoma mansoni* em comparação aos outros organismos. Uma possível causa para esta discrepância é a quantidade de proteínas essenciais identificadas nos organismos modelo. Infelizmente a base extraída do DEG para o organismo *Caenorhabditis elegans* possuía poucas proteínas catalo-

gadas como essenciais em comparação ao proteoma. Dado que o *C. elegans* é um organismo mais complexo do que a *S. cerevisiae* é bem possível que ele possua muito mais proteínas essenciais.

V.2 Método Baseado em Aprendizado de Máquina

O método baseado em aprendizado de máquina foi construído com base em características baseadas em contexto, representado pelo PPI, e características baseadas em sequência. Outro ponto a destacar é que a base de essencialidade gerada no método baseado em homologia também foi utilizada para rotulação dos dados de treinamento. A seguir são explicadas todas as etapas executadas.

V.2.1 Características Baseadas em Contexto

No primeiro momento foram calculadas as medidas de centralidade e *clustering* nos grafos de PPI dos organismos modelo. Essas foram calculadas separadamente para cada organismo com funções da biblioteca NetworkX. Foram três execuções com duração de cerca de 70 minutos cada. Devido às quantidades de proteínas nos dois grupos (essencial e não essencial) não serem balanceadas, optou-se por calcular a distribuição de probabilidade em cada grupo separadamente.

A figura V.1 apresenta cinco gráficos de distribuição de probabilidade das medidas calculadas no experimento. Os gráficos representam, respectivamente, centralidade de grau, de autovetor, de intermediação, de aproximação e o coeficiente de *clustering*. Especificamente nas medidas de *degree centrality*, *betweenness centrality* e *clustering*, principalmente nas proteínas não essenciais, havia muitos valores baixos de probabilidade, próximos de zero. Percebe-se que as distribuições de probabilidade são muito similares entre essenciais e não essenciais, porém, há uma maior diferenciação nas medidas de *closeness centrality* e *clustering*, o que indica que estas métricas podem separar melhor as proteínas essenciais das não essenciais.

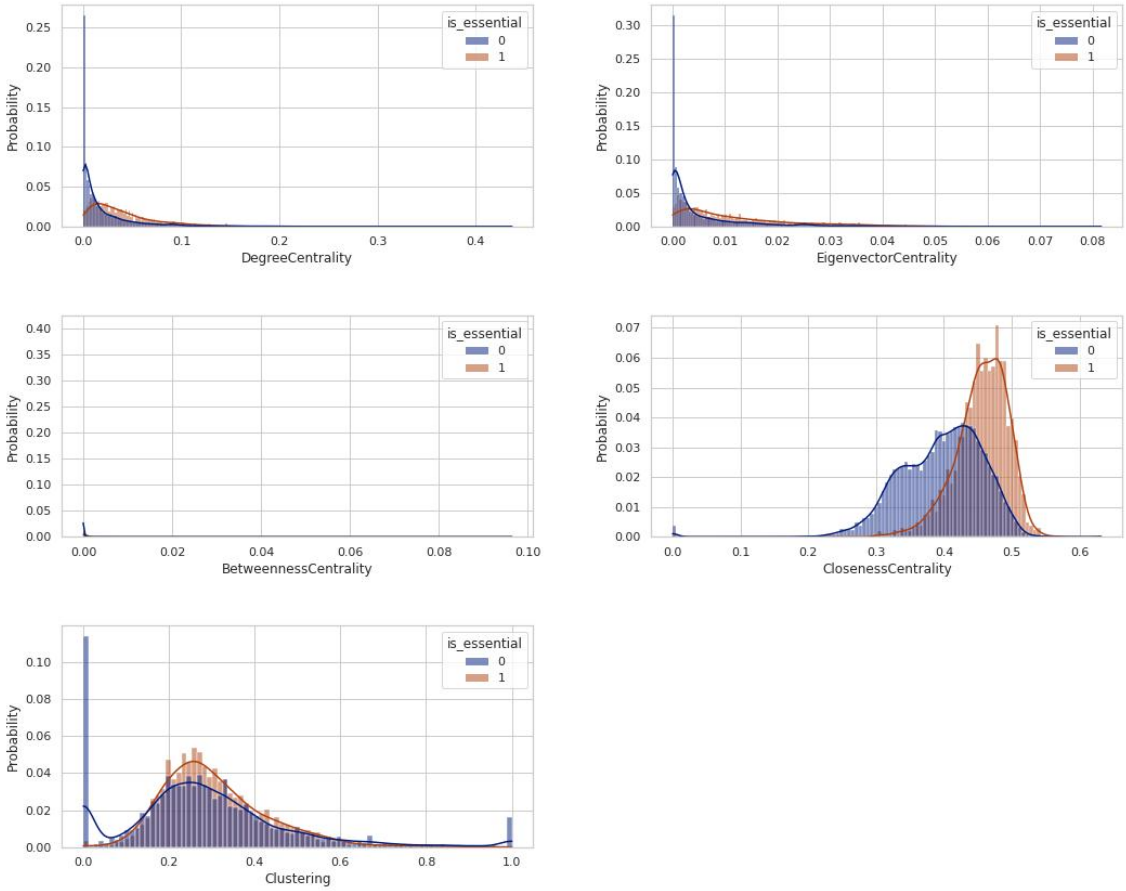

Figura V.1: Distribuição de probabilidade das medidas de centralidade e *clustering*

V.2.2 Características Baseadas em Sequência

As características ou medidas baseadas em sequência foram calculadas com apoio da biblioteca Biopython e por meio da implementação de algumas funções para o tamanho da sequência e a fração da estrutura secundária. No total foram calculadas 25 medidas baseadas em sequência, sendo cinco relacionadas ao tamanho da sequência, aromaticidade e fração da estrutura secundária (*helix, sheet e turn*), e as 20 restantes relacionadas ao percentual da presença dos aminoácidos na sequência da proteína.

A figura V.2 apresenta a distribuição de probabilidade das medidas baseadas em sequência. Percebe-se que as distribuições de probabilidade são muito similares entre essenciais e não essenciais, porém existe certa diferenciação para alguns tipos de aminoácidos (C, E, F, I, K e Y), na propriedade de aromaticidade e nas estruturas alfa-hélice e beta-folha (*Sec_Struct_Helix* e *Sec_Struct_Sheet*).

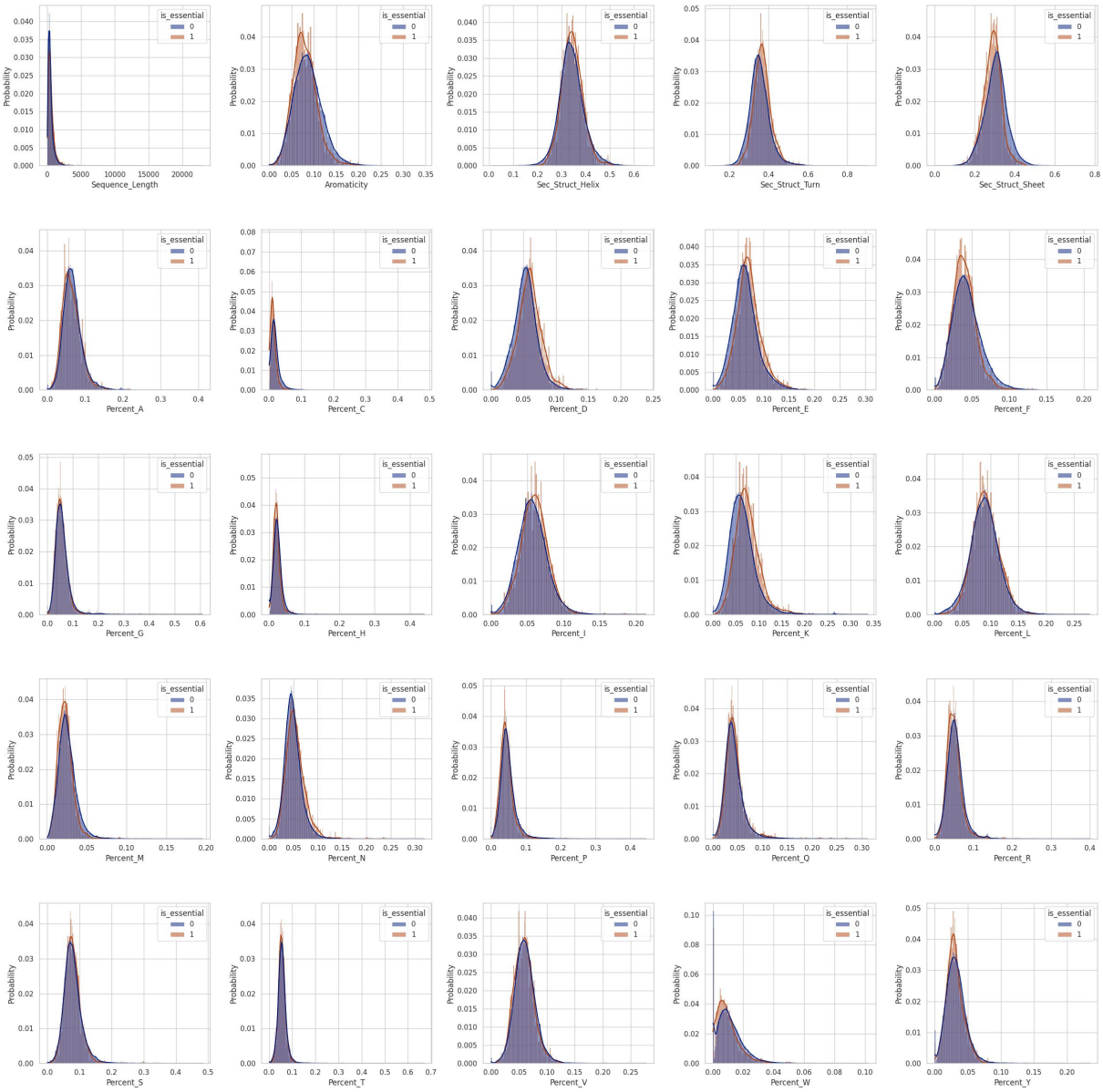

Figura V.2: Distribuição de probabilidade das medidas baseadas em sequência

V.2.3 Experimento Preliminar com Medidas de PPI

Um experimento preliminar foi realizado utilizando medidas de centralidade e *clustering* como entradas para cinco algoritmos de aprendizado de máquina da Silva Costa et al. [2022]. Como forma de rotular os dados de treinamento, este experimento também utilizou a base de essencialidade mencionada anteriormente. O experimento trabalhou com a busca de hiperparâmetros e a aplicação de duas técnicas de balanceamento: *Oversampling* e *Undesrsampling* para realizar testes para identificar os melhores resultados de precisão e principalmente de *recall*, dado que a base de essencialidade é desbalanceada.

Os resultados do experimento são apresentados na tabela V.6 em que são informados os valo-

res de precisão e *recall* do experimento realizado com os cinco algoritmos: KNN (K-vizinhos mais próximos), SVM (Máquinas de vetores suporte), *Random Forest*, CART (Árvore de Decisão) e Regressão Logística. Observou-se que o algoritmo *Random Forest* apresentou melhores resultados de *recall* comparado aos outros algoritmos. Além disso, o uso de técnicas de balanceamento apresentara melhores resultados em comparação aos dados originais desbalanceados.

Tabela V.6: Precisão e *Recall* dos melhores hiperparâmetros para cada tipo de balanceamento e algoritmo

Dados	**Classe**	**KNN**		**SVM**		**R. Forest**		**CART**		**Logística**	
		Pre.	**Rec.**	**Pre.**	**Rec.**	**Pre.**	**Rec.**	**Pre.**	**Rec.**	**Pre.**	**Rec.**
Orig.	Essencial	0.54	0.1	0	0	0.5	0.02	0.46	0.07	0	0
	Não Essen.	0.84	0.98	0.83	1	0.83	1	0.84	0.9	0.83	1
Over.	Essencial	0.26	0.43	0.29	0.69	0.28	0.78	0.27	0.72	0.3	0.65
	Não Essen.	0.86	0.74	0.91	0.65	0.93	0.58	0.91	0.59	0.9	0.68
Under.	Essencial	0.27	0.72	0.3	0.64	0.28	0.76	0.29	0.7	0.3	0.65
	Não Essen.	0.91	0.6	0.9	0.69	0.92	0.59	0.91	0.64	0.9	0.69

Diante o resultado do experimento preliminar, optou-se por prosseguir com o uso de modelos baseados em árvore, especificamente os modelos de *Ensemble*, para os próximos passos da pesquisa. Neste novo cenário o *Random Forest* e *XGBoost* foram escolhidos, o *Random Forest* por ser amplamente mencionado na literatura e por ter se apresentado com os melhores resultados no experimento preliminar e o XGBoost por ser uma técnica bastante utilizada em outras áreas, inclusive no mercado financeiro e em outros temas da área de saúde como o câncer [Li et al., 2023; Guan et al., 2023]. Assim os experimentos seguintes foram focados na nova abordagem de modelos com o acréscimo das características baseadas em sequência, as quais trazem mais informações sobre as proteínas.

V.2.4 Treinamento

A união dos dois tipos de características, contexto e sequência, gerou uma base de treinamento e teste com 30 características que serviram como entradas para o modelo de classificação. A figura V.3 apresenta um gráfico com a correlação entre as entradas e a classe preditora. As correlações entre entradas e a presença de essencialidade apresentou correlações positivas, mas baixas, em geral abaixo de 0.2. Porém observou-se três entradas com correlação positiva mais significativa: a *Degree Centrality*, *Eigenvector Centrality* e a *Closeness Centrality*.

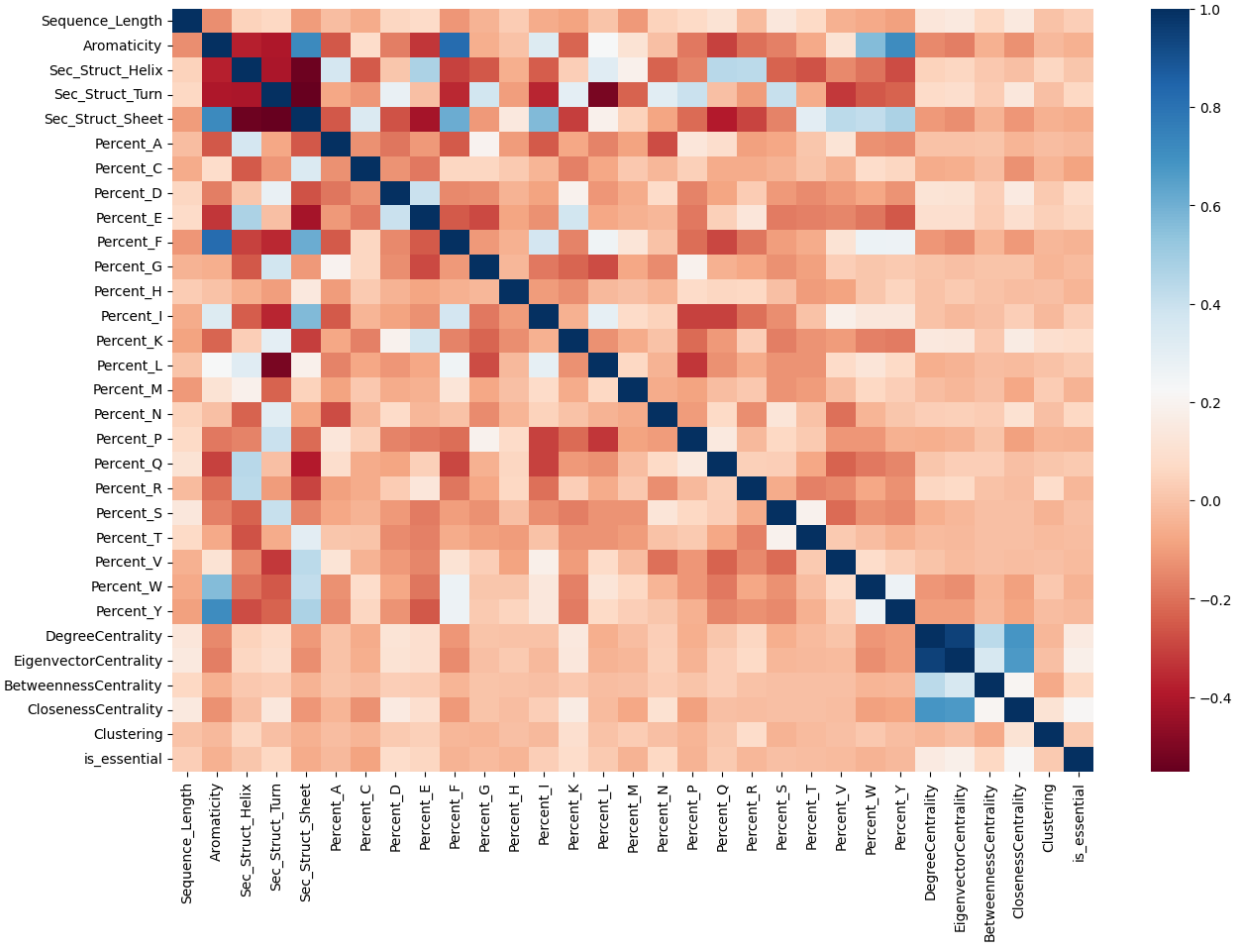

Figura V.3: Correlação entre as características de entrada

Com relação a proporção das classes, na base trabalhada havia a presença de 28843 proteínas não essenciais e 1668 proteínas essenciais, o que mostra um grande desbalanceamento. Diante disso foram testadas algumas técnicas de balancamento: *Oversampling*, *Undersampling* e uma técnica Híbrida, o SMOTEEN. No treinamento dos modelos com XGBoost, *Gradient Boosting* e *Random Forest*, a base de dados foi separada em base de treino e base de teste, na proporção 80% e 20% para treino e teste, respectivamente. Além disso é realizada de forma estratificada entre as classes. As técnicas de balanceamento são aplicadas após a separação de treino e teste, somente no grupo de treino.

Para escolher os melhores hiperparâmetros foi realizada uma busca *Gridsearch* com grupos de parâmetros para os algoritmos XGBoost e Random Forest. Especificamente para o Gridsearch foi utilizada 80% da base de treino para as execuções. Para a busca a métrica de avaliação utilizada foi a AUC. A validação cruzada utilizada foi de 5 *k_folds*. O tempo de execução durou cerca de cinco horas para cada algoritmo. A tabela V.7 apresenta os grupos de hiperparâmetros selecionados para o algoritmo XGBoost. A tabela V.8 apresenta os grupos de hiperparâmetros selecionados para o algoritmo Random Forest. Todos estes grupos de parâmetros foram utilizados para gerar

seis modelos e selecionar o melhor grupo de hiperparâmetros para cada algoritmo. Com relação aos três modelos com *Gradient Boosting*, os hiperparâmetros padrão foram profundidade da árvore de 6 e o número de árvores foi 300.

Tabela V.7: Resultado *Gridsearch* do XGBoost

Dados	**Hiperparâmetro**	**Valor**
Oversampling	Learning rate	0.3
	Max Depth	10
	N estimators	300
Undersampling	Learning rate	0.1
	Max Depth	7
	N estimators	100
SMOTEEN	Learning rate	0.3
	Max Depth	8
	N estimators	300

Tabela V.8: Resultado *Gridsearch* do Random Forest

Dados	**Hiperparâmetro**	**Valor**
Oversampling	Bootstrap	False
	Max Depth	10
	Criterion	gini
	N estimators	300
Undersampling	Bootstrap	False
	Max Depth	10
	Criterion	gini
	N estimators	100
SMOTEEN	Bootstrap	False
	Max Depth	10
	Criterion	gini
	N estimators	300

Após o treinamento dos seis modelos foi realizada a avaliação das métricas de precisão, *recall* e *F1-Score* de todos os modelos com os dados de teste. A tabela V.9 apresenta os valores de precisão, *recall* e *F1-Score* dos modelos de treinamento e detalha melhor os resultados por classe. Observa-se que as técnicas de SMOTEEN e *Undersampling* apresentaram resultados bem similares para algoritmo XGBoost, no entanto o valor de *recall* para técnica de *Undersampling* foi um pouco maior e por isso foi escolhida para as próximas etapas. Com relação aos algoritmos *Random Forest* e *Gradient Boosting*, também houve um comportamento semelhante, mas com diferenças maiores para o *recall*, por isso também a técnica *Undersampling* escolhida para a predição das proteínas do organismo alvo.

Tabela V.9: Precisão, Recall e F1-Score dos Modelos com os dados de Teste

Dados	**Classe**	**XGBoost**			**Random Forest**			**Gradient Boosting**		
		Prec.	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
Under.	**Essencial**	0.14	**0.76**	0.24	0.14	**0.77**	0.23	0.15	**0.76**	0.24
	Não Essenc.	0.98	0.74	0.84	0.98	0.72	0.83	0.98	0.74	0.85
Over.	Essencial	0.25	0.34	0.29	0.15	0.65	0.25	0.21	0.5	0.29
	Não Essenc.	0.96	0.94	0.95	0.97	0.8	0.83	0.97	0.89	0.93
SMOTEEN	Essencial	0.22	0.47	0.30	0.13	0.74	0.22	0.18	0.57	0.27
	Não Essenc.	0.97	0.9	0.93	0.98	0.72	0.83	0.97	0.84	0.9

V.2.5 Experimento de Validação: *Mus musculus*

Para validar os experimentos com os algoritmos XGBoost e Random Forest foi conduzido um experimento de validação com outro organismo modelo que não estava no grupo de treinamento, o *Mus musculus*. Tal experimento foi desenhado de modo a validar os resultados alcançados no teste dos modelos treinados. Para isso foi necessário executar o mesmo processamento de dados realizado nos dados de treinamento: calcular as medidas de PPI e as de sequência.

Com relação às medidas de PPI, o cálculo foi realizado por meio da execução em uma instância de Notebooks na nuvem, especificamente na *Google Cloud Platform*, pois seria uma execução demorada devido a quantidade de proteínas, em torno de 19 mil proteínas e quatro milhões de ligações. A complexidade da medida *betweeness centrality* é quadrática e dada a quantidade de nós foi necessário usar um parâmetro k para estimar a quantidade de nós usados para calcular a medida e não todos, como seria por padrão. Foi adotado o valor que representava em torno de 10% da quantidade de nós, 190, e após a execução observou-se que valores resultantes estavam na mesma escala dos dados de treinamento. O tempo total do cálculo das medidas foi de sete horas em nuvem. Com as medidas de sequência não houve alteração na forma de cálculo.

Foram executados todos os modelos do treinamento para validar a melhor abordagem para ser utilizada na predição das proteínas essenciais do *S. mansoni*. O experimento mostrou que a técnica de *Undersampling* apresentou maior *recall* para todos os modelos propostos. A tabela V.10 apresenta os valores de precisão, *recall* e *F1-Score* do experimento de validação.

Tabela V.10: Precisão, Recall e F1-Score do Experimento com *Mus Musculus*

Dados	**Classe**	**XGBoost**			**Random Forest**			**Gradient Boosting**		
		Prec.	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
Under.	**Essencial**	0.13	**0.52**	0.21	0.16	**0.6**	0.25	0.16	**0.62**	0.25
	Não Essenc.	0.91	0.95	0.93	0.93	0.76	0.84	0.94	0.64	0.76
Over.	Essencial	0.14	0.05	0.07	0.16	0.31	0.21	0.16	0.12	0.14
	Não Essenc.	0.91	0.97	0.94	0.92	0.82	0.87	0.91	0.93	0.92
SMOTEEN	Essencial	0.16	0.10	0.12	0.17	0.45	0.24	0.15	0.16	0.16
	Não Essenc.	0.91	0.95	0.92	0.93	0.76	0.84	0.91	0.9	0.91

V.2.6 Predição de Proteínas Essenciais do *Schistosoma mansoni*

Os modelos selecionados para a previsão das proteínas essenciais do organismo *Schistosoma mansoni* foram treinados com a técnica de *Undersampling*, abrangendo 2668 proteínas no treino. O critério de escolha foi baseada nos resultados encontrados no teste e na validação com os dados do organismo *Mus musculus*. Embora a técnica SMOTEEN tenha apresentado um resultado notável no experimento de validação, de forma geral, levando em consideração dados de teste e validação, a técnica *Undersampling* foi melhor. Com os modelos definidos e treinados, foram observadas as importâncias das entradas (*features*) desses modelos, assim foi possível identificar quais delas foram mais relevantes para detectar essencialidade.

As figuras V.4 e V.5 apresentam a importância das entradas para os modelos XGBoost e *Random Forest*, respectivamente. Nota-se que a entrada baseada em PPI *closeness centrality* teve grande importância no modelo *XGBoost*, seguida da entrada *Degree Centrality* e da entrada baseada em sequência relacionada ao aminoácido C. No modelo *Random Forest*, todas as entradas baseadas em PPI têm importância notável no modelo e novamente a entrada de sequência relacionada ao aminoácido C aparece com importância notável, mas seguida da entrada relacionado ao aminoácido K.

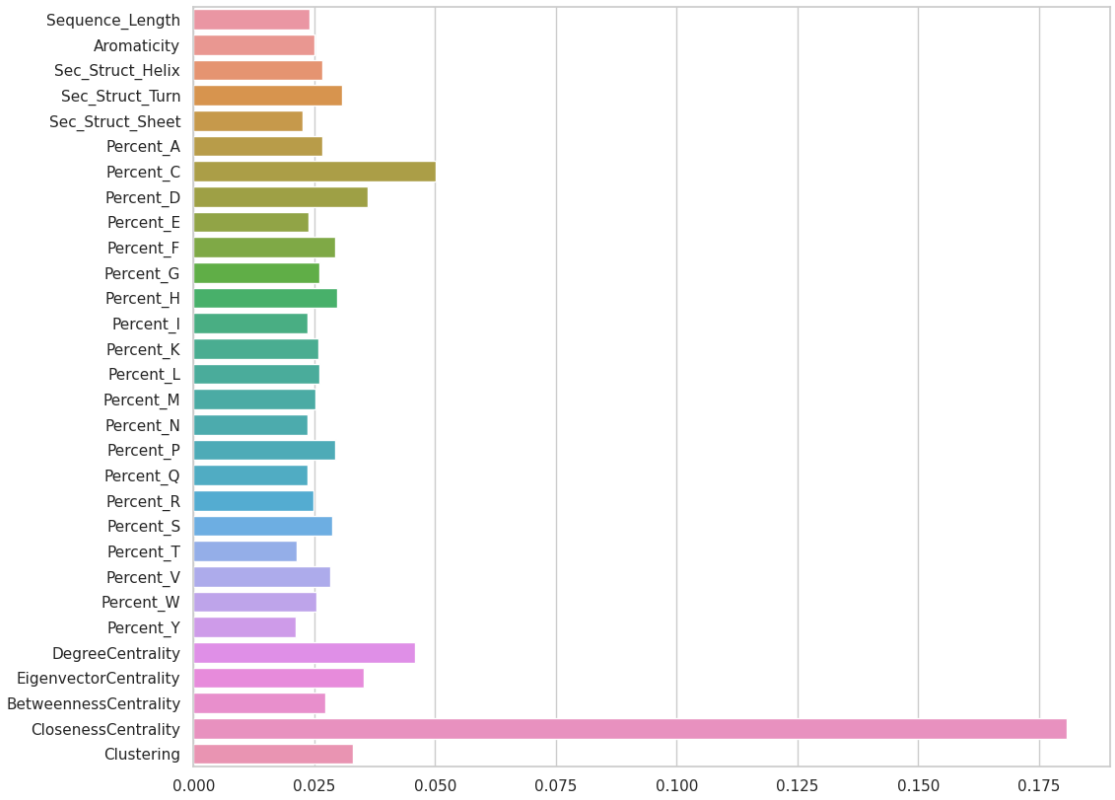

Figura V.4: Importância de cada *feature* para o XGBoost

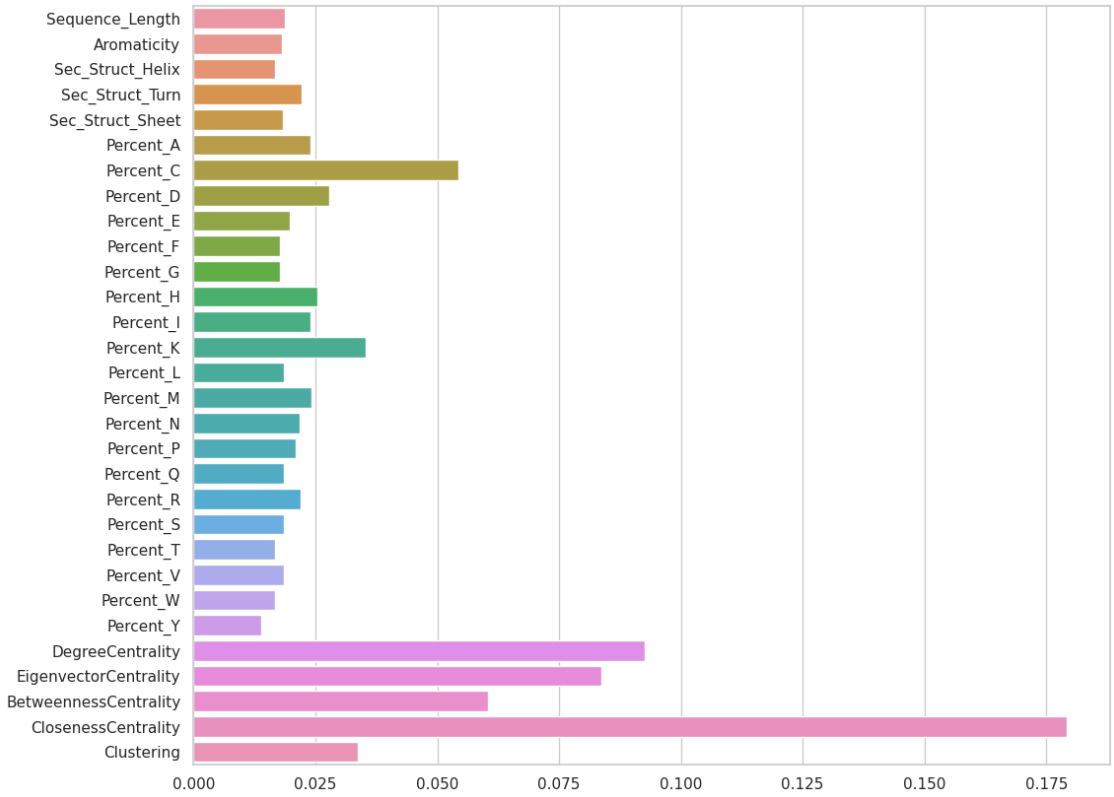

Figura V.5: Importância de cada *feature* para o Random Forest

Já o modelo *Gradient Boosting*, o Tensorflow apresenta a importância das entradas no log de treinamento, três entradas foram destacadas como mais importantes: *Closeness Centrality*, *Degree Centrality* e *Percent_C*. A figura V.6 apresenta uma das árvores formadas pelo modelo *Gradient Boosting* até o quarto nível de profundidade, assim é possível visualizar como ela realiza a divisão. Um ponto de destaque é que a primeira ramificação da árvore começa pela entrada *Closeness Centrality*, apontada anteriormente como importante para o modelo.

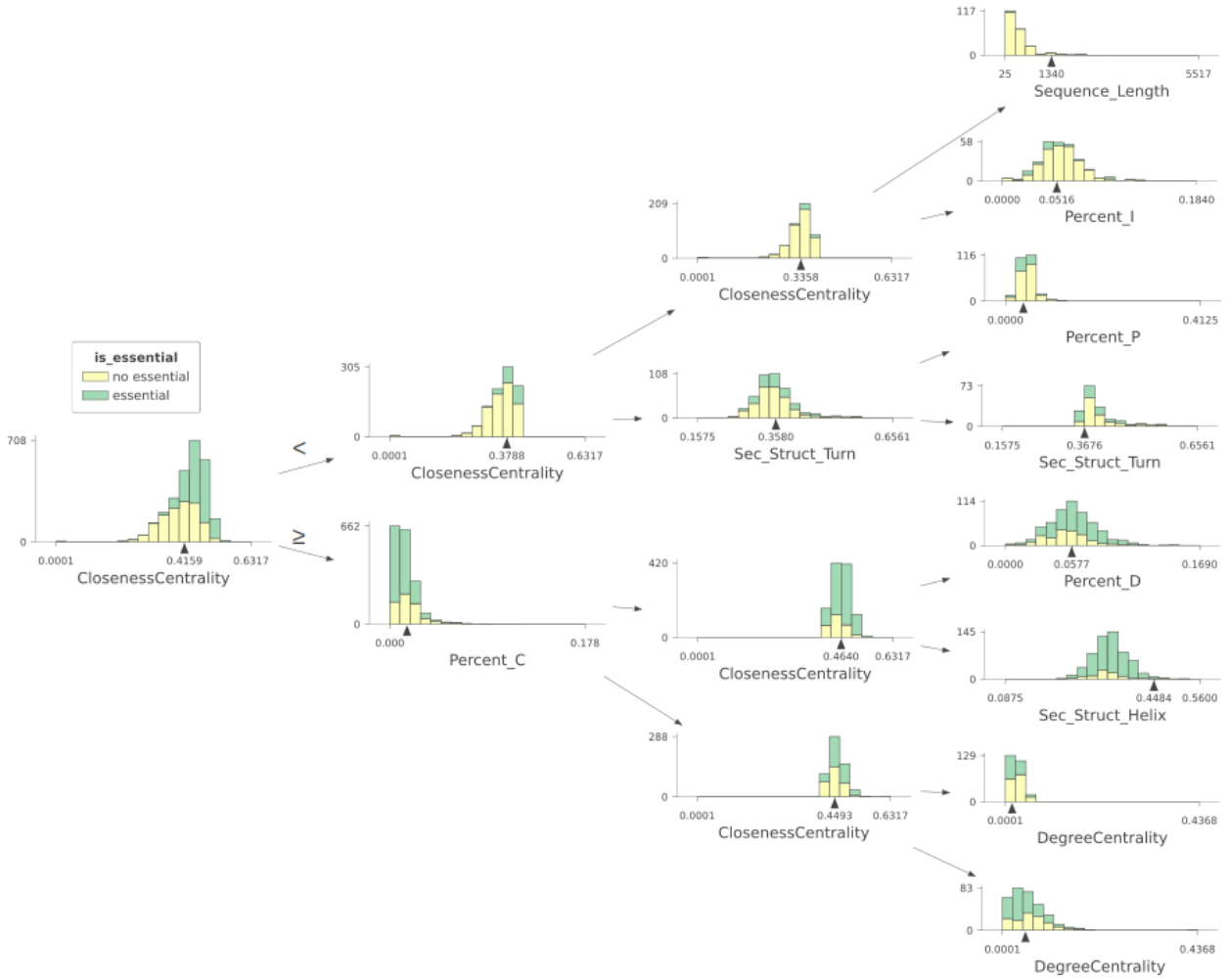

Figura V.6: Trecho de uma das árvores do modelo *Gradient Boosting*

Após todas as validações e análises, foi realizada a predição das proteínas do *Schistosoma mansoni*. O processamento dos cálculos das entradas do organismo durou cerca de três horas, para medidas de PPI e sequência, para um total de 6.347 proteínas. A figura V.7 apresenta a distribuição das entradas do conjunto de dados do *S. mansoni*. Observa-se que a distribuição dos dados do organismo é muito similar à distribuição dos dados de treinamento dos outros organismos utilizados.

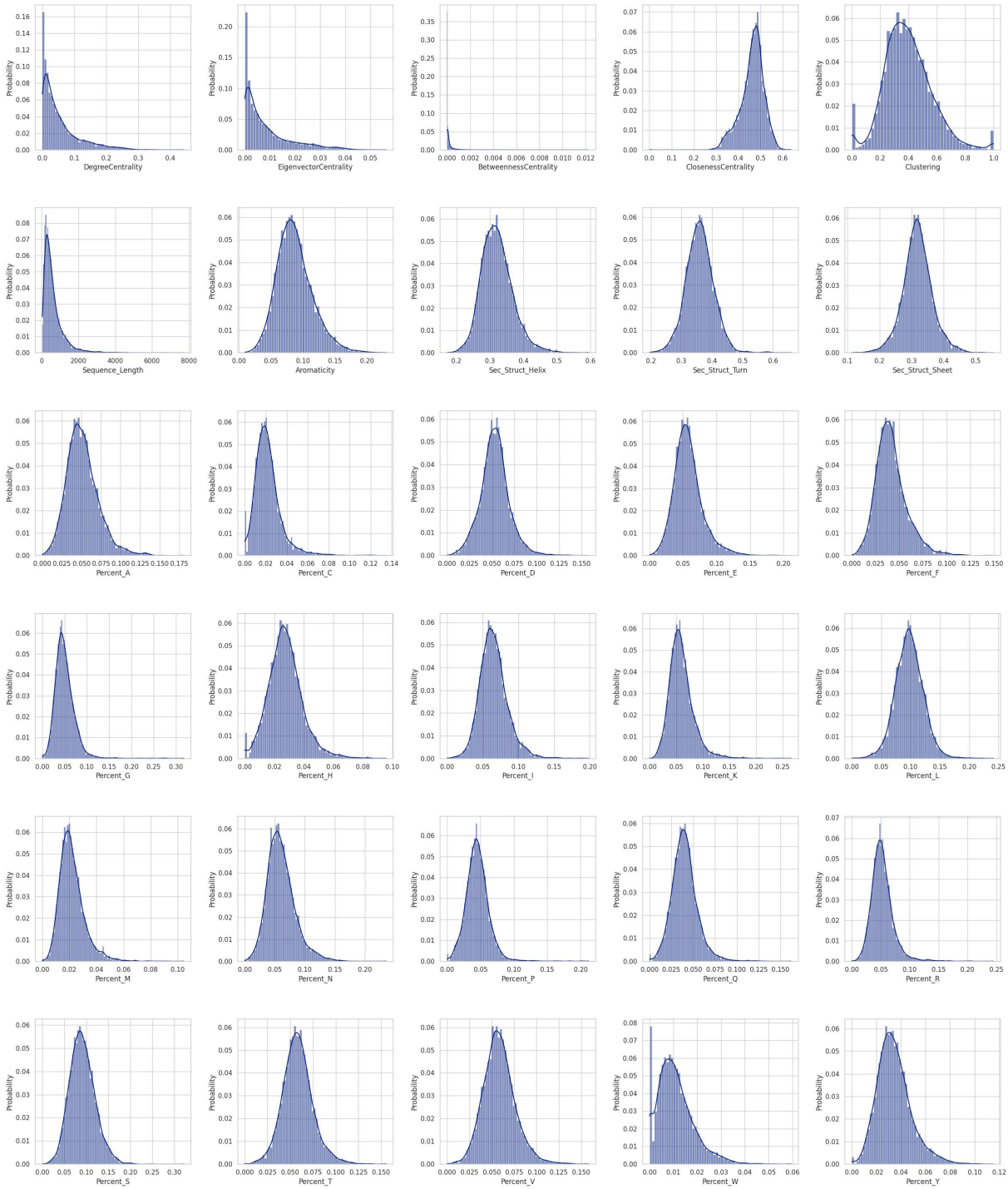

Figura V.7: Distribuição das entradas do *Schistosoma mansoni*

As predições foram muito rápidas e dois modelos classificaram as 6347 proteínas do organismo *mansoni*. Embora o organismo possua um número maior de proteínas (10700), este conjunto é o que está presente na rede de PPI utilizada neste trabalho. Para o modelo XGBoost foram classificadas 4184 proteínas como essenciais e 2163 como não essenciais; o modelo *Random Forest* classificou 4023 proteínas como essenciais e 2324 como não essenciais; enquanto que o modelo *Gradient Boosting*

classificou 4282 proteínas como essenciais e 2065 como não essenciais. Observa-se que o modelo *Gradient Boosting* classificou mais proteínas como essenciais, seguido do XGBoost e do *Random Forest*.

Em comum, os três modelos classificaram 3297 proteínas como essenciais, o que demonstra certa semelhança nos resultados entre os algoritmos. Algo a destacar ainda sobre este conjunto de proteínas em comum classificadas como essenciais é que 471 proteínas estão anotadas *"Uncharacterized protein"*, o que caracteriza como proteínas hipotéticas, que não tem função conhecida. A tabela V.11 apresenta uma amostra com dez proteínas apontadas como possíveis candidatas a essenciais pelo três algoritmos preditivos: XGBoost, *Random Forest* e *Gradient Boosting*.

Tabela V.11: Proteínas Encontradas

Cód. da Proteína	**Anotação**
6183.Smp_000030.1	26s proteasome regulatory particle subunit,putative
6183.Smp_000040.1	Kinesin light chain, putative
6183.Smp_000050.1	Transient receptor potential cation channel,putative
6183.Smp_000075.1	Transcription factor, putative
6183.Smp_000100.1	Filamin
6183.Smp_000710.1	Ubiquitin-specific peptidase 46 (C19 family)
6183.Smp_000990.1	Exocyst complex component 1
6183.Smp_004040.1	Transcriptional adaptor 2 (Ada2)-related
6183.Smp_004470.1	Peroxiredoxin, Prx3

V.3 Análises de Resultados

Esta seção apresenta uma análise dos resultados entre os métodos para verificar as proteínas indicadas por ambos os métodos como essenciais em termos quantitativos. Além disso é realizada uma verificação na literatura para buscar proteínas que foram indicadas como essenciais por algum dos métodos e que estão sendo estudadas como possíveis alvos de fármacos.

V.3.1 Ortologia e Aprendizado de Máquina

Embora os métodos aplicados não sejam comparáveis entre si, optamos por analisar o conjunto de proteínas candidatas a essenciais obtidas por cada método. Dessa forma, observou-se a interseccionalidade dos resultados do método baseado em homologia e do método baseado em aprendizado de máquina. Assim é possível reforçar com mais evidências os resultados, dado que mais de um método apresentou a mesma proteína. Para a comparação foram selecionadas as proteínas comuns aos dois modelos do método baseado em aprendizado de máquina. Após essa seleção, comparou-se com os quatro experimentos (organismos modelo x alvo) do método baseado em homologia, encontrando os seguintes números:

- *Saccharomyces cerevisiae* apresentou 266 proteínas;
- *Drosophila melanogaster* apresentou 150 proteínas;
- *Caenorhabditis elegans* apresentou 11 proteínas;
- e com todos os organismos apresentou 6 proteínas.

V.3.2 Literatura

Além de comparar os resultados entre métodos, foi realizada uma busca na literatura especializada nas áreas biomédica e biológica de estudos sobre proteínas essenciais e alvos de fármacos do organismo *Schistosoma mansoni*. O artigo de Cheuka [2022] apresentou uma lista de proteínas que são potenciais alvos de fármacos, incluindo diversas proteínas que apresentam funções de essencialidade. Dessa busca foram encontradas seis proteínas que interseccionam com os resultados de ambos os métodos preditivos, enquanto no método baseado em ortologia não foram encontradas proteínas mencionadas na literatura mais recente. Já o trabalho de Gava et al. [2019] menciona a proteína *Mitogen-activated protein kinase* (Smp_172240.1) como envolvida no processo de desenvolvimento, sobrevivência do patógeno, características de essencialidade. A tabela V.12 apresenta as proteínas encontradas na literatura e nos resultados apresentados no método preditivo.

Tabela V.12: Proteínas essenciais encontradas na literatura e que foram indicadas nos métodos preditivos

Cód. da Proteína	**Anotação**
Smp_048430.1	Thioredoxin glutathione reductase (TGR)
Smp_198690.1	3-hydroxy-3-methylglutaryl-coenzyme A reductase
Smp_150560.1	lysine-specific histone demethylase 1
Smp_134140.1	cyclic nucleotide phosphodiesterase 4A (SmPDE4A)
Smp_078900.1	histone methyl- transferase EZH2
Smp_159710.1	Carbonic anhydrase-related
Smp_172240.1	Mitogen-activated protein kinase SmJNK

Importante destacar um trabalho de Coelho et al. [2023] publicado recentemente que realiza um estudo sobre a proteína *arginine methyltransferase 1* (SmCARM1), código Smp_029240. Experimentos foram realizados para verificar a importância dessa proteína na reprodução dos parasitos, mas o artigo indica que estudos adicionais são necessários. Essa proteína apareceu como essencial nos modelos *Random Forest* e *Gradient Boosting* e não apareceu no modelo XGBoost e nem no método baseado em homologia.

Capítulo VI Considerações Finais

Este trabalho apresentou *workflows* para duas abordagens de identificação de proteínas candidatas a essenciais. Além disso apresentou uma forma de integrar dados do DEG com a geração de uma base integrada de essencialidade com cinco organismos eucariotos. Dessa maneira é possível realizar integrações adicionais com outros bancos de dados e facilitar a condução de pesquisas com dados biológicos. Foram abordados dois métodos para identificação de proteínas, um baseado somente na sequência, o método baseado em homologia, e outro baseado em outros dados além da sequência, o método baseado em aprendizado de máquina. Por isso ambos não são comparáveis.

Com relação ao método baseado em homologia, a pesquisa utilizou a ferramenta Orthofinder para verificação de ortologia, que trouxe maior agilidade na execução dos processos, além de menos dependências principalmente com ferramentas de banco de dados. Os arquivos resultantes da ferramenta em arquivos texto são mais flexíveis para trabalhos com *scripts* de integração e limpeza de dados, trazendo maior agilidade ao trabalho. Outro ponto a destacar do método baseado em homologia é que organismos modelo que tenham um conjunto maior de proteínas essenciais dentro de seu proteoma foram mais relevantes para sugerir candidatas ao organismo alvo, caso que ocorreu entre o organismo modelo *Saccharomyces cerevisiae* e o *Schistosoma mansoni*.

Com relação ao método baseado em aprendizado de máquina, embora o XGBoost tenha apresentado resultados relevantes, os modelos *Gradient Boosting* e *Random Forest*, principalmente quando se observa o experimento de validação, apresentou melhores resultados de *recall*. Importante destacar que estes resultados são relacionados às entradas utilizadas, no caso PPI e característica de sequência, ou seja, novos tipos de entradas podem trazer outros resultados. Com relação ao balanceamento de classes, observou-se que a técnica de *Undersampling* apresentou resultados mais satisfatórios. Os três modelos XGBoost, *Random Forest* e *Gradient Boosting* classificaram mais de 3200 proteínas como essenciais, o que pode indicar que os algoritmos detectaram padrões parecidos.

As maiores limitações do trabalho foram relacionadas aos dados e ao processamento de algumas entradas. A primeira limitação foi a dificuldade com a integração de dados devido às chaves (identificadores) de proteínas que não eram compatíveis. Foi necessária a extração de uma nova chave para integrar com proteomas através de *Web Scraping* nas páginas Web do DEG. Outra limitação, com relação ao processamento, foi o custo do cálculo de medidas de centralidade em redes de PPI muito grandes, por isso foi necessário avaliar parâmetros que tornassem o cálculo

viável. Um problema enfrentado foi o desbalanceamento entre proteínas essenciais e não essenciais que necessitaram de estratégias de balanceamento.

Como evolução desse trabalho, especificamente para o método baseado em aprendizado de máquina, seria interessante a adição de mais dados relacionados a características baseadas em contexto. Expressão gênica, localização da proteína, propriedades de domínio *Gene Ontology* são mencionadas na literatura como boas entradas para algoritmos de aprendizado de máquina por estarem muita relacionadas a função de um gene e consequentemente da sua proteína. Outra entrada interessante seria a categoria (classificação funcional) do KOG (KOG - *Eukaryotic Orthologous Groups of proteins*). Além disso outra entradas que podem ser utilizadas são relacionadas ao gene, o que pode agregar mais entradas relacionadas a sequência do gene, como *GC content*, *codon usage*, que podem ajudar também a identificar essencialidade. No trabalho atual não foram utilizadas entradas relacionadas ao gene. Outra evolução desse trabalho é a adição de dados de procariotos, o que aumentaria a base e possibilitaria a aplicação de outras técnicas de aprendizado de máquina como redes neurais. Uma comparação com os ortólogos universais seria interessante para este trabalho, inclusive com os resultados dos dois métodos propostos: baseado em homologia e baseado em aprendizado de máquina,

Essa pesquisa produziu um artigo publicado no evento 15º *Brazilian Symposium on Bioinformatics* – BSB 2022 ocorrido em Setembro de 2022. O artigo intitulado *Evaluating Machine Learning Models for Essential Protein Identification* focou em um experimento de comparação de cinco algoritmos de aprendizado de máquina treinados com medidas de centralidade e *clustering* com os dados de PPI. O artigo demonstrou que *Random Forest* apresentou resultados mais satisfatórios na identificação de essencialidade [da Silva Costa et al., 2022].

Referências Bibliográficas

Altenhoff, A. and Dessimoz, C. (2012). Inferring orthology and paralogy. *Methods in molecular biology (Clifton, N.J.)*, 855:259–79. 7

Ankeny, R. A. and Leonelli, S. (2020). *Model organisms*. Cambridge University Press. 6

Aromolaran, O., Aromolaran, D., Isewon, I., and Oyelade, J. (2021). Machine learning approach to gene essentiality prediction: A review. *Briefings in Bioinformatics*, 22(5). 17, 26

Azhagesan, K., Ravindran, B., and Raman, K. (2018). Network-based features enable prediction of essential genes across diverse organisms. *PLOS ONE*, 13(12):1–13. 18

Beder, T., Aromolaran, O., Dönitz, J., Tapanelli, S., Adedeji, E., Adebiyi, E., Bucher, G., and Koenig, R. (2021). Identifying essential genes across eukaryotes by machine learning. *NAR Genomics and Bioinformatics*, 3(4). lqab110. 20

Belloze, K., Campos, L., Matias, R., Luques, I., and Bezerra, E. (2020). *A Review of Artificial Neural Networks for the Prediction of Essential Proteins*, pages 45–68. 1, 11

Bhardwaj, S. K., Vishwakarma, S., Bihari, A., Tripathi, S., Agrawal, S., and Joshi, P. (2024). Protein enzyme sequence class prediction using computational model. *GMSARN International Journal*, 18(1):62 – 70. 13

Biswas, R., Basu, A., Nandy, A., Deb, A., Haque, K., and Chanda, D. (2020). Drug discovery and drug identification using ai. In *2020 Indo – Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN)*, pages 49–51. 1

Bonaccorso, G. (2017). *Machine learning algorithms*. Packt Publishing Ltd. 13

Bonacich, P. (1987). Power and centrality: A family of measures. *American Journal of Sociology*, 92(5):1170–1182. 12

Breiman, L. (2001). Random forests. *Machine Learning*, 45:5–32. 14

Buchfink, B., Xie, C., and Huson, D. (2014). Fast and sensitive protein alignment using diamond. *Nature methods*, 12. 8

Camacho, C., Coulouris, G., Avagyan, V., Ma, N., Papadopoulos, J., Bealer, K., and Madden, T. L. (2009). Blast+: architecture and applications. *BMC Bioinformatics*, 10:421. 6

Campos, T. L., Korhonen, P. K., Gasser, R. B., and Young, N. D. (2019). An evaluation of machine learning approaches for the prediction of essential genes in eukaryotes using protein sequence-derived features. *Computational and Structural Biotechnology Journal*, 17:785–796. 9

Chavan, S. S., Shaughnessy, J. D., and Edmondson, R. D. (2011). Overview of biological database mapping services for interoperation between different 'omics' datasets. *Human Genomics*, 5:703 – 708. 10

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794, New York, NY, USA. Association for Computing Machinery. 15

Chen, X. and Ishwaran, H. (2012). Random forests for genomic data analysis. *Genomics*, 99(6):323–329. 14

Cherry, J. M., Hong, E. L., Amundsen, C., Balakrishnan, R., Binkley, G., Chan, E. T., Christie, K. R., Costanzo, M. C., Dwight, S. S., Engel, S. R., Fisk, D. G., Hirschman, J. E., Hitz, B. C., Karra, K., Krieger, C. J., Miyasato, S. R., Nash, R. S., Park, J., Skrzypek, M. S., Simison, M., Weng, S., and Wong, E. D. (2011). Saccharomyces Genome Database: the genomics resource of budding yeast. *Nucleic Acids Research*, 40(D1):D700–D705. 10

Cheuka, P. M. (2022). Drug discovery and target identification against schistosomiasis: A reality check on progress and future prospects. *Current Topics in Medicinal Chemistry*, 22(19):1595–1610. 43

Coelho, F. S., Gava, S. G., Andrade, L. F., Geraldo, J. A., Tavares, N. C., Lunkes, F. M. N., Neves, R. H., Machado-Silva, J. R., Pierce, R. J., Oliveira, G., and Mourão, M. M. (2023). Schistosoma mansoni coactivator associated arginine methyltransferase 1 (smcarm1) effect on parasite reproduction. *Frontiers in Microbiology*, 14. 43

Cunningham, F., Allen, J. E., Allen, J., Alvarez-Jarreta, J., Amode, M., Armean, I., Austine-Orimoloye, O., Azov, A., Barnes, I., Bennett, R., Berry, A., Bhai, J., Bignell, A., Billis, K., Boddu, S., Brooks, L., Charkhchi, M., Cummins, C., Da Rin Fioretto, L., Davidson, C., Dodiya, K., Donaldson, S., El Houdaigui, B., El Naboulsi, T., Fatima, R., Giron, C. G., Genez, T., Martinez, J., Guijarro-Clarke, C., Gymer, A., Hardy, M., Hollis, Z., Hourlier, T., Hunt, T., Juettemann, T., Kaikala, V., Kay, M., Lavidas, I., Le, T., Lemos, D., Marugán, J. C., Mohanan, S., Mushtaq, A., Naven, M., Ogeh, D., Parker, A., Parton, A., Perry, M., Piližota, I., Prosovetskaia,

I., Sakthivel, M., Salam, A., Schmitt, B., Schuilenburg, H., Sheppard, D., Pérez-Silva, J., Stark, W., Steed, E., Sutinen, K., Sukumaran, R., Sumathipala, D., Suner, M.-M., Szpak, M., Thormann, A., Tricomi, F. F., Urbina-Gómez, D., Veidenberg, A., Walsh, T., Walts, B., Willhoft, N., Winterbottom, A., Wass, E., Chakiachvili, M., Flint, B., Frankish, A., Giorgetti, S., Haggerty, L., Hunt, S., IIsley, G., Loveland, J., Martin, F., Moore, B., Mudge, J., Muffato, M., Perry, E., Ruffier, M., Tate, J., Thybert, D., Trevanion, S., Dyer, S., Harrison, P., Howe, K., Yates, A., Zerbino, D., and Flicek, P. (2021). Ensembl 2022. *Nucleic Acids Research*, 50(D1):D988–D995. 10

Cutler, A., Cutler, D. R., and Stevens, J. R. (2012). Random forests. *Ensemble machine learning: Methods and applications*, pages 157–175. 14

da Silva Costa, J., Rodrigues, J. G., and Belloze, K. (2022). Evaluating machine learning models for essential protein identification. In Scherer, N. M. and de Melo-Minardi, R. C., editors, *Advances in Bioinformatics and Computational Biology*, pages 38–43, Cham. Springer Nature Switzerland. 26, 33, 45

Emms, D. and Kelly, S. (2019). Orthofinder: Phylogenetic orthology inference for comparative genomics. *Genome Biology*, 20. 7, 8

Enright, A. J., Van Dongen, S., and Ouzounis, C. A. (2002). An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Research*, 30(7):1575–1584. 8

FDA (2022). The drug development process. `https://www.fda.gov/patients/learn-about-drug-and-device-approvals/drug-development-process`, Last accessed on 2022-05-12. 1

Fischer, S., Brunk, B., Chen, F., Gao, X., Harb, O., Iodice, J., Shanmugam, D., Roos, D., and Stoeckert, C. (2011). Using orthomcl to assign proteins to orthomcl-db groups or to cluster proteomes into new ortholog groups. *Current protocols in bioinformatics / editoral board, Andreas D. Baxevanis ... [et al.]*, Chapter 6:Unit 6.12.1–19. 7, 8

Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social Networks*, 1(3):215–239. 11

Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139. 14

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232. 14

Garcia, F., Guedes, G., and Belloze, K. (2020). Identifying schistosoma mansoni essential protein candidates based on machine learning. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11347 LNBI:123–128. 2, 19, 22

Garcia, F. P. and Belloze, K. T. (2018). Integração de dados na detecção de alvos para fármacos de schistosoma mansoni. In *Anais do XII Brazilian e-Science Workshop*. SBC. 2, 18

Gava, S. G., Tavares, N. C., Falcone, F. H., Oliveira, G., and Mourão, M. M. (2019). Profiling transcriptional regulation and functional roles of schistosoma mansoni c-jun n-terminal kinase. *Frontiers in Genetics*, 10. 43

Golbeck, J. (2013). Chapter 3 - network structure and measures. In Golbeck, J., editor, *Analyzing the Social Web*, pages 25–44. Morgan Kaufmann, Boston. 11, 12

Guan, X., Du, Y., Ma, R., Teng, N., Ou, S., Zhao, H., and Li, X. (2023). Construction of the xgboost model for early lung cancer prediction based on metabolic indices. *BMC Medical Informatics and Decision Making*, 23(1). 34

Guillame-Bert, M., Bruch, S., Stotz, R., and Pfeifer, J. (2023). Yggdrasil decision forests: A fast and extensible decision forests library. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2023, Long Beach, CA, USA, August 6-10, 2023*, pages 4068–4077. 15

Gurumayum, S., Jiang, P., Hao, X., Campos, T. L., Young, N. D., Korhonen, P. K., Gasser, R. B., Bork, P., Zhao, X.-M., He, L.-j., and Chen, W.-H. (2020). OGEE v3: Online GEne Essentiality database with increased coverage of organisms and human cell lines. *Nucleic Acids Research*, 49(D1):D998–D1003. 1, 10

Hadizadeh, M., Tabatabaiepour, S. N., Tabatabaiepour, S. Z., Hosseini Nave, H., Mohammadi, M., and Sohrabi, S. M. (2018). Genome-wide identification of potential drug target in enterobacteriaceae family: A homology-based method. *Microbial Drug Resistance*, 24(1):8–17. PMID: 28520499. 2, 18

Hafez, M. M., Ebeid, R. S., and Mosa, D. T. (2023). Detecting heart attacks using learning classifiers. *Information Sciences Letters*, 12(7):2859 – 2875. 13

Hagberg, A. A., Schult, D. A., and Swart, P. J. (2008). Exploring network structure, dynamics, and function using networkx. In Varoquaux, G., Vaught, T., and Millman, J., editors, *Proceedings of the 7th Python in Science Conference*, pages 11 – 15, Pasadena, CA USA. 11, 12, 26

He, X., Kuang, L., Chen, Z., Tan, Y., and Wang, L. (2021). Method for identifying essential proteins by key features of proteins in a novel protein-domain network. *Frontiers in Genetics*, 12. 19

Hughes, J., Rees, S., Kalindjian, B., and Philpott, K. (2010). Principles of early drug discovery. *British journal of pharmacology*, 162:1239–49. 1

Irion, U. and Nüsslein-Volhard, C. (2022). Developmental genetics with model organisms. *Proceedings of the National Academy of Sciences*, 119(30):e2122148119. 6

Jungck, J. R. and Viswanathan, R. (2015). Chapter 1 - graph theory for systems biology: Interval graphs, motifs, and pattern recognition. In Robeva, R. S., editor, *Algebraic and Discrete Mathematical Methods for Modern Biology*, pages 1–27. Academic Press, Boston. 11, 12

Júnior, A. F., Neto, A. F., BRAGA, J., Sandovetti, K., Feitosa, L., Xisto, T., Figueiredo, T., Lima, Y. K., and Gabriel, J. (2022). Os bancos de dados de informaÇÃo biolÓgica e sua potencial aplicabilidade Às ciÊncias mÉdicas: Uma revisÃo. *Visão Acadêmica*, 23(1). 10

Kingsford, C. and Salzberg, S. L. (2008). What are decision trees? *Nature biotechnology*, 26(9):1011–1013. 13

Koonin, E. (2005). Orthologs, paralogs, and evolutionary genomics 1. *Annual review of genetics*, 39:309–38. 6, 7

Lehne, B. and Schlitt, T. (2009). Protein-protein interaction databases: Keeping up with growing interactomes. *Human genomics*, 3:291–7. 8

Li, C., Liu, M., Zhang, Y., Wang, Y., Li, J., Sun, S., Liu, X., Wu, H., Feng, C., Yao, P., Jia, Y., Zhang, Y., Wei, X., Wu, F., Du, C., Zhao, X., Zhang, S., and Qu, J. (2023). Novel models by machine learning to predict prognosis of breast cancer brain metastases. *Journal of Translational Medicine*, 21(1). 34

Li, S., Zhang, Z., Li, X., Wang, L., and Chen, Z. (2021). An iteration model for identifying essential proteins by combining comprehensive ppi network with biological information. *BMC Bioinformatics*, 22. 8, 19

Li, W., Yin, Y., Quan, X., and Zhang, H. (2019). Gene expression value prediction based on xgboost algorithm. *Frontiers in Genetics*, 10. 15

Liu, C., Ma, Y., Zhao, J., Nussinov, R., Zhang, Y.-C., Cheng, F., and Zhang, Z.-K. (2020). Computational network biology: Data, models, and applications. *Physics Reports*, 846:1–66. Computational network biology: Data, models, and applications. 8

Lobry, J. and Gautier, C. (1994). Hydrophobicity, expressivity and aromaticity are the major trends of amino-acid usage in 999 Escherichia coli chromosome-encoded genes. *Nucleic Acids Research*, 22(15):3174–3180. 9

Luo, H., Lin, Y., Liu, T., Lai, F.-L., Zhang, C.-T., Gao, F., and Zhang, R. (2020). DEG 15, an update of the Database of Essential Genes that includes built-in analysis tools. *Nucleic Acids Research*, 49(D1):D677–D686. 1, 10, 11

Manzo, M., Giordano, M., Maddalena, L., Guarracino, M. R., and Granata, I. (2023). Novel data science methodologies for essential genes identification based on network analysis. In *Data Science in Applications*, pages 117–145. Springer. 20

Moreira, L. M. (2015). *Ciências genômicas : fundamentos e aplicações*. Sociedade Brasileira de Genética. 7

Nagai, J. S., Sousa, H., Aono, A. H., Lorena, A. C., and Kuroshu, R. M. (2018). Gene essentiality prediction using topological features from metabolic networks. In *2018 7th Brazilian Conference on Intelligent Systems (BRACIS)*, pages 91–96. 1, 11

Needleman, S. B. and Wunsch, C. D. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48:443–453. 6

Newman, M. E. J. (2010). 168Measures and metrics: An introduction to some standard measures and metrics for quantifying network structure, many of which were introduced first in the study of social networks, although they are now in wide use in many other areas. In *Networks: An Introduction*. Oxford University Press. 11, 12

Nigatu, D., Sobetzko, P., Yousef, M., and Henkel, W. (2017). Sequence-based information-theoretic features for gene essentiality prediction. *BMC Bioinformatics*, 18. 9

Palladino, W. (2009). *Conceitos de Genética*. Artmed Editora. 6

Parquet, A. (2022). Apache Parquet. `https://parquet.apache.org/`. [Online; accessed 25-feb-2022]. 28

Patil, A. (2019). Protein–protein interaction databases. In Ranganathan, S., Gribskov, M., Nakai, K., and Schönbach, C., editors, *Encyclopedia of Bioinformatics and Computational Biology*, pages 849–855. Academic Press, Oxford. 8

Pearson, W. R. (2013). An introduction to sequence similarity ("homology") searching. *Current protocols in bioinformatics*, 42(1):3–1. 6

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830. 15

Peng, C., Lin, Y., Luo, H., and Gao, F. (2017). A comprehensive overview of online resources to identify and predict bacterial essential genes. *Frontiers in Microbiology*, 8. 1, 2, 17, 22

Rigden, D. J. and Fernández, X. M. (2023). The 2023 Nucleic Acids Research Database Issue and the online molecular biology database collection. *Nucleic Acids Research*, 51(D1):D1–D8. 1, 9

Santos Filho, O. A. and Alencastro, R. B. d. (2003). Modelagem de proteínas por homologia. *Química Nova*, 26:253 – 259. 6

Schapke, J., Tavares, A., and Recamonde-Mendoza, M. (2022). Epgat: Gene essentiality prediction with graph attention networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 19(3):1615–1626. 20

Senthamizhan, V., Ravindran, B., and Raman, K. (2021). Netgenes: A database of essential genes predicted using features from interaction networks. *Frontiers in Genetics*, 12. 19

Shastry, K. A. and Sanjay, H. A. (2020). *Machine Learning for Bioinformatics*, pages 25–39. Springer Singapore, Singapore. 13

Shazadi, K. (2021). Decision tree in biology. *European Journal of Biology*, 6(1):1–15. 13

Shen, L., Zhang, J., Wang, F., and Liu, K. (2022). Predicting essential proteins based on integration of local fuzzy fractal dimension and subcellular location information. *Genes*, 13(2). cited By 0. 8, 11, 19

Shingate, P. and Sowdhamini, R. (2012). Analysis of domain-swapped oligomers reveals local sequence preferences and structural imprints at the linker regions and swapped interfaces. *PloS one*, 7:e39305. 9, 26

Szklarczyk, D., Kirsch, R., Koutrouli, M., Nastou, K., Mehryary, F., Hachilif, R., Gable, A. L., Fang, T., Doncheva, N., Pyysalo, S., Bork, P., Jensen, L., and von Mering, C. (2022). The STRING database in 2023: protein–protein association networks and functional enrichment analyses for any sequenced genome of interest. *Nucleic Acids Research*, 51(D1):D638–D646. 8, 11

Vakili, M., Ghamsari, M., and Rezaei, M. (2020). Performance analysis and comparison of machine and deep learning algorithms for iot data classification. *arXiv preprint arXiv:2001.09636*. 15, 16

Vasconcelos, E. J. R., Mesel, V. C., daSilva, L. F., Pires, D. S., Lavezzo, G. M., Pereira, A. S. A., Amaral, M. S., and Verjovski-Almeida, S. (2018). Atlas of Schistosoma mansoni long non-coding RNAs and their expression correlation to protein-coding genes. *Database*, 2018. bay068. 30

Wang, T., Birsoy, K., Hughes, N. W., Krupczak, K. M., Post, Y., Wei, J. J., Lander, E. S., and Sabatini, D. M. (2015). Identification and characterization of essential genes in the human genome. *Science*, 350(6264):1096–1101. 1

Wen, Q.-F., Wei, W., and Guo, F.-B. (2022). Geptop 2.0: Accurately select essential genes from the list of protein-coding genes in prokaryotic genomes. *Essential Genes and Genomes: Methods and Protocols*, pages 423–430. 18

WHO, W. H. O. (2022). Schistosomiasis. Accessed on January 30, 2022. 5, 6

Wunderlich, J. (2022). Updated list of transport proteins in plasmodium falciparum. *Frontiers in Cellular and Infection Microbiology*, 12. 18

Xenarios, I., Rice, D. W., Salwinski, L., Baron, M. K., Marcotte, E. M., and Eisenberg, D. (2000). Dip: the database of interacting proteins. *Nucleic acids research*, 28(1):289–291. 8

Xiaoqin, Y., Xiujuan, L., and Jie, Z. (2021). Essential protein prediction based on shuffled frog-leaping algorithm. *Chinese Journal of Electronics*, 30(4):704–711. 19

Xie, Y., Yang, Y., He, Y., Wang, X., Zhang, P., Li, H., and Liang, S. (2020). Synthetic biology speeds up drug target discovery. *Frontiers in Pharmacology*, 11. 1

Yue, Y., Ye, C., Peng, P.-Y., Zhai, H.-X., Ahmad, I., Xia, C., Wu, Y.-Z., and Zhang, Y.-H. (2022). A deep learning framework for identifying essential proteins based on multiple biological information. *BMC Bioinformatics*, 23(1). 20

Zhang, J., Li, W., Zeng, M., Meng, X., Kurgan, L., Wu, F.-X., and Li, M. (2020). Netepd: A network-based essential protein discovery platform. *Tsinghua Science and Technology*, 25(4):542–552. 2, 19, 26

Zhang, Z. and Ren, Q. (2015). Why are essential genes essential? - the essentiality of saccharomyces genes. *Microbial Cell*, 2:280–287. 1

Zhu, X., Zhu, Y., Tan, Y., Chen, Z., and Wang, L. (2022). An iterative method for predicting essential proteins based on multifeature fusion and linear neighborhood similarity. *Frontiers in Aging Neuroscience*, 13. cited By 0. 19

Zou, D., Ma, L., Yu, J., and Zhang, Z. (2015). Biological databases for human research. *Genomics, Proteomics Bioinformatics*, 42. 10