UNIVERSIDADE FEDERAL DO TOCANTINS
CAMPUS UNIVERSITÁRIO DE PALMAS
CURSO DE CIÊNCIA DA COMPUTAÇÃO

GENE FINDER EASY: UMA FERRAMENTA PARA IDENTIFICAÇÃO DE GENES

IVO PONTES ARAÚJO

PALMAS (TO)
2019

IVO PONTES ARAÚJO

GENE FINDER EASY: UMA FERRAMENTA PARA IDENTIFICAÇÃO DE GENES

Trabalho de Conclusão de Curso II apresentado à Universidade Federal do Tocantins para obtenção do título de Bacharel em Ciência da Computação, sob a orientação do(a) Prof.(a) Dr Wosley da Costa Arruda.

Orientador: Dr Wosley da Costa Arruda

PALMAS (TO)

2019

IVO PONTES ARAÚJO

GENE FINDER EASY: UMA FERRAMENTA PARA IDENTIFICAÇÃO DE GENES

Trabalho de Conclusão de Curso II apresentado à UFT – Universidade Federal do Tocantins – Campus Universitário de Palmas, Curso de Ciência da Computação foi avaliado para a obtenção do título de Bacharel e aprovada em sua forma final pelo Orientador e pela Banca Examinadora.

Data de aprovação: 17 / 4 / 2019

Banca Examinadora:

Prof. Alexandre Tadeu Rossini da Silva Dr.

Prof. Marcelo Lisboa Rocha Dr.

Prof. Horllys Gomes Barreto Dr.

Prof. Wosley da Costa Arruda Dr.

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema de Bibliotecas da Universidade Federal do Tocantins

A663g Araújo, Ivo Pontes.
Gene Finder Easy: Uma Ferramenta Para Identificação de Genes. / Ivo Pontes Araújo. – Palmas, TO, 2019.
55 f.

Monografia Graduação - Universidade Federal do Tocantins – Câmpus Universitário de Palmas - Curso de Ciências da Computação, 2019.
Orientador: Wosley da Costa Arruda

1. Bioinformática. 2. Alinhamento Múltiplo. 3. Análise Filogenética. 4. Domíno Conservado. I. Título

CDD 004

Elaborado pelo sistema de geração automatica de ficha catalográfica da UFT com os dados fornecidos pelo(a) autor(a).

UNIVERSIDADE FEDERAL DO TOCANTINS
CÂMPUS DE PALMAS
CURSO DE CIÊNCIA DA COMPUTAÇÃO
109 Norte Av. NS-15, ALCNO- 14, bloco II, Sal. 20 | 77001-090 | Palmas/TO
(63) 3232-8027 | www.uft.edu.br | comppalmas@uft.edu.br

ATA DE DEFESA DA DISCIPLINA DE PROJETO DE GRADUAÇÃO II

Ao Décimo Sétimo dia do mês de Abril de 2019 realizou-se a defesa de Projeto de Graduação, da disciplina de Projeto de Graduação II do discente **Ivo Pontes Araújo** do curso de Ciência da Computação do Campus Universitário de Palmas da Universidade Federal do Tocantins (UFT), intitulado "GENE FINDER EASY: UMA FERRAMENTA PARA IDENTIFICAÇÃO DE GENES", realizado sob a responsabilidade do Orientador Prof. Dr. Wosley da Costa Arruda. Tendo como Comissão Avaliadora, os professores: Prof. Dr. Alexandre Tadeu Rossini da Silva, Prof. Dr. Marcelo Lisboa Rocha e Prof. Dr. Horllys Gomes Barreto, os quais após avaliação, consideraram o discente APROVADO. Nada mais tendo a constar, assinaram esta Ata os presentes:

Dr. Wosley da Costa Arruda

Dr. Alexandre Tadeu Rossini da Silva

Dr. Horllys Gomes Barreto

Dr. Marcelo Lisboa Rocha

Aos meus pais, Alda e Clodomir

À Elzilane

AGRADECIMENTOS

Agradeço aos meus pais, por me apoiarem desde antes de entrar na graduação e, principalmente, por esperarem todos esses anos para que me vejam formado.

Aos meus irmãos, pelo apoio à minha decisão em cursar Ciência da Computação.

Também agradeço à minha namorada, Elzilane, que esteve comigo desde o começo da graduação.

Ao meu orientador, professor Doutor Wosley da Costa Arruda, pelo apoio e paciência durante todo esse tempo me ajudando a aprender mais sobre Bioinformática.

Aos meus amigos que tive o prazer de conhecer durante a graduação.

Aos professores do curso de Ciência da Computação, na UFT, assim como à coordenação do curso onde estagiei e tive a oportunidade de conhecer os problemas e vi os técnicos e professores trabalhando em conjunto para resolvê-los.

RESUMO

Um dos objetivos da bioinformática é identificar, assim como analisar e entender as funções de nucleotídeos e proteínas. A representação digital dessas macromoléculas é feita por letras do alfabeto justapostas que formam fitas ou sequências de informação. No processo de montagem dos genes, algumas partes das sequências podem diferir da estrutura real da sequência, o que pode ser corrigido com técnicas de bioinformática. Com vista facilitar a análise feita por pesquisadores e estudantes, este trabalho visa construir uma ferramenta que auxilia na identificação de genes por meio de de alinhamento múltiplo de sequências e análise filogenética.

Palavra-chave: Bioinformática. Alinhamento Múltiplo. Análise Filogenética. Proteína. Domínio Conservado.

ABSTRACT

One of the main goals in bioinformatics is to identify, as well as analyze and understand proteins and nucleotides functions. The digital representation of these macromolecules is made by juxtaposed alphabet letters that form string or sequences that contains information in it. In the process for assembling the genes, some portions of the sequences may differ from the actual sequence structur which can be corrected by bioinformatics techniques. In order to ease the analysis proccess done by researchers and students, this work aims to build a tool that assists the identification of potentially conserved domain genes trough multiple sequence aligment and phylogenetic analysis.

Keywords: Bioinformatics. Multiple Alignment. Phylogenetic Analysis. Protein. Conserved Domain.

ABREVIAÇÕES

DNA	Ácido Desoxirribonucleico
NCBI	National Center for Biotechnology Information
PDB	Protein Data Bank
RNA	Ácido Ribonucleico
OTU	Operational Taxonomic Unit
HTU	Hypotetical Taxonomic Unit
UPGMA	Unweighted-Pair Group Method with Arithmetic means
WPGMA	Weighted-Pair Group Method with Arithmetic means
NJ	Neighbor Joinig
PAM	Point Accepted Mutation
BLOSUM	Blocks SUbstitution Matrix
Muscle	MUltiple Sequence Comparison by Log-Expectation
EMBL	European Molecular Biology Laboratory
DDBJ	DNA Data Bank of Japan
UniProt	Universal Protein Resource
UPGMA	Unweighted-Pair Group Method with Arithmetic means
WPGMA	Weighted-Pair Group Method with Arithmetic means
MEGA	Molecular Evolutionary Genetic Analysis
RPS-BLAST	ReversePosition-SpecificBLAST

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

O genoma é um conjunto composto pelo material genético que caracteriza sua espécie, portanto, nele é feito o mapeamento do DNA, sendo possível a localização dos genes identificados ou não identificados, permitindo a manipulação genética. Como o genoma humano é fundamentalmente informação, computadores foram essenciais para determinar sequências e para aplicações em biologia e medicina (LESK, 2000).

De acordo com Verli (2014), "a bioinformática refere-se ao emprego de ferramentas computacionais no estudo de problemas e questões biológicas, abrangendo também as aplicações relacionadas à saúde humana como o planejamento de novos fármacos". Muitos dos estudos e simulações feitos *in silico* tendem a ter menor custo e gerar resultados em menos tempo.

A fim de salvar a informação do genoma de forma organizada e estruturada, as sequências são guardadas em banco de dados biológicos. Os bancos de dados podem ser públicos ou de iniciativa privada e podem ser encontrados, por exemplo, no *National Center for Biotechnology Information* (*NCBI*) e no *Protein Data Bank* (*PDB*). E com as sequências armazenadas, é possível comparar uma sequência de RNA ou proteína com o propósito de encontrar sequências no banco de dados com estruturas similares.

Frederick Sanger ganhou o Prêmio Nobel por determinar a sequência de aminoácidos da insulina, a proteína usada para tratar diabetes. A técnica de Sanger digeriu a insulina com proteases e sequenciou seus fragmentos. Por sobreposição, ele reconstruiu a sequência inteira de aminoácidos. Seu trabalho influenciou diretamente o primeiro sequenciamento de RNA (PEVZNER, 2008).

Com técnicas de sequenciamento cada vez mais eficazes, os bancos de dados biológicos precisam de mais armazenamento para guardar os genomas sequenciados. Além disso, algoritmos foram criados para alinhar as sequências. Os alinhamentos são técnicas de comparação entre duas ou mais sequências biológicas, que buscam séries de caracteres individuais que se encontram na mesma ordem nas sequências analisadas (VERLI, 2014).

Os algoritmos podem alinhar aminoácidos ou nucleotídeos, onde cada molécula é representada por uma letra do alfabeto. A ideia central dos algoritmos de alinhamento é encontrar as similaridades para obter o máximo de letras correspondentes entre as sequências. Desse modo, é possível obter informações sobre o grau de ancestralidade entre os genes, por exemplo. Esse grau de ancestralidade é chamado de homologia (PATCHER; STURMFELS, 2005).

Outrossim, em relação à quantidade, os alinhamentos são divididos em dois tipos: simples e múltiplos. Os alinhamentos simples analisam apenas duas sequências por vez. Por outro lado, os alinhamentos múltiplos analisam três ou mais sequências. Quanto a

localização, os alinhamentos podem ser divididos em alinhamento global, que busca todas as sequências do início ao fim, e alinhamento local que busca por regiões específicas e menores.

Um dos algoritmos de alinhamento mais importantes é o Needleman–Wunsch que faz alinhamento global e utiliza programação dinâmica para comparar as sequências. Já outro algoritmo que teve muita relevância foi o Smith-Waterman, que faz o alinhamento local de sequências biológicas.

Além disso, é importante reiterar que o algoritmo de Needleman–Wunsch foi essencial para fazer o alinhamento global de sequências e o algoritmo de Smith-Waterman apresentou uma solução fundamental que ainda é utilizada hoje no alinhamento local pelo *Clustal Omega*, por exemplo.

A programação dinâmica é uma técnica que resolve o problema de alinhamento de múltiplas sequências.Desde que o custo dessa abordagem com programação dinâmica é $O(2n^k)$, onde n é o tamanho da sequência e k é a quantidade de sequências, variações desses algoritmos foram criadas, e uma delas é a aplicação de heurísticas para otimizar o alinhamento de sequências múltiplas (PEVZNER, 2008).

A filogenia ajuda visualizar relações de ancestralidade e similaridade de organismos, que uma vez alinhados podem ser agrupados em famílias ou subfamílias. Essas informações facilitam na escolha de quais organismos devem ser analisados para encontrar os possíveis domínios conservados compartilhados entre eles.

1.1 Justificativa

Como já citado, o alinhamento múltiplo de sequências é uma técnica bastante utilizada para identificar relacionamentos entre sequências, já que o alinhamento é o primeiro passo para várias outras técnicas de análise, como a análise filogenética e identificação de homologia entre os organismos estudados. O *software Clustal Omega* é utilizado em diversas ferramentas de bioinformática, ele é o sucessor do *ClustalW*, esse ainda utilizado em ferramentas muito conhecidas como *Molecular Evolutionary Genetic Analysis* (*MEGA*).

Outra técnica utilizada em bioinformática é a análise filogenética. Existem vários *sofwares* que reconstroem árvores filogenéticas e um muito importante é o *MEGA*. O maior problema dessa aplicação é que apesar de ser possível ver e executar alinhamentos em formato do *Clustal* como os formatos ALN e MSF, há uma necessidade de compilar os domínios conservados de sequências diferentes para novos arquivos no formato FASTA a fim de gerar novos alinhamentos.

Tendo em conta os pontos aqui abordados, este projeto visa construir uma ferramenta que automatize esse processo que pode ser necessário utilizar até 3 *softwares* diferentes para resolver esse problema de descobrir os domínios conservados a partir de um conjunto de sequências.

1.2 Objetivos

1.2.1 Objetivo Geral

Desenvolver um *software* para identificar genes com domínios conservados.

1.2.2 Objetivos Específicos

1. Implementar um servidor *web* para ser possível executar a aplicação a partir de qualquer sistema
2. Implementar um *software* cliente inicialmente para plataforma web e posteriormente multiplataforma
3. Gerar Árvores Filogenéticas
4. Alinhar sequências a partir de um grupo selecionado de sequências
5. Disponibilizar um arquivo de alinhamento com as sequências selecionadas

1.3 Organização do Projeto

No Capítulo 2, toda a fundamentação teórica é abordada, desde o dogma Central da Biologia Computacional, aminoácidos, ácidos nucleicos e proteínas até os conhecimentos de filogenia, alinhamentos de sequências, assim como algumas ferramentas de bioinformática.

No Capítulo 3, a metodologia do projeto é descrita, partindo da linguagem de programação usada às métricas aplicadas.

Agora no Capítulo 4 são explicados a arquitetura e o funcionamento do *GENE FINDER EASY*.

No Capítulo 5 mostra os experimentos seus respectivos resultados obtidos após a execução do *GENE FINDER EASY*.

E finalmente, no Capítulo 6, são apresentadas conclusões sobre o presente trabalho, juntamente com as propostas de trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Considerações Iniciais

Este capítulo aborda os conceitos básicos e necessários a compreensão deste estudo, dentro das áreas de biologia molecular e bioinformática. Primeiro serão abordados os temas aminoácidos e proteínas, logo após, haverá uma introdução sobre os conceitos de ácidos nucleicos.

Subsequentemente, os conceitos explicados serão a expressão gênica, homologia e filogenia. Ainda, os conceitos de bioinformática abordados estão nessa ordem: banco de dados biológicos, alinhamento múltiplo de sequências, técnicas de alinhamento múltiplo de sequências e ferramentas de bioinformática.

2.2 Ácidos Nucleicos e Síntese de Proteínas

Segundo Tortora, Funke e Case (2012),"O DNA e outra substância denominada ácido ribonucleico (RNA) são designados em conjunto como ácidos nucleicos, pois foram descobertos pela primeira vez nos núcleos das células". E diferente dos aminoácidos, os ácidos nucleicos têm os nucleotídeos como unidades estruturais.

Os ácidos nucleicos têm grande importância para todos os organismos vivos. Pode-se ressaltar que é dos ácidos nucleicos que as células recebem informações sobre síntese de proteínas e função de moléculas. Em outras palavras, os ácidos nucleicos guardam e transmitem informação genética na célula (ZAHA; FERREIRA; PASSAGLIA, 2014).

Existem dois tipos de ácidos nucleicos, o ácido desoxirribonucleico (DNA) e o ácido ribonucleico (RNA) que são polímeros, sequências de moléculas similares de nucleotídeos, unidos por ligações fosfodiéster. Uma ligação fosfodiéster é o resultado de um grupamento hidroxílico ao C3 do açúcar de um nucleotídio, ligação fosfato com éster. Essa ligação gera uma desidratação e gera uma estrutura de bases ligadas (ZAHA; FERREIRA; PASSAGLIA, 2014).

Da mesma forma que as proteínas são formadas por aminoácidos, os ácidos nucleicos são formados por nucleotídeos, as unidades estruturais do ácidos nucleicos. De acordo com Tortora, Funke e Case (2012), "Cada nucleotídeo tem três partes: a base nitrogenada, uma pentose (açúcar de cinco carbonos) denominada desoxirribose ou ribose e um grupo fosfato (ácido fosfórico)", conforme apresenta-se na Figura 2.1.

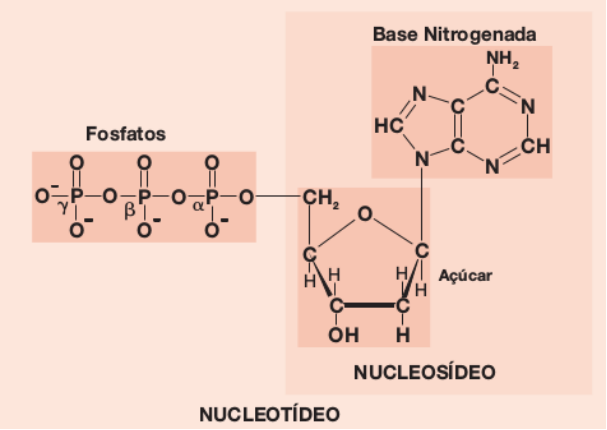

Figura 2.1 – Estrutura dos Nucleotídeos. Adaptada de Zaha, Ferreira e Passaglia (2014).

Essas bases são a adenina, timina, citosina, guanina e uracila, representadas respectivamente pelas letras A, T, C, G, U. As bases A e G são chamadas de purinas, e as bases T, C e U são chamadas de pirimidinas, como é visto na Figura 2.2. Além disso, existe o nucleosídeo que é uma purina ou pirimidina mais uma pentose.

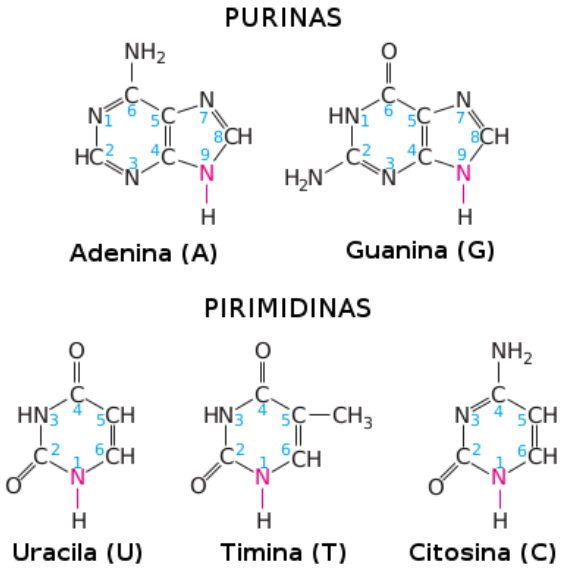

Figura 2.2 – As bases nitrogenadas. Adaptada de Lodish et al. (2008)

2.2.1 DNA

O Ácido desoxirribonucleico, chamado de DNA, é a substância que compõe os genes. A molécula do DNA é estruturada em duas cadeias antiparalelas que se dobram e estão ligadas em forma de dupla hélice. É o DNA que mantém a informação genética de um organismo Tortora, Funke e Case (2012).

Essa estrutura de nucleotídeos ligados, é representada em orientação 5' $\rightarrow$ 3'. Essa cadeia rege a sequência que gera a estrutura primária do DNA (ZAHA; FERREIRA; PASSAGLIA, 2014).

Em 1953, houve uma importante descoberta da organização feita por DNA de James Watson e Francis Crick. Eles descobriram que os nucleotídeos são unidos por ligações químicas entre fosfato e açúcar. A sequência de bases nitrogenadas constituem duas cadeias de nucleotídeos e como a ordem das combinações é sempre A-T e G-C, pode-se dizer que as cadeias do DNA são complementares, como observa-se na Figura 2.3 (SNUSTAD; SIMMONS, 2013).

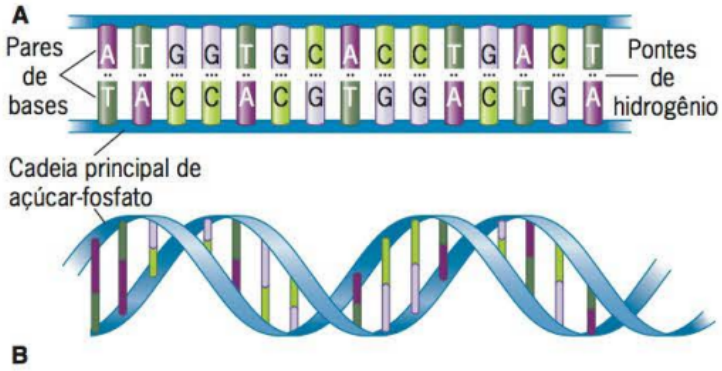

Figura 2.3 – Estrutura da cadeia de DNA. Adaptada de Snustad e Simmons (2013)

2.2.2 RNA

De acordo com Zaha, Ferreira e Passaglia (2014), "O ácido ribonucleico (RNA) é a molécula de ácido nucleico formada, em geral, por uma única cadeia com grande diversidade de conformações". Essa cadeia é composta por sequências de bases que constituem as estruturas do RNA.

A estrutura primária do RNA é similar à do DNA, porém se tem uracila (U) no lugar da timina (T). Na estrutura secundária do RNA, ocorre pareamento entre as bases C-G e A-U. Já na estrutura terciária, existem características referentes às funções que são obtidas a partir de interações de molécula.

Existem três principais tipos de RNAs, são eles o mRNA, o tRNA e o rRNA. O RNA mensageiro (mRNA) carrega instruções do DNA com a ordem correta dos aminoácidos, durante a síntese de proteína. A junção de aminoácidos para a síntese de proteínas ocorre por tradução do mRNA. Nesse processo, a informação que está no mRNA é interpretada pelo RNA transportador (tRNA) com o auxílio do RNA ribossômico (rRNA) (LODISH et al., 2008).

A Tabela 2.1 mostra as classes de RNAs, em qual parte do processo celular estão e suas respectivas funções. É mais discutido sobre as classes de mRNA, tRNA e rRNA a seção 2.4 devido à sua importância no dogma central da biologia molecular.

Tipo	Processo Celular	Classe	Função
Codificador	Síntese de proteínas	mRNA (mensageiro)	Contém a sequência para a síntese das proteínas
Não Codificador	Síntese de proteínas	rRNA (ribossômico)	Forma junto com as proteínas ribossômicas os ribossomos. Local da síntese de proteínas
		tRNA (transportador)	Transporta os aminoácidos ao local da síntese das proteínas
	Processamento do rRNA e tRNA	snoRNA (RNA nucleolar pequeno)	Maturação de ribossomos e RNAs transportadores
	Processamento do pré-RNA	snRNA (RNA nuclear pequeno)	Síntese do mRNA eucariótico
	Regulação da expressão gênica	sRNA (RNA pequeno)	Modula a tradução e a estabilidade de mRNAs
		CRISP RNA (RNA CRISP)	Interfere com a infecção por bacteriófagos e com a conjugação
		miRNA (micro RNA)	Silenciamênto gênico. Reprime a tradução ou cliva mRNAs alvo
		siRNA (RNA de interferência pequeno)	Silênciamento gênico. Cliva mRNAs(vírus)
		piRNA(RNA que liga a proteína PIWI)	Silenciamento gênico em células germinativas
		lncRNA(RNA não codificador longo)	Muito transcrito. "Transcrição generalizada"

Tabela 2.1 – Classes de RNAs. Adaptada de Zaha, Ferreira e Passaglia (2014).

2.3 Aminoácidos e Proteínas

2.3.1 Aminoácidos

Os aminoácidos são ácidos orgânicos que possuem um carbono ligado a quatro grupamentos químicos diferentes. Como existem 20 aminoácidos, uma proteína pode ter 20^n sequências, ou seja, se uma proteína tem um tamanho de 152 aminoácidos em sequência, logo, ela terá 20^{152} possíveis sequências de proteínas.

Na Tabela 2.2 mostra os 20 aminoácidos, juntamente com suas representações em 3 letras e 1 letra. Essas representações são utilizadas em locais e/ou propósitos específicos, por exemplo, a representação de 1 letra é utilizada para o alinhamento de sequências. As proteínas podem servir como componentes estruturais de uma célula, sensores que podem moldar a temperatura e propriedades da célula. Um exemplo são as enzimas, que catalisam o processo de reações químicas (LODISH et al., 2008).

Aminoácido	**3 letras**	**1 letra**
Alanina	Ala	A
Cisteína	Cys	C
Ác. Aspártico	Asp	D
Ác. glutâmico	Glu	E
Fenilalanina	Phe	F
Glicina	Gly	G
Histidina	His	H
Isoleucina	Ile	I
Lisina	Lys	K
Leucina	Leu	L
Metionina	Met	M
Asparagina	Asn	N
Prolina	Pro	P
Glutamina	Gln	Q
Arginina	Arg	R
Serina	Ser	S
Treonina	Thr	T
Valina	Val	V
Triptofano	Trp	W
Tirosina	Tyr	Y

Tabela 2.2 – Estrutura dos 20 aminoácidos. Adaptada de Verli (2014).

É possível obter aminoácidos sintetizando a partir de outras moléculas ou quebrando proteínas ingeridas. Quando uma cadeia de aminoácidos é criada, ela se dobram e criam formas complexas, onde cada cadeia de aminoácidos tem um estrutura tridimensional e função diferentes (LODISH et al., 2008).

Somente quando uma estrutura tridimensional da proteína estiver correta, a proteína terá uma função eficiente. Para conseguir entender o funcionamento das proteínas, é preciso entender as estruturas tridimensionais formadas pelas sequências de aminoácidos.

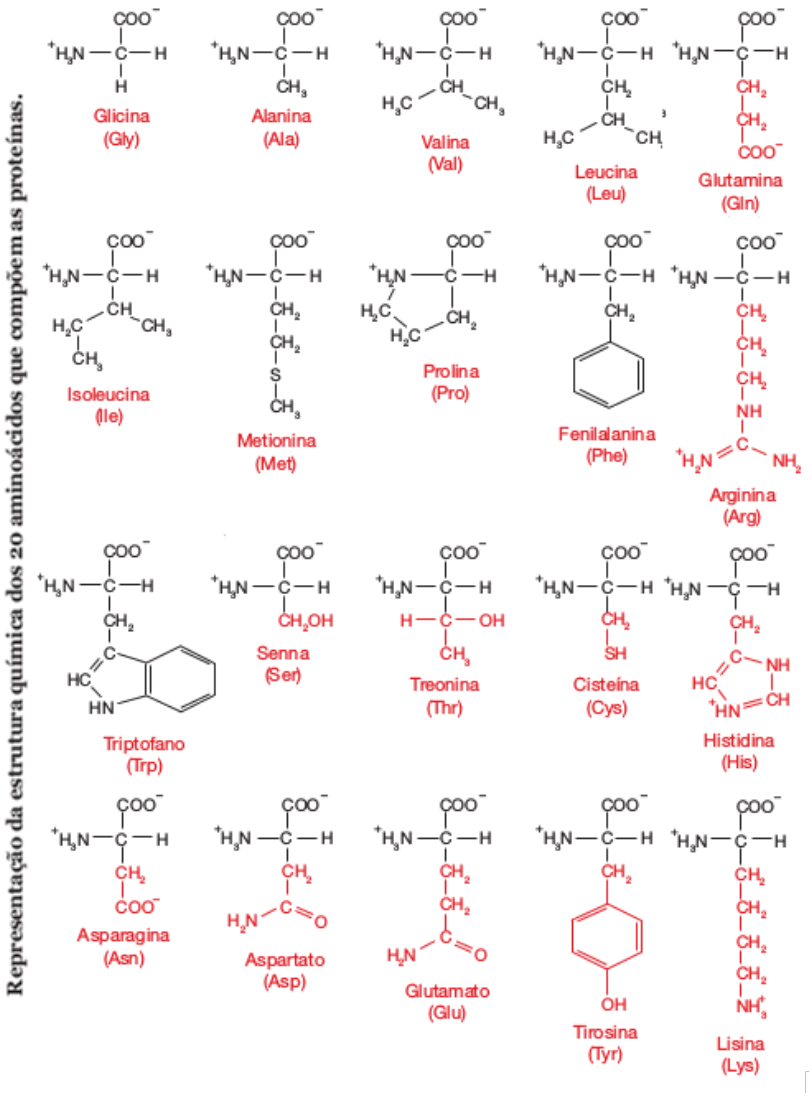

Figura 2.4 – Estrutura dos 20 aminoácidos. Adaptada de Zaha, Ferreira e Passaglia (2014).

2.3.2 Proteínas

Segundo Verli (2014), "As proteínas são polímeros sintetizados pelas células a partir de aminoácidos", conforme observa-se na Figura 2.4. São moléculas bem versáteis, capazes de serem muito grandes ou muito pequenas. As proteínas comumente têm tamanho entre 100 a 10000 aminoácidos, mas algumas podem ser muito menores ou muito maiores.

Além disso, de acordo com Zaha, Ferreira e Passaglia (2014), "As proteínas resultam da expressão da informação contida no gene". Por isso, é o gene que determinará a sequência de aminoácidos de uma proteína específica.

Algumas proteínas que são similares são consideradas membros de uma família de proteínas. Algumas proteínas já têm suas funções conhecidas, como trabalhar em locais específicos dentro de uma célula (LODISH et al., 2008). Como foi dito anteriormente, a estrutura da proteína determina sua função, logo, os as características estruturais da proteína devem ser analisadas, pois desempenham diversas funções biológicas.

As proteínas têm quatro níveis de organização. A estrutura primária, Figura 2.5, da proteína é um arranjo linear simples de aminoácidos. De acordo com Zaha, Ferreira e Passaglia (2014), "são as ligações peptídicas que estabilizam esse tipo de estrutura". As ligações peptídicas são ligações formadas entre dois aminoácidos. Além disso, cada proteína tem sua estrutura primária que compõe a estrutura tridimensional.

Figura 2.5 – Estrutura Primária das Proteínas. Adaptada de Zaha, Ferreira e Passaglia (2014).

Os autores Tortora, Funke e Case (2012, grifo nosso) dizem que, "A estrutura secundária de uma proteína, Figura 2.6, é o dobramento localizado e repetitivo da cadeia polipeptídica". Assim, as estruturas secundárias das proteínas podem ser tanto espirais em sentido horário, as hélices e as dobras pregueadas, que se formam a partir de partes quase paralelas das sequências.

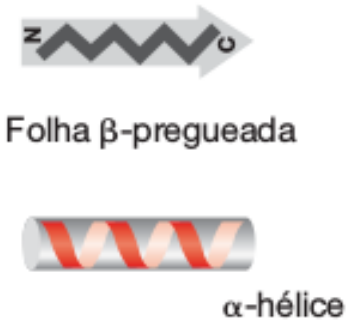

Figura 2.6 – Estrutura Secundária das Proteínas. Adaptada de Zaha, Ferreira e Passaglia (2014).

De acordo com Verli (2014, grifo nosso), "A estrutura terciária, Figura 2.7, de uma biomolécula corresponde à montagem dos seus elementos de estrutura secundária". Além disso, é a estrutura terciária que irá exercer a função biológica de uma molécula.

Figura 2.7 – Estrutura Terciária das Proteínas. Adaptada de Zaha, Ferreira e Passaglia (2014).

É importante ressaltar sobre o processo de dobramento, *folding*, pois é o processo de organização das estruturas secundárias. Nesse processo, a combinação tenta adotar uma conformação (função) mais estável.

Ainda, existem algumas proteínas que têm uma estrutura quaternária. Além do mais, Tortora, Funke e Case (2012, grifo nosso) conceituam que "uma estrutura quaternária, Figura 2.8, consiste em uma agregação de duas ou mais cadeias polipeptídicas, que operam como uma unidade funcional única". As ligações da estrutura quaternária são as mesmas que as da estrutura terciária.

Figura 2.8 – Estrutura Quaternária das Proteínas. Adaptada de Zaha, Ferreira e Passaglia (2014).

O objetivo da comparação de sequência de proteínas é descobrir similaridades estruturais ou funcionais entre as proteínas. Outro objetivo é o de identificar todas as proteínas do organismo para conseguir sequenciar um genoma, que é um conjunto completo de genes de um organismo (LODISH et al., 2008).

2.3.3 Domínios Proteicos

Algumas proteínas caracterizadas como longas são subdividas em regiões conhecidas como domínios, que têm variação de tamanho entre 100-150 aminoácidos e esses aminoácidos situados em domínios normalmente são formados por combinações de *motifs*. Esses, também chamados de motivos, são conjuntos de aminoácidos que se repetem e seguem um padrão específico. Nas proteínas é possível encontrar vários *motifs* nos quais possuem funções e estruturas diferentes (LODISH et al., 2008).

Nos estudos de microbiologia molecular, os domínios são testados e combinados de formas e arranjos diferentes a fim de sintetizar novas proteínas, com diferentes funções. Já os domínios conservados são formados pelos *motifs*, os quais são analisados por pesquisadores com ferramentas de bioinformática, já que recorrências podem ser encontradas mais rapidamente com técnicas computacionais (NCBI. . . , 2019).

2.4 Expressão Gênica

O fluxo da informação genômica segue na ordem DNA → RNA → polipeptídio, uma sequência que geralmente é considerada como o dogma central da biologia molecular. Ainda, Snustad e Simmons (2013) ressalta que "o dogma central da biologia é que as informações armazenadas no DNA são transferidas para as moléculas de RNA durante a transcrição e para as proteínas durante a tradução".

As informações das moléculas são passadas de DNA para DNA e RNA para RNA. No caso das proteínas, o processo de transferência de informações leva duas etapas: A transcrição e a tradução. A transição transfere informações para o RNA e logo após ocorre a tradução onde o RNA transfere as informações para as proteínas (SNUSTAD; SIMMONS, 2013).

A transcrição é a síntese de um a fita complementar de RNA a partir de um molde de DNA. Para isso é usada uma enzima chamada RNA-polimerase, a partir de uma parte do DNA que se torna mRNA, que é um RNA do tipo mensageiro, transforma a informação codificada no DNA em proteínas específicas do DNA (TORTORA; FUNKE; CASE, 2012).

O processo de síntese do RNA começa em uma região de iniciação, chamada região promotora e só para o processo de transcrição quando o RNA-polimerase encontra uma região de terminadora.

Após a transcrição, ocorre a tradução, que produz as proteínas, a partir das informações genéticas do mRNA. O local onde ocorre essa tradução é o ribossomo, onde o tRNA reconhece grupos de três nucleotídeos chamados códons, esses determinam quais aminoácidos serão gerados na sequência da proteína e finalmente a proteína é transportada para o local onde ela exercerá sua função (GRAUR; LI, 2000).

A tradução começa quando o tRNA encontra um códon de iniciação e vai até encontrar um códon de terminação. A Tabela 2.3 abaixo mostra as bases e seus respectivos os códons. Os códons em destaque são **AUG**, códon de iniciação e os códons **UAA**, **UAG** e **UGA**, códons de terminação.

	Segunda base do código genético								
Primeira base do código genético		U		C		A		G	
	U	UUU	Phe	UCU	Ser	UAU	Tyr	UGU	Cys
		UUC		UCC		UAC		UGC	
		UUA	Leu	UCA		**UAA**	STOP	**UGA**	STOP
		UUG		UCG		**UAG**		UGG	Trp
	C	CUU	Leu	CCU	Pro	CAU	His	CGU	Arg
		CUC		CCC		CAC		CGC	
		CUA		CCA		CAA	Gln	CGA	
		CUG		CCG		CAG		CGG	
	A	AUU	Ile	ACU	Thr	AAU	Asn	AGU	Ser
		AUC		ACC		AAC		AGC	
		AUA		ACA		AAA	Lys	AGA	Arg
		AUG	Met	ACG		AAG		AGG	
	G	GUU	Val	GCU	Ala	GAU	Asp	GGU	Gly
		GUC		GCC		GAC		GGC	
		GUA		GCA		GAA	Glu	GGA	
		GUG		GCG		GAG		GGG	

Tabela 2.3 – Códigos Genéticos. Adaptada de Zaha, Ferreira e Passaglia (2014).

2.5 Homologia

A homologia é uma forma de classificação baseada em similaridades e diferenças entre sequências de aminoácidos, proteínas e nucleotídeos. Esses organismos com ancestral comum são chamadas de homólogas. A evidência principal de homologia as proteínas um ancestral comum entre elas, que pode ser entendida como uma similaridade entre as sequências de proteínas.

Logo, as proteínas homólogas podem ser classificadas como da mesma família, a partir de uma comparação de sequências de organismos. O dobramento das estruturas tridimensionais das proteínas é similar mesmo em partes que aparentemente não exista evidência de homologia em suas estruturas primárias (LODISH et al., 2008).

2.6 Filogenia

A filogenia é a representação do histórico de ramificação das rotas de heranças dos organismos. Além disso os estudos filogenéticos têm objetivo de reconstruir traços genealógicos entre organismos para comparar diferenças de tempo e características genéticas, por exemplo (GRAUR; LI, 2000).

As relações entre organismos são representadas em uma árvore filogenética. A árvore filogenética pode ser caracterizada como um grafo onde cada nó se conecta com

dois nós adjacentes.

Os nós terminais (folhas), são chamados de Unidades Operacionais Taxonômicas, *Operational Taxonomic Units (OTUS)*. Os OTUs podem ser espécies, genes, organismos inteiros ou sequências. Já os nós internos, são chamados de Unidades Hipotéticas Taxonômicas, *Hypotetical Taxonomic Units (HTUs)*, esses representam as unidades ancestrais.

De acordo com Ticona (2008),"É importante salientar que as árvores filogenéticas podem ter ou não raiz". Uma árvore com raiz implica que há um ancestral em comum entre todas as espécies e a distância representa o grau de antiguidade da OTU.

Árvores filogenéticas podem ser representadas de várias formas, cladograma inclinado, cladograma retangular, árvore radial, árvore livre, filograma e dendrograma. Dentre as formas descritas, o cladograma mostra a estrutura da árvore, o filograma tem a característica de seus ramos terem comprimento relativo às diferenças genéticas e o Dendrograma é voltado à diferença temporal dos organismos.

A Figura 2.9 exemplifica a estrutura de uma árvore evolucionária, ao analisar as similaridades entre guaxinins e ursos. Árvores geradas a partir de matrizes de distâncias funcionam em dois passos, primeiramente é necessário realizar o cálculo das distâncias genéticas, as quais o resultados são dispostos na forma de matriz, por fim é feito a reconstrução da árvore a partir destes dados.

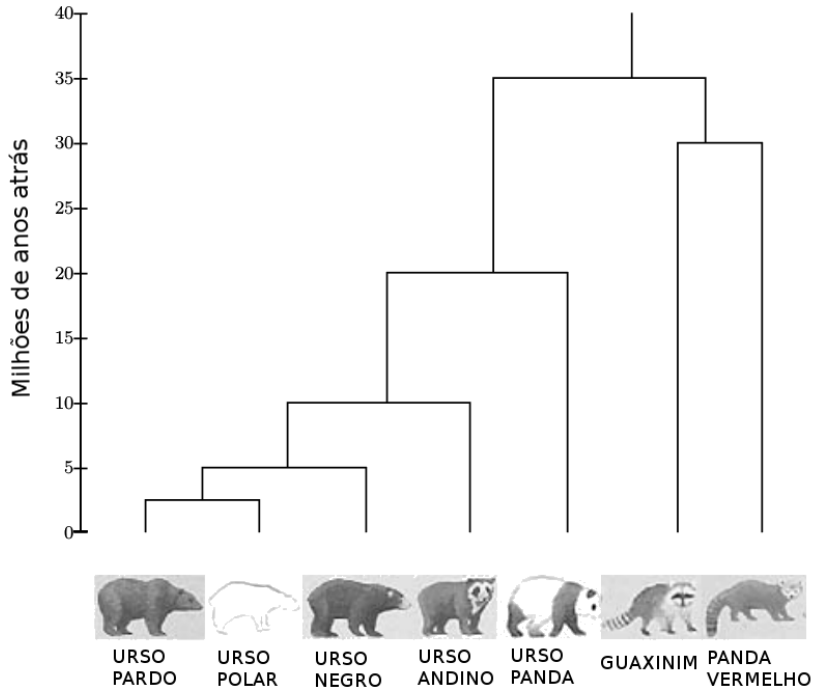

Figura 2.9 – Uma árvore evolucionária que mostra a diferença entre guaxinins e ursos. Apesar do tamanho e formas, as duas famílias são muito próximas. Adaptada de Jones e Pevzner (2004).

A matriz de distância é uma matriz triangular superior ou inferior com diagonal nula, contendo as distâncias evolutivas entre as espécies de estudo. Estas matrizes podem

ser geradas por meio de programas que realizam o alinhamento e o analisam através de sequências de nucleotídeos ou aminoácidos. Em geral métodos de alinhamento levam em consideração a mutação de um nucleotídeo para outro. Inclusive, o Neighbor-Joining realiza os cálculos com base nas diferenças entre as sequências analisadas.

Os métodos mais conhecidos de agrupamento para reconstrução filogenética são o *Unweighted-Pair Group Method with Arithmetic means* (*UPGMA*), o *Weighted-Pair Group Method with Arithmetic means* (WPGMA) e o Neighbor-Joining. Esses métodos obtêm resultados mais rapidamente e com uma quantidade reduzida de soluções.

Este trabalho utiliza o método Neighbor-Joining devido à sua eficácia, já que as distâncias das sequências são geradas seguindo o princípio da evolução mínima. Os métodos UPGMA é o método que tem menos eficiência dentre os métodos citados, já que usa apenas uma média para calcular o agrupamento.

Já o WPGMA tem mais eficiência do que o UPGMA por utilizar a correção das distâncias utilizando o modelo de Jukes-Cantor. Embora o WPGMA tenha mais eficácia que o UPGMA, ambos os métodos só produzem árvores corretas se todos os ramos estiverem a mesma distância da raiz (UNICAMP, 2004).

2.7 Bancos de Dados Biológicos

Atualmente, utiliza-se bancos de dados de sequências para tentar retirar informações relevantes sobre os organismos vivos. Com essas informações, é possível fazer as anotações dos genes desses organismos. Alguns dos mais conhecidos são o *National Center for Biotechnology Information* (*NCBI*) e o *Protein Data Bank* (*PDB*).

O principal banco de dados de estrutura de macromoléculas é o *PDB*. Ele contém estruturas de proteínas, ácidos nucleicos e uns poucos carboidratos. E com as sequências armazenadas, é possível comparar uma sequência de RNA ou proteína com o propósito de encontrar sequências no banco de dados com estruturas similares (LESK, 2000).

O *Conserved Domain Database* (*CDD*) tem uma coleção de alinhamentos que representam os domínios conservados. O CDD tem domínios curados pela NCBI, ou seja, verificado se há possíveis erros de anotação, e também domínios adicionados por outros pesquisadores. O objetivo do banco CDD é dar uma ideia de como são os relacionamentos e propriedades funcionais dos dos domínios conservados (VERLI, 2014).

O *NCBI* possui *links* dos registros dos domínios conservados das sequências de proteínas do CDD. É possível executar o BLAST e o PSI-BLAST por meio de uma interface *web*. No banco de dados, é possível conseguir os modelos de domínios conservados a partir de uma *query*. O algoritmo utilizado na busca dos domínios é o RPS-BLAST, *Reverse Position-Specific BLAST*, uma variação do PSI-BLAST (MARCHLER-BAUER et al., 2010).

Existem vários outros bancos de dados com diferentes propósitos como por exem-

plo, o *GenBank*, o *European Molecular Biology Laboratory* (*EMBL*) e o *DNA Data Bank of Japan (DDBJ)*. Ainda, existem bancos de dados específicos, como o Rfam, para estrutras de RNAs e o *Universal Protein Resource (UniProt)*, para estudos de proteínas (ZOU et al., 2015).

2.8 Alinhamento de Sequências

O alinhamento de sequências pode ser entendido como a similaridade entre duas sequências biológicas. Além de ser um dos problemas básicos em bioinformática, já que vários outros problemas necessitam do alinhamento de sequências para sua solução, o alinhamento de sequências é uma técnica frequentemente utilizada por quem estuda biologia molecular.

Deve-se atentar que se há comparações entre duas sequências e essas são homólogas, semelhantes, consegue-se inferir características da sequência, como conformação e funções a partir de suas estruturas.

Em uma comparação de sequências, utiliza-se a distância de Hamming, que fornece uma média simples de comparação de duas strings. Ela é definia por duas strings do mesmo tamanho como o número de posições, porém com letras diferentes. Também há a distância de Levenshtein, na qual é a sequência entre duas sequências de qualquer tamanho (CROCHEMORE; HANCART; LECROQ, 2007).

Os alinhamentos são divididos em dois tipos: os alinhamentos simples e os alinhamentos múltiplos. Os alinhamentos simples analisam apenas duas sequências por vez. Por outro lado, os alinhamentos múltiplos analisam três ou mais sequências.

Quanto a localização, os alinhamentos podem ser divididos em alinhamento global, que busca todas as sequências do início ao fim, e alinhamento local que busca por regiões específicas e menores.

2.8.1 Alinhamento Global

Inicialmente, o algoritmo criado para o alinhamento de sequências de aminoácidos foi o Needleman-Wunsch, que fazia o alinhamento global, uma comparação entre as duas sequências completas, procurando a maior similaridade entre elas, assim como é visto na Figura 2.10. Ainda é possível afirmar que o algoritmo inicial de alinhamento de Needleman-Wunsh é executado em tempo cúbico (CROCHEMORE; HANCART; LECROQ, 2007).

```
--T--CC-C-AGT--TATGT-CAGGGGACACG--A-GCATGCAGA-GAC
  |  || |  ||  | | | |||    || |  | |  | ||||   |
AATTGCCGCC-GTCGT-T-TTCAG----CA-GTTATG--T-CAGAT--C
```

Figura 2.10 – Um alinhamento Global. Adaptada de Jones e Pevzner (2004).

2.8.2 Alinhamento Local

O objetivo do alinhamento local é encontrar regiões altamente similares, que não necessariamente precisam ser o tamanho completo das sequências, conforme observa-se na Figura 2.11. Assim, existe maior possibilidade de encontrar similaridades entre um conjunto de sequências. Porém, o custo de execução do algoritmo chega a tempo quadrático (CROCHEMORE; HANCART; LECROQ, 2007).

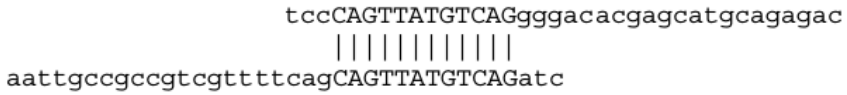

Figura 2.11 – Um alinhamento Local. Adaptada de Jones e Pevzner (2004).

2.8.3 Alinhamento Múltiplo de Sequências

O alinhamento múltiplo de sequências foi proposto como um método de programação dinâmica para tratar problemas com mais de duas sequências biológicas. O método utiliza a soma ponderada de pares, para calcular os pares de sequências (VERLI, 2014).

A cada alinhamento entre duas sequências, há um *score* que representa sua respectiva pontuação. As formas de pontuação são as seguintes: +1 ponto, chamado de *match*, caso sejam dois caracteres idênticos, -1 ponto, chamado de *mismatch*, caso sejam caracteres diferentes e -2 caso uma coluna esteja vazia (SETUBAL; MEIDANIS, 1997).

Assim, o melhor alinhamento terá o maior *score*. Porém, existe um problema referente aos espaços, onde se existe uma sequência de espaços p $>$ 1, por exemplo, essa sequência é chamada de *gap*.

Um sistema de pontuação(*score*) precisa considerar a substituição de resíduos, inserções ou deleções. Uma medida de divergência de sequências é o *Point Accepted Mutation* (*PAM*); 1 *PAM* = 1% de mutação aceita(1 *Point Accepted Mutation*) (LESK, 2000).

No algoritmo Needleman-Wunsch, a matriz *PAM* é utilizada como referência para aplicação de penalidades e cálculo de *score*, de acordo com a Tabela 2.4. A relação entre o *score PAM* e a percentagem de identidade entre sequências é:

Tabela 2.4 – Relação de Identidade - PAM

PAM	0	30	80	110
IDENTIDADE	100	75	50	60

Como a matriz *PAM* tinha um cálculo de *score* baseado em estimativa com logaritmos(log-odds), há uma margem de erro no processo. Por isso, S. Henikoff e J.G.Henikoff desen-

volveram a matriz *BLOSUM* para calcular o *score* de aminoácidos nas sequências de proteínas.

O objetivo principal foi substituir a matriz *PAM* por uma matriz que tivesse um melhor desempenho com sequências mais distantes. O nome *BLOSUM* é um acrônimo para *Blocks SUbstitution Matrix*, uma matriz de substituição baseada em BLOCKS, um banco de dados de sequências de proteínas alinhadas. Com um limite mínimo de 62% de identidade, obtem-se a matriz BLOSUM62, que é a matriz mais utilizadas na maioria dos algoritmos de alinhamento de proteínas (LESK, 2000).

2.9 Técnicas de Alinhamento Múltiplo de Sequências

Acerca dos alinhamentos globais e locais, que são técnicas de alinhamento, esses algoritmos que usam programação dinâmica têm complexidade na ordem de O($m.n$), onde m é o tamanho da primeira sequência e n da segunda. Já os MSAs, a diferença é que a complexidade aumenta de acordo com o número de sequências (OGATA, 2006).

Em 1994, Wang e Jiang (1994) provaram que o MSA é um problema NP-Completo. Logo, como utiliza-se a programação dinâmica, o problema é resolvido apenas com sequências de tamanho pequeno ou médio. O Algoritmo de Needleman e Wunsch, por exemplo, tem a ordem de $\Omega(k2^k n^k)$ de tempo e ainda $\Omega(n^k)$ de espaço, onde k é o número de sequências e n é o tamanho da maior sequência (SOUZA, 2010).

E como o não existe solução em tempo polinomial para o problema do Alinhamento Múltiplo de Sequências, técnicas que utilizam heurísticas, como o algoritmos genéticos, podem gerar soluções aceitáveis para o problema.

2.9.1 Alinhamentos Progressivos

O alinhamento progressivo é uma das formas mais simples de alinhar sequências. Um dos algoritmos progressivos mais usados para o problema de MSA é o *Clustal Omega*. A estratégia de um alinhamento progressivo é calcular a matriz de distância e construir a árvore filogenética a partir dessas distâncias de pares juntamente com as sequências alinhadas por alinhamento múltiplo (SOUZA, 2010).

Sobre suas características, o desempenho de um alinhamento progressivo é um diferencial, porém, como é um algoritmo guloso, não é possível voltar nem mesmo um passo para retomar uma decisão, assim pode-se perceber se um alinhamento é ruim já no início da execução algoritmo.

2.9.2 Alinhamentos Iterativos

Os alinhamentos iterativos são alinhamentos que ao contrário dos progressivos, eles conseguem melhorar os alinhamentos, ou sub alinhamentos já criados. A cada repetição, o algoritmo iterativo para resolver o problema de alinhamento de sequências tenta melhorar

os alinhamentos de forma determinística ou estocástica. Como espera-se de um algoritmo iterativo, ele refina de forma gradual a solução do problema (ORDINE, 2015).

2.10 Ferramentas de Bioinformática

2.10.1 BLAST

O *Basic Local Alignment Search Tool* (*BLAST*) é uma das ferramentas de análise de sequências biológicas de domínio público mais utilizadas hoje. Ele pode ser executado por uma interface *web*[1] em forma de serviço *web* ou instalado e executado localmente. O servidor do *BLAST* fica no *National Center for Biotechnology Information* (*NCBI*), bem como a maioria de seus diferentes bancos de dados biológicos (MCGINNIS; MADDEN, 2004).

Nos aspectos de execução, os principais tipos de buscas feitos pelo *BLAST* são buscas por nucleotídeos, proteínas, traduções e genomas. Uma vez que as sequências são submetidas, conhecidas como *query*, o *BLAST* funcionará de acordo com o algoritmo selecionado e retornará um relatório com uma tabela de *hits*, sequências com similaridades. Esse relatório pode ter formatos diferentes, como o XML e ASN.

Para exemplificar, as sequências da Figura 2.12 fazem parte de um arquivo no formato *FASTA*, onde '>' é o símbolo de início da descrição de uma sequência. Essa descrição tem um *SEQUENCE_ID*, que e o identificador da sequência de uma espécie e ele deve ser único. A linha seguinte deve ser a sequência com proteínas ou pares de bases (SIEVERS et al., 2011a).

```
 1 >StWOX13 (PGSC0003DMP400002839)
 2 MEWEKQQQQQPPVSAPQQTVEELNGAVSGGMFVKVMTDEQMEVLRKQIAVYATICEQLVD
 3 LHKSMASQHDLAGARLGNLYCDPLVTSAGHKITGRQRWTPTPMQLQILERIFEQGNGTPT
 4 KQKIKDITSELSQHGQISETNVYNWFQNRRARSKRKQQVAATNTTESEVETEVESPNEKK
 5 TKPEDLQSSHMPTSMAEDLGYENPDVSSGMHSLDPRTSKPEPMFPSDGSSKPAASYGQMS
 6 FYGMSNPRMDQLMGKMEVPGSYHPYLHADDYNMTG
 7
 8 >SlWOX13 (Solyc02g082670.2.1)
 9 MDWEKQQQQPPVSAPQQTAEELNGTVSGGMFVKVMTDEQMEVLRKQIAVYATICEQLVDL
10 HKSMASQHDLAGARLGNLYCDPLVTSAGHKITGRQRWTPTPMQLQILERIFEQGNGTPTK
11 QKIKEITSELSQHGQISETNVYNWFQNRRARSKRKQQVAATNNTESEVETEVESPNEKKT
12 KPEDLQSSHMPTSMAEDLGYENPDVSSGMHSLDPRTSKPEPMFPSDGSSKPAASYGQMSF
13 YGMSNPRMDQLMGKMEVPGSYHPYIHADDYNMTG
```

Figura 2.12 – Exemplo de um arquivo no formato fasta.

Existem várias formas de fazer uma busca utilizando o *BLAST*, como já visto, pode-se escolher o se é proteína ou nucleotídeo, por exemplo. Também é possível informar a quantidade de processadores que podem ser utilizados, assim como em qual banco de dados a sequência será comparada.

[1] http://www.ncbi.nlm.nih.gov/BLAST/

Em um relatório do *BLAST*, é possível informar o parâmetro de confiança (*e-value*) que tem a função de filtrar *hits* com resultados menos satisfatórios ou muito distantes do resultado desejado. Ainda, o resultado disponibiliza as posições inicial e final da região *query* que conseguiu *hit* e as posições inicial e final da da região da sequência encontrada no banco de dados biológico.

Existem vários algoritmos que compõem o *BLAST* que podem ser utilizados de diferentes formas, existem várias formas e parâmetros para pesquisas com diferentes focos. O blastn procura sequências de nucleotídeos de alta similaridade e as mostra em ordem de identidade. O tblastn é um algoritmo que converte os aminoácidos traduzidos em nucleotídeos (NCBI, 2013).

Além dos algoritmos citados, existe o blastp que busca a similaridade entre proteínas, além de outras características. Já o blastx faz uma busca a partir de uma *query* de nucleotídeos contra um banco de dados de proteínas e o resultado é a tradução da *query*.

Outro algoritmo existente é o *Position-Specific Iterative Basic Local Alignment Search Tool* (*PSI-BLAST*), ele gera matrizes *PSSMs*, as *pre-calculated position-specific scoring matrices*, que são criadas a partir de resultados de buscas com blastp. As *PSSMs* registram os padrões de conservação dos alinhamentos de proteínas que estejam em um limiar de *e-value* específico.

Além disso, existe o *RPS-BLAST* que faz uma busca contra PSSMs. Esse algoritmo tenta encontrar os *hits* de domínios conservados no banco *Conserved Domain Database* (*CDD*). O *RPS-BLAST* tem um comportamento contrário do *PSI-BLAST*, já que sua função é tentar recuperar domínios conservados a partir das *PSSMs*.

2.10.2 *Clustal Omega*

O *Clustal Omega* é uma ferramenta que gera alinhamentos mais precisos a partir de sequências de tamanhos bem variados. Em testes de *benchmark*, o *Clustal Omega* mostrou-se mais acurado que os métodos mais tradicionais de MSAs (SIEVERS et al., 2011b).

O alinhamento é feito a partir de um arquivo no formato *FASTA*, formato de texto que representa uma sequência de nucleotídeos ou aminoácidos, onde o resultado é um arquivo de extensão ALN com as informações do alinhamento. Também é possível gerar a saída no formato MSF na qual será usada neste trabalho.

A Figura 2.13 é um exemplo de saída no formato MSF de um arquivo de entrada no formato *FASTA* que foi alinhada e as sequências alinhadas são encontradas na Figura 2.12, já mencionada. É possível visualizar no próprio arquivo informações de tamanho de alinhamento, tipo de moléculas alinhadas, os nomes das sequências, juntamente com o alinhamento.

```
 1 !!AA_MULTIPLE_ALIGNMENT 1.0
 2  squid.msf  MSF: 275  Type: P  April 09, 2019 17:17  Check: 2690  ..
 3
 4  Name: StWOX13  Len:    275  Check: 6650  Weight: -1.00
 5  Name: SlWOX13  Len:    275  Check: 6040  Weight: -1.00
 6
 7 //
 8
 9          1                                                    50
10 StWOX13  MEWEKQQQQQ PPVSAPQQTV EELNGAVSGG MFVKVMTDEQ MEVLRKQIAV
11 SlWOX13  MDWEKQQ.QQ PPVSAPQQTA EELNGTVSGG MFVKVMTDEQ MEVLRKQIAV
12
13          51                                                   100
14 StWOX13  YATICEQLVD LHKSMASQHD LAGARLGNLY CDPLVTSAGH KITGRQRWTP
15 SlWOX13  YATICEQLVD LHKSMASQHD LAGARLGNLY CDPLVTSAGH KITGRQRWTP
16
17          101                                                  150
18 StWOX13  TPMQLQILER IFEQGNGTPT KQKIKDITSE LSQHGQISET NVYNWFQNRR
19 SlWOX13  TPMQLQILER IFEQGNGTPT KQKIKEITSE LSQHGQISET NVYNWFQNRR
20
21          151                                                  200
22 StWOX13  ARSKRKQQVA ATNTTESEVE TEVESPNEKK TKPEDLQSSH MPTSMAEDLG
23 SlWOX13  ARSKRKQQVA ATNNTESEVE TEVESPNEKK TKPEDLQSSH MPTSMAEDLG
24
25          201                                                  250
26 StWOX13  YENPDVSSGM HSLDPRTSKP EPMFPSDGSS KPAASYGQMS FYGMSNPRMD
27 SlWOX13  YENPDVSSGM HSLDPRTSKP EPMFPSDGSS KPAASYGQMS FYGMSNPRMD
28
29          251                    275
30 StWOX13  QLMGKMEVPG SYHPYLHADD YNMTG
31 SlWOX13  QLMGKMEVPG SYHPYIHADD YNMTG
```

Figura 2.13 – Exemplo de saída do *Clustal Omega* no fromato .msf

2.10.3 MEGA

O *Molecular Evolutionary Genetics Analysis* (*MEGA*) é uma ferramenta para fazer diversas análises biológicas a partir das sequências biológicas. Essa aplicação faz de alinhamento de sequências, inferência filogenética, verifica padrões de genes. Sua versão atual é o *MEGA X* e funciona em Windows e Linux (KNYAZ et al., 2018).

Outra função bem útil do *MEGA* é o módulo *prototype*, que ajuda bastante a fazer as configurações para posteriormente serem utilizadas em linha de comando. Ainda, em conjunto com as configurações do módulo *prototype* utiliza-se o *MEGA-CC*, a versão do *MEGA* que pote ser executada totalmente por linha de comando, o que dá mais controle para automação de tarefas e análises.

2.10.4 MView

MView é uma ferramenta para converter resultados de uma busca de sequências de banco de dados biológicos em uma apresentação que evidencia a identidade entre as sequências e essa apresentação pode ser em formato de *HTML*. A intenção de exibir os alinhamentos em formato *HTML* é para facilitar ferramentas que têm formato *web* (BROWN; LEROY; SANDER, 1998).

2.11 Considerações Finais

Neste capítulo foram apresentados os conceitos de biologia molecular e bioinformática, tais conceitos são fundamentais para o entendimento deste projeto. Portanto, foram abordados os conceitos de aminoácidos, proteínas e ácidos nucleicos. Ainda, foram introduzidos os conceitos de homologia e filogenia. Por fim, foram mostradas algumas técnicas e ferramentas de bioinformática.

3 METODOLOGIA

3.1 Considerações Iniciais

Este capítulo aborda os métodos utilizados na criação do *GENE FINDER EASY*. Aqui é explicado processo do sistema e também como as ferramentas foram utilizadas em conjunto a fim de solucionar o problema proposto neste projeto.

3.2 O Pipeline

Para Melo (2009), "Pipeline é um modelo de arquitetura no qual ocorre a divisão de um processamento sequencial de etapas". No modelo de *pipeline*, uma sequência direta de etapas compõem o processo geral de um sistema, onde os dados de saída de uma etapa são usados como parâmetros de entrada para a próxima etapa. A Figura 3.1 ilustra o *pipeline* do *GENE FINDER EASY*, com a finalidade de mostrar a sequência exata do processo de execução do sistema.

Ainda, deve-se ter o cuidado em não confundir *pipelines* e workflows. Como já informado, um *pipeline* segue um fluxo direto, onde uma etapa obrigatoriamente é antecede a próxima. Já na arquitetura de *workflow* é possível haver várias etapas que sucedem uma mesma etapa anterior, logo deve-se tomar uma decisão para qual caminho escolher (MELO, 2009).

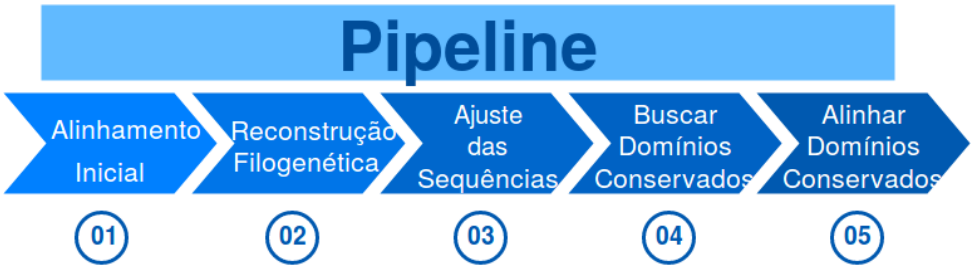

Figura 3.1 – Pipeline

3.3 Alinhamento Inicial

A primeira etapa efetua o alinhamento global de todas as sequências para permitir que a próxima etapa, a reconstrução filogenética, seja executada de forma adequada, já que é necessário que as sequências estejam alinhadas para a reconstrução filogenética.

3.4 Reconstrução Filogenética

Na segunda etapa, a árvore filogenética é reconstruída a partir de um arquivo de alinhamento de sequências. Nessa etapa também é feito o teste de filogenia com objetivo

de certificar a qualidade da árvore filogenética gerada.

3.5 Ajuste das Sequências

Após a reconstrução filogenética, a terceira etapa é ajustar as sequências selecionadas a partir da análise feita pelo pesquisador da árvore guia. Na etapa atual as sequências escolhidas serão destacadas em arquivos no formato *FASTA* para a busca de seus possíveis domínios conservados.

3.6 Buscar Domínios Conservados

Na quarta etapa do processo, é feita a busca dos domínios conservados nas sequências de proteínas. Essas sequências serão comparadas contra um banco de dados biológico de domínios conservados, o CDD. Caso haja sequências com essas mesmas regiões, serão retornados *hits* com as informações do organismo encontrado.

3.7 Alinhar Domínios Conservados

E na última parte do *pipeline*, todas as regiões encontradas nas sequências de entrada serão alinhadas para verificar a possibilidade desses genes serem de uma mesma família. Se os genes encontrados possuírem alta taxa de similaridade, pode-se dizer que esses genes são da mesma família e têm função similar.

3.8 Ferramentas utilizadas

A linguagem utilizada para a implementação da maior parte do *GENE FINDER EASY* foi Python, tanto por ser uma linguagem de fácil utilização quanto pela sua aceitação e extensa utilização em bioinformática. Além disso, também é utilizada a linguagem PHP e o Laravel *Framework*.

O Laravel é *Framework* que utiliza o padrão *Model-View-Controller* MVC para construir aplicações *web*. Com a intenção de implementar uma *API*, utiliza-se o módulo de *API* do Laravel, assim disponibiliza-se um serviço *web* para aplicações mobile e interfaces *web*.

Além disso, foi utilizado o *Quasar Framework*, para gerenciar a interface multiplataforma criada. O *Quasar Framework* é um *framework* multiplataforma que gera distribuições da aplicação *mobile* para as plataformas Android, iOS e também *web*.

Quasar Framework também tem uma grande quantidade de componentes reponsivos e sua linguagem principal de desenvolvimento é o VueJS, um *framework* em javascript baseado em componentes, com funcionalidade similar ao *Angular*.

Outra linguagem usada apenas como uma chamada de *script* para a geração de imagem da árvore filogenética é a linguagem R. E também foi utilizada em forma de *script* a linguagem *Perl* para gerar a tabela com domínios de sequências.

3.9 Considerações Finais

Este capítulo abordou a metologia usada no presente projeto, quais ferramentas e técnicas compõem o *GENE FINDER EASY*. Aqui foi explicado todo o processo do sistema em forma de *pipeline*.

4 GENE FINDER EASY

4.1 Considerações Iniciais

Nesse capítulo é explicada a arquitetura do *GENE FINDER EASY*, a aplicação proposta por esse projeto.

4.2 Arquitetura da Aplicação

A arquitetura da aplicação segue na Figura 4.1 e mostra como é o processo de geração dos alinhamentos múltiplos de sequências das conservações encontradas. Na arquitetura, o sistema se divide em em interface *web* com um cliente que se comunica com a *API* e essa retorna os resultados obtidos.

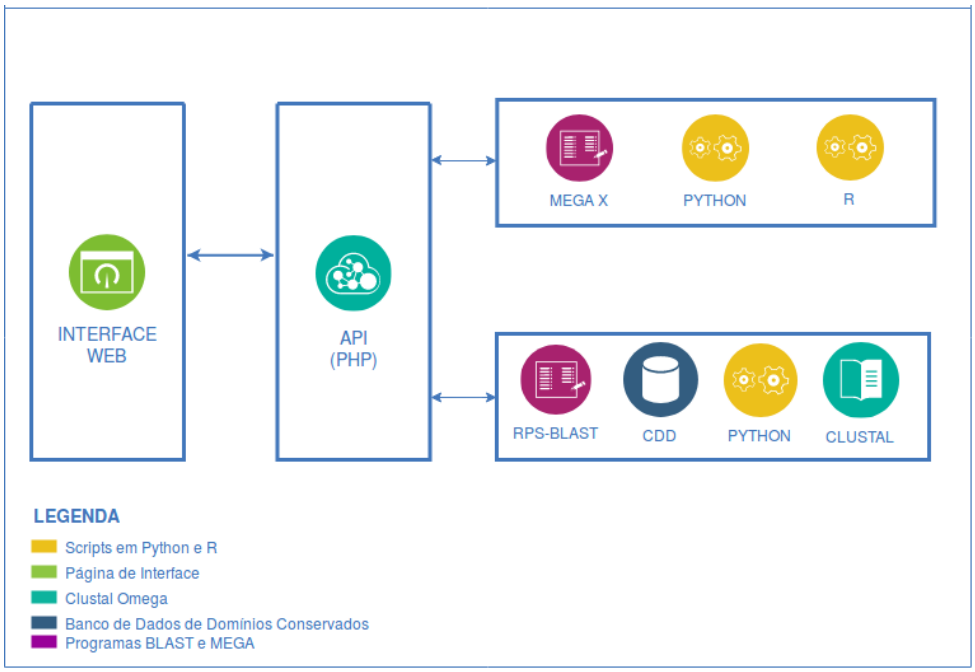

Figura 4.1 – Pipeline do *GENE FINDER EASY*

4.3 MEGA

A aplicação mega-cc é utilizada para gerar o alinhamento inicial das sequências onde a saída é um arquivo de alinhamento no formato MEG, em seguida o mega-cc é novamente executado para reconstruir a árvore filogenética a partir da saída do alinhamento.

Ao executar o mega-cc, se qualquer configuração fora da execução for necessária, é preciso criar um arquivo no módulo *prototype* do *MEGA X*. As configurações de alinhamento foram utilizar o algoritmo *ClustalW* para alinhamentos de proteína, como mostra

a Figura 4.2. Além das opções de alinhamento, também foi necessário criar um arquivo de opções para reconstrução da árvore filogenética das sequências enviadas.

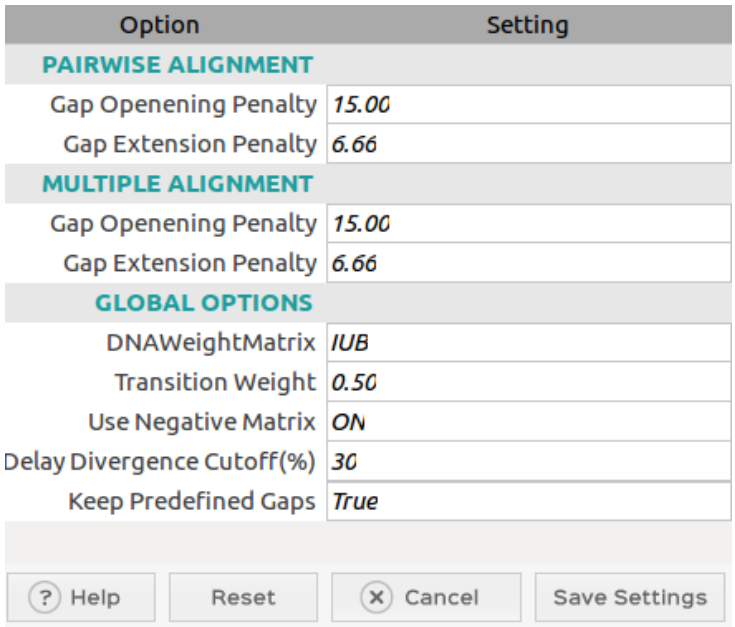

Figura 4.2 – Configurações do MEGA feitas no modo *prototype* MSA

Da mesma forma, no módulo *prototype*, foi selecionada a opção de filogenia e reconstrução e testes de árvore filogenética com o método *Neighbor-Joining*, como mostra a Figura 4.3.

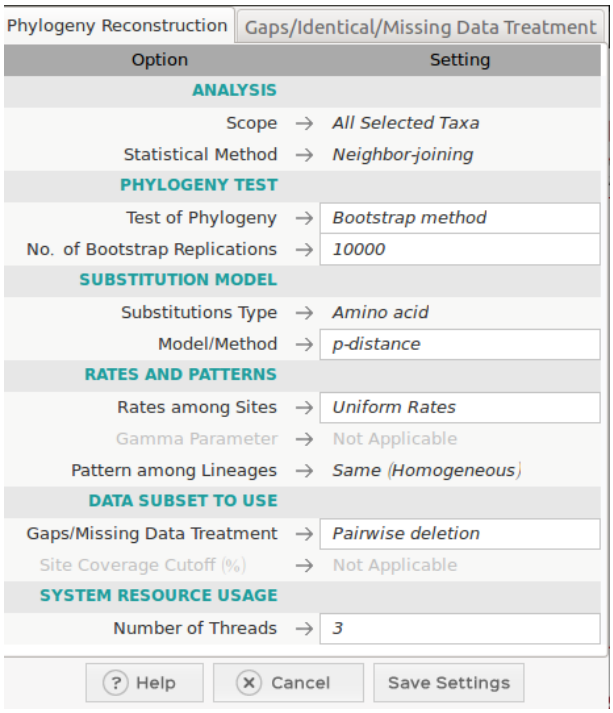

Figura 4.3 – Configurações do MEGA feitas no modo *prototype* para filogenia

Os parâmetros usados para teste de filogenia foram o método foi *Boostrap*, o qual

é conhecido como um valor de confiança que terá o número de 10000 replicações. O método de *Bootstrap* assume que cada aminoácido na sequência pode evoluir de forma independente.

Para calcular a variância e covariância de replicações é costume usar 1000 replicações, pois é o suficiente para os testes em sequências de até tamanho 100 (KUMAR; NEI, 2000). Foi decidido o número de 10000 replicações para testar a robustez das árvores, assim como (MANCINI et al., 2019) e (TAN et al., 2018).

O modelo de substituição foi o *p-distance* aplicado em aminoácidos, essa distância é a proporção de aminoácidos que são diferentes em uma comparação de duas sequências. e para otimizar a execução, o próprio MEGA já adiciona a quantidade aconselhável de *threads* para sistema.

4.4 Linguagem R

Como a saída da execução do mega-cc é um arquivo ou conjunto de arquivos, foi necessário criar um *script* com a linguagem R que utiliza a biblioteca de bioinformática *bioconductor*, especificamente o método "ggtree"para criar a imagem da árvore filogenética a partir da do arquivo de saída no formato NWK do mega-cc.

4.5 BLAST

O BLAST é executado para encontrar as similaridades das proteínas selecionadas no sistema e como o objetivo dessa aplicação é encontrar informações sobre os domínios conservados dos organismos, o tipo de algoritmo utilizado foi rpsblast, que busca os domínios no banco de dados CDD.

Por conseguinte, um *script* em python é executado para encontrar a região de *query* que obteve o melhor *hit* contra o banco CDD. E por fim, um arquivo no formato .fasta composto com as regiões encontradas pelo rpsblast e essa será a nova entrada para o alinhamento do *Clustal*.

4.6 Clustal

O *Clustal Omega* alinha as entradas com as sequências referentes as regiões encontradas no rpsblast e salva o arquivo no formato MSF. Esse arquivo é disponibilizado para *download* e é feita a apresentação do alinhamento na página de resultado da aplicação.

4.7 Bancos de Dados Biológicos

Neste trabalho, o banco de dados biológico utilizado foi o CDD. Com a finalidade de montar o banco de dados, é necessário ter os arquivos encontrados no próprio repositório

FTP do CDD [1]. No processo de montagem do banco de dados são gerados arquivos com o nome do banco de dados e formatos FREQ, LOO, PHR, PIN, PSD, PSI, PSQ, RPS e esses são usados com o algoritmo *RPS-BLAST*.

4.8 Execução da Aplicação

Parte inicial da aplicação é uma interface *web*, onde o usuário pode escolher em colocar as sequências ou o arquivo em formato fasta, como é mostrado na Figura 4.4. Depois, quando o botão "GERAR ÁRVORE" é clicado, o sistema faz uma chamada na *API* para alinhar as sequências e gerar a árvore filogenética das mesmas.

Nota-se que na Figura 4.1, a *API* executa os *scripts* em python e R, juntamente com o MEGA X para retornar o resultado: uma árvore filogenética das sequências e um conjunto de *checkboxes* para o usuário poder escolher quais sequências devem ser alinhadas, de acordo com a análise de similaridade feita.

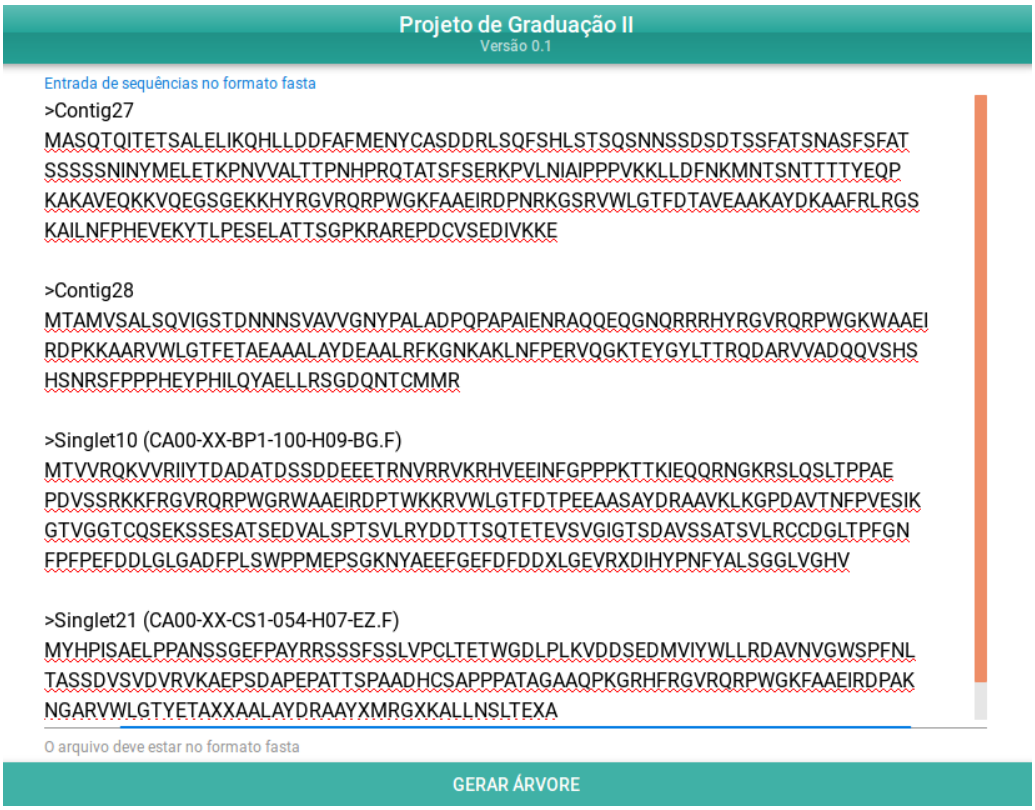

Figura 4.4 – Tela inicial do *GENE FINDER EASY*

Na Figura 4.5 que representa segunda parte da aplicação, quando o botão "ENVIAR SEQUÊNCIAS" é clicado, uma nova chamada na *API* é feita e o sistema executa o rpsblast com a *query* contra o banco CDD a fim de encontrar as regiões com maior similaridade e após o *script* em python criar a entrada com as partes da sequência mais importantes, o novo alinhamento é feito a partir do *Clustal Omega*.

[1] ftp://ftp.ncbi.nih.gov/pub/mmdb/cdd

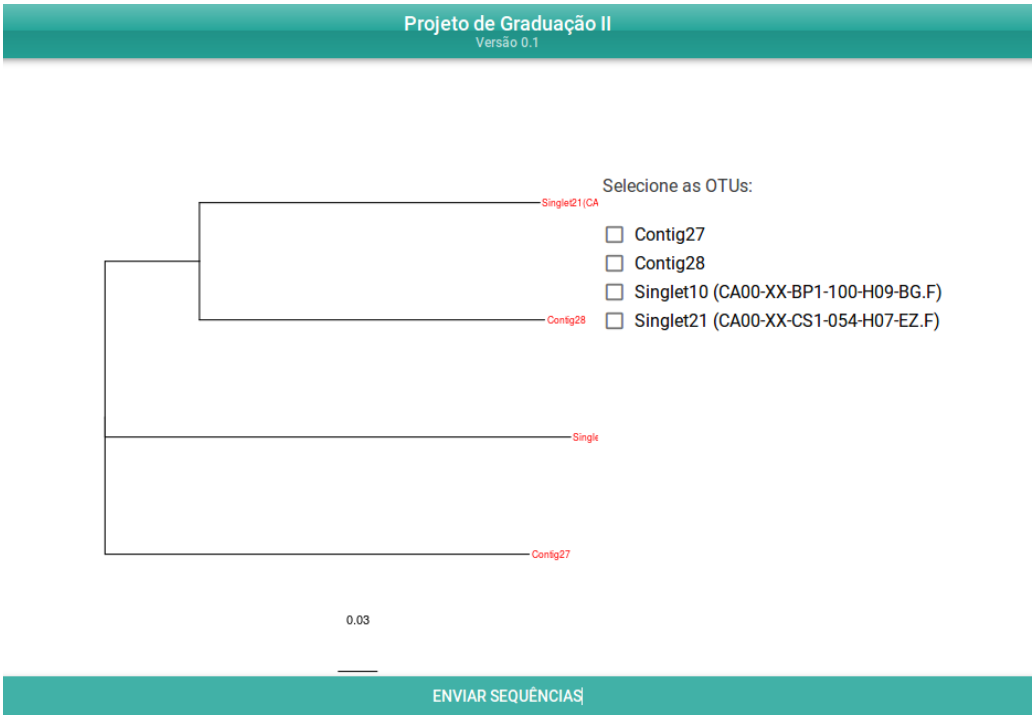

Figura 4.5 – Tela de seleção de sequências

E na última parte da aplicação, a tela de resultados na Figura 4.6 mostra o alinhamento feito só com as regiões mais similares encontradas, a árvore filogenética de todas as sequências enviadas e o arquivo de alinhamento do *Clustal Omega*.

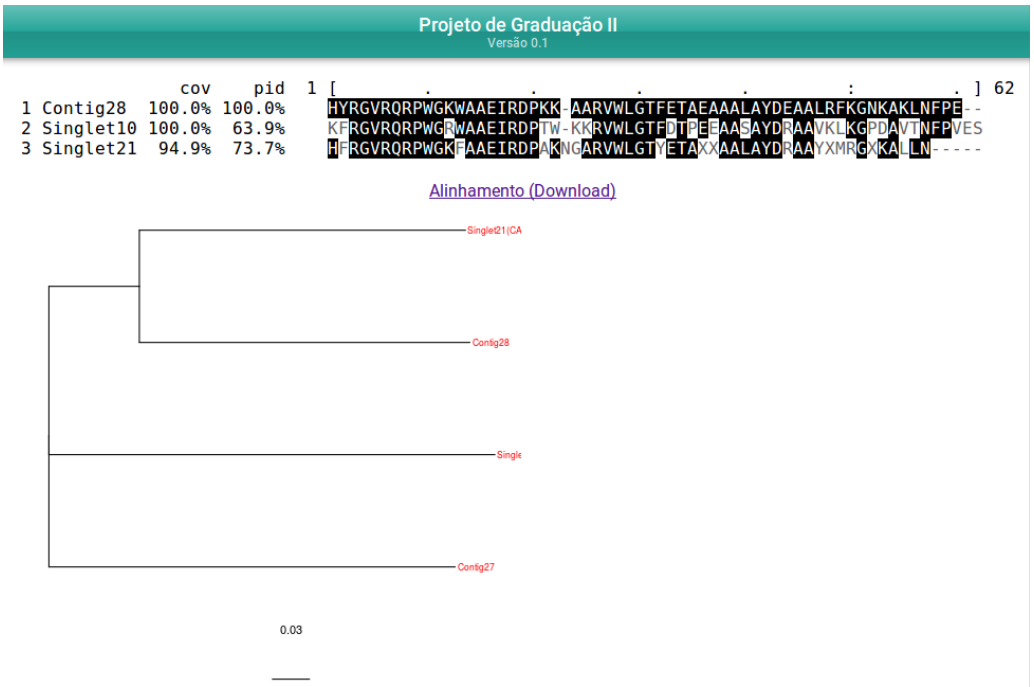

Figura 4.6 – Resultado do *GENE FINDER EASY*

4.9 Considerações Finais

Nesse capítulo abordou como o sistema *GENE FINDER EASY* é executado. Primeiro foi explicado a arquitetura do sistema, em seguida todos os *scripts* e os aplicativos *MEGA X* e *Clustal Omega*, que foram chamados no *GENE FINDER EASY*. Também foram exibidos as saídas dos alinhamentos múltiplos de sequências das conservações encontradas.

5 EXPERIMENTOS E RESULTADOS

5.1 Considerações Iniciais

Este capítulo aborda os testes e resultados feitos durante o desenvolvimento deste trabalho. Acerca dos testes feitos neste projeto, foi utilizada uma máquina para os testes que possui um processador *Intel Core* i5-5200U com 4 *cores* de 2.20GHz, com memória RAM DDR3 de 8 GB. O sistema operacional utilizado foi o Ubuntu versão 16.04 LTS. A interface *web* estava em um servidor *Apache* versão 2.4.18.

5.2 Experimentos

5.2.1 *Coffea Arabica*

O *Coffea Arabica*, Figura 5.1, é uma cultura de café que tem grande valor econômico no brasil, um dos maiores produtores no mundo. Essa cultura de café é suscetível as principais doenças do cafeeiro e estudos como o método de REML/BLUP (*Residual or Restricted Maximum Likelihood/ Best Linear Unbiased Prediction*) tentam encontrar melhores indivíduos ou parâmetros genéticos para melhoramento dessa espécie (SILVA et al., 2018).

Figura 5.1 – Imagem da *Coffea Arabica*. Adaptada de LIMA et al. (2018)

5.2.2 *Arabidopsis thaliana*

A *Arabidopsis thaliana* Figura 5.2 é um dos mais importantes organismos modelo de plantas nos estudos de genética e outras áreas relacionadas a biologia. Em seu genoma

existe uma grande variedade de estruturas que podem ser interpretadas com tecnologias de sequenciamento (TAKOU et al., 2018).

Figura 5.2 – Imagem da *Abidopsis thaliana*. Adaptada de Barrero (2005)

5.2.3 Dados e Parâmetros Selecionados

A fim de fazer uma análise com o *GENE FINDER EASY*, foram testados dois grupos de sequências, o primeiro grupo foram testadas sequências de proteínas do *Arabidopsis thaliana* e foram elas: AtWOX10 (NP_173494.1), AtWOX13 (NP_195280.1), AtWOX14 (NP_173493.2), StWOX13 (PGSC0003DMP400002839), SlWOX13 (Solyc02g082670.2.1).

Já no segundo grupo as sequências de proteínas do *Coffea Arabica* selecionadas foram: CaERF1 (AHA93903.1), CaERF2 (AHA93901.1), CaERF3 (AHA93906.1), CaERF4 (AHA93909.1), CaERF5 (AHA93905.1), CaERF6 (AHA93907.1), CaERF7 (AHA93912.1), CaERF8 (AHA93902.1), CaERF9 (AHA93908.1), CaERF10 (AHA93900.1), CaERF11 (AHA93910.1), CaERF12 (AHA93911.1), CaERF13 (AHA93904.1), CaERF15 (AHA93913.1).

Esses dados foram retirados do estudo de Daúde (2018), o qual analisou a expressão de alguns genes da família *Coffea Arabica*. Em sua análise *in silico*, a busca por *motifs* retornou um *motif* altamente conservado entre o grupo de sequências usado.

5.3 Resultados

O sistema foi executado como é mostrado na Figura 4.1 duas vezes, uma para cada grupo de proteínas. A saída do alinhamento mostra na coluna chamada **cov** a porcentagem de covariância e na coluna chamada **pid** mostra a porcentagem da identidade, assim como é visto na Figura 5.4. O alinhamento também mostra as posições inicial e final de onde ocorreu o alinhamento.

No grupo de sequências 1, a árvore filogenética criada, Figura 5.3 mostrou relacionamento maior entre as sequências AtWOX10 (NP_173494.1) e AtWOX14 (NP_173493.2). Desta forma, essas sequências foram selecionadas para executar um alinhamento com o propósito de encontrar as regiões de domínios conservados.

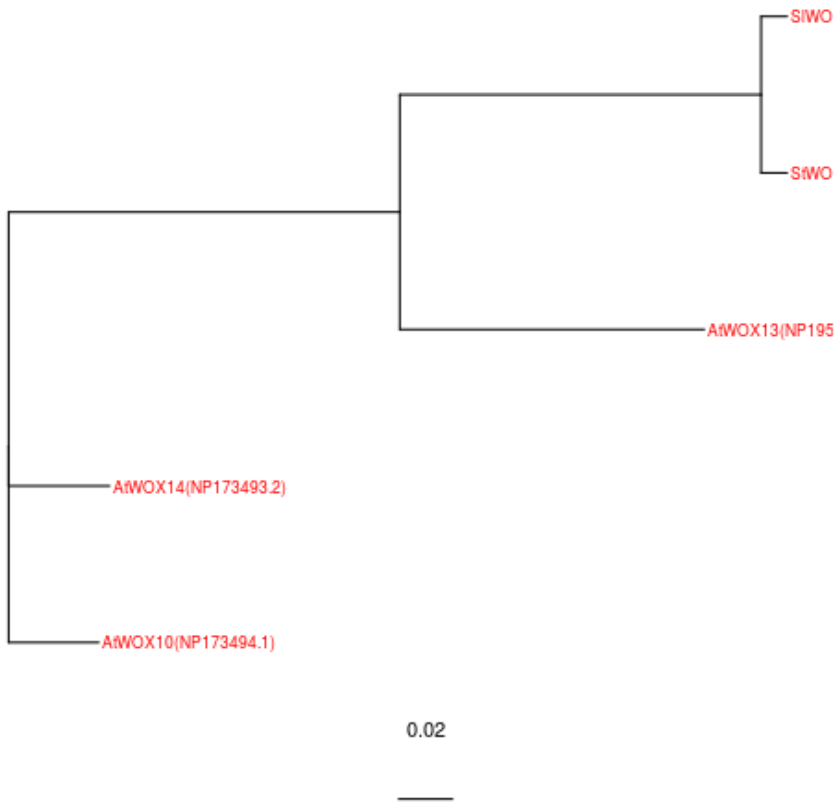

Figura 5.3 – Árvore do grupo 1 gerada pelo megacc

Após a execução do método de alinhamento do *Clustal Omega*, a imagem com MSA, Figura 5.4, apresentou uma taxa de 100% de covariância e 98,3% de identidade entre os domínios AtWOX10 (NP_173494.1) e AtWOX14 (NP_173493.2), com tamanho de 59 resíduos de aminoácidos nas regiões que encontraram melhores *hits* no RPS-BLAST.

```
                cov    pid  1 [        .         .         .         .         :       ] 58
1 StWOX13 100.0% 100.0%    RWTPTPMQLQILERIFEQGNGTPTKQKIKDITSELSQHGQISETNVYNWFQNRRARSK
2 SlWOX13 100.0%  98.3%    RWTPTPMQLQILERIFEQGNGTPTKQKIKEITSELSQHGQISETNVYNWFQNRRARSK
```

Figura 5.4 – Alinhamento gerado pelo *Clustal Omega* do grupo 1

Posteriormente, no grupo de sequências 2, a árvore filogenética criada, Figura 5.5 mostrou relacionamento maior entre as sequências CaERF3 (AHA93906.1) e CaERF5 (AHA93905.1). Então as sequências foram selecionadas e enviadas para efetuar uma busca com o propósito de encontrar as regiões de domínios conservados.

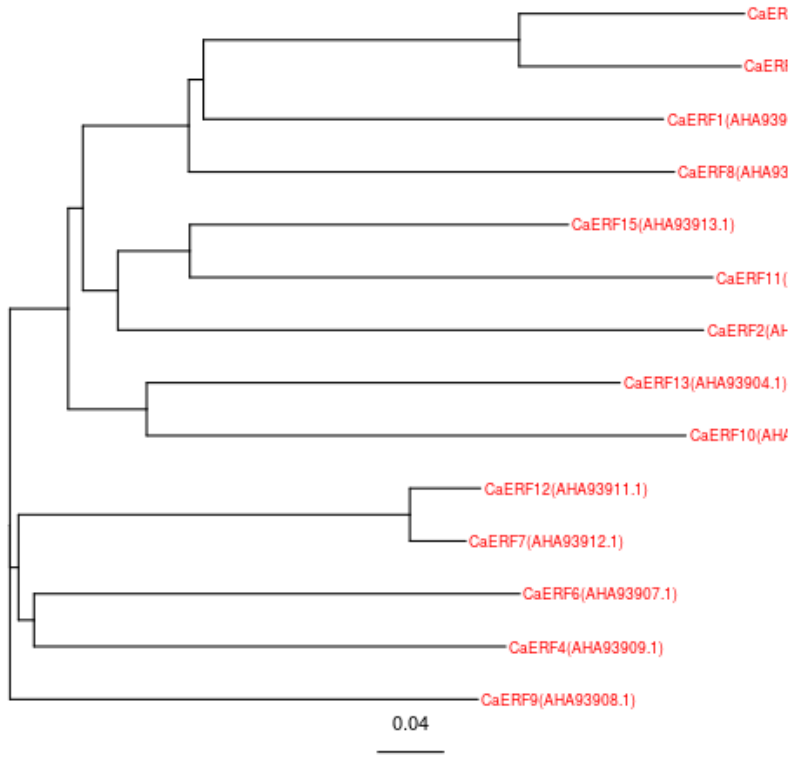

Figura 5.5 – Árvore do grupo 2 gerada pelo megacc

Após a execução do método de alinhamento do *Clustal Omega*, a Figura 5.6, apresentou uma taxa de 100% de covariância e 94,8% de identidade entre os domínios de CaERF3 (AHA93906.1) e CaERF5 (AHA93905.1), com tamanho de 58 resíduos de aminoácidos nas regiões que encontraram melhores *hits* no RPS-BLAST.

```
                cov     pid  1 [      .         .         .         .         :       ] 58
1 CaERF3 100.0% 100.0%    HYRGIRQRPWGKWAAEIRDPRKGVRVWLGTFNTAEAAARAYDTEARRIRGKKAKLNFP
2 CaERF5 100.0%  94.8%    QYRGIRQRPWGKWAAEIRDPRKGVRVWLGTFNTAEEAARAYDTEARRIRGKKAKVNFP
```

Figura 5.6 – Alinhamento gerado pelo *Clustal Omega* do grupo 2

5.4 Considerações Finais

Neste capítulo de experimentos e resultados, foi descrito o ambiente que executou o *GENE FINDER EASY* e também foram mostrados os resultados obtidos com este trabalho. Além disso, foram descritas quais sequências de proteínas foram utilizadas e a que organismos elas pertencem, como a *Arabidopsis thaliana* e *Coffea Arabica*.

6 CONCLUSÃO

Em aplicações e algoritmos de descoberta de *motifs*, leva-se em consideração que é menos preciso procurar estes padrões em sequências inteiras, já que os *motifs* geralmente são pequenos e variam muito de tamanho e padrão. E como parte da solução para esse problema, deve-se alinhar os algoritmos para encontrar buscar o resultado nas sequências alinhadas, pois a região de busca pode ser bem reduzida (MOHAMED; ELLOUMI; THOMPSON, 2016).

Portanto, diante dos resultados obtidos, foi possível apontar que as regiões encontradas pelo *GENE FINDER EASY* têm alta taxa de similaridade. Em ambos os grupos dos genes em *Arabidopsis thaliana* e *Coffea Arabica* , as taxas de identidade foram superiores a 90%.

Nota-se que com essa taxa de similaridade e o tamanho das sequências alinhadas, a região de busca se tornou menor. Agora basta verificar se existem os *motifs* procurado nos domínios conservados.

6.1 Trabalhos Futuros

Tendo em vista o limite de tempo para o término desse projeto, não foi possível testar mais ferramentas para otimizar as gerações de árvores filogenéticas como imagens, bem como as ferramentas de transformar as saídas no formato MSF para o formato *HTML*.

A fim de tornar o sistema mais homogêneo possível, as propostas para trabalhos futuros são:

* Gerar as árvore filogenéticas com a linguagem python, devido a quantidade de bibliotecas de bioinformática disponíveis.
* Desenvolver uma biblioteca em python similar ao *MView* para gerar os alinhamentos em formato *HTML*
* Transformar a aplicação *GENE FINDER EASY* em um sistema distribuído com suporte a *multithreading* em algumas partes que ainda não estão distribuídas.

REFERÊNCIAS

BARRERO, J. **Análisis genético y molecular de mutantes alterados en la síntesis del ácido abscísico en Arabidopsis thaliana.** Tese (Doutorado), 04 2005.

BROWN, N.; LEROY, C.; SANDER, C. Mview: A web compatible database search or multiple alignment viewer. **Bioinformatics**, v. 14, n. 4, p. 380–381, 1998.

CROCHEMORE, M.; HANCART, C.; LECROQ, T. **Algorithms on Strings.** Vuibert, Paris: Cambridge Press, 2007.

DAúDE, M. M. **Análises moleculares de genes pertencentes à subfamília ERF Responsivos ao etileno durante a embriogênese somática em Coffea arabica**. 2018. Monografia (Bacharel em Engenharia de Bioprocessos e Biotecnologia), UFT (Universidade Federal do Tocantins), Gurupi, Brazil.

GRAUR, D.; LI, W.-H. **Fundamentals of molecular evolution**. 2nd. ed. Sunderland, MA: Sinauer, 2000.

JONES, N.; PEVZNER, P. **An Introduction to Bioinformatics Algorithms.** Cambridge, MA: Mol Biol Evol, 2004.

KNYAZ, C. et al. MEGA X: Molecular Evolutionary Genetics Analysis across Computing Platforms. **Molecular Biology and Evolution**, v. 35, n. 6, p. 1547–1549, 05 2018. ISSN 0737-4038. Disponível em: <https://doi.org/10.1093/molbev/msy096>.

KUMAR, S.; NEI, M. **Molecular Evolution and Phylogenetics**. 2nd. ed. 198 Madison Avenue, New York, New York: Oxford University Press, 2000.

LESK, A. M. **Introdução à Bioinformática**. Porto Alegre, RS - Brasil: Artmed, 2000.

LIMA, M. d. F. et al. Produtividade do café arábica (coffea arabica) submetido diferentes doses de lodo de estação de tratamento de esgoto tratado com cal virgem. In: IN: CONGRESSO BRASILEIRO DE RESÍDUOS ORGÂNICOS, 1., 2009, VITÓRIA. ANAIS .... [S.l.], 2018.

LODISH, H. F. et al. **Molecular cell biology.** New York, US: New York, W.H. Freeman, 2008.

MANCINI et al. Molecular characterization of human sapovirus in untreated sewage in italy by amplicon-based sanger and next-generation sequencing. **Journal of applied microbiology**, p. 324–331, 2019.

MARCHLER-BAUER, A. et al. Cdd: a conserved domain database for the functional annotation of proteins. **Nucleic Acids Research**, National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health,Bldg. 38 A, Room 8N805, 8600 Rockville Pike, Bethesda, MD, USA, v. 39, p. D225–D229, 2010.

MCGINNIS, S.; MADDEN, T. Blast: at the core of a powerful and diverseset of sequence analysis tool. **Nucleic Acids Research**, National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health,Bldg. 38 A, Room 8N805, 8600 Rockville Pike, Bethesda, MD, USA, v. 32, p. D225–D229, 2004.

MELO, H. V. F. **Desenvolvimento de um pipeline para análise genômica e transcriptômica com base em Web services**. Dissertação (Mestrado) — Universidade Federal de São Carlos - UFSCar, São Carlos, Brasil, 2009.

MOHAMED, S.; ELLOUMI, M.; THOMPSON, J. Motif discovery in protein sequences. In: RAMAKRISHNAN, S. (Ed.). **Pattern Recognition**. Rijeka: IntechOpen, 2016. cap. 1. Disponível em: <https://doi.org/10.5772/65441>.

NCBI. **The NCBI Handbook**. 2nd. ed. [S.l.]: Bethesda (MD), National Center for Biotechnology Information (US), 2013.

NCBI - Conserved Domain Database (CDD). 2019. Disponível em: <https://www.ncbi.nlm.nih.gov/Structure/cdd/cdd_help.shtml>.

OGATA, A. K. O. **Multialinhamento de sequências biológicas utilizando algoritmos genéticos**. Dissertação (Mestrado) — Instituto de Ciências Matemáticas e de Computação - ICMC-USP, São Paulo, Brasil, 2006.

ORDINE, S. J. R. **Alinhamento múltiplo de proteínas utilizando algoritmos genéticos**. Dissertação (Mestrado) — Instituto de Computação - Universidade Estadual de Campinas, Campinas, Brasil, 2015.

PATCHER, L.; STURMFELS, B. **Algebraic Statistics for Computational Biology**. University of California at Berkeley: Cambridge University Press, 2005.

PEVZNER, P. A. **Computational Molecular Biology: An Algorithmic Approach**. Cambridge, Massachusetts, London - UK: The MIT Press, 2008.

SETUBAL, C.; MEIDANIS, J. **Introduction to Computational Molecular Biology.** Boston - MA: PWS Publishing, 1997.

SIEVERS, F. et al. Fast, scalable generation of high-quality protein multiple sequence alignments using clustal omega. **Molecular Systems Biology**, EMBO and Macmillan, p. 539, 2011.

SIEVERS, F. et al. Fast, scalable generation of high-quality protein multiple sequence alignments using clustal omega. **Molecular systems biology**, UCD Conway Institute of Biomolecular and Biomedical Research, University College Dublin, Belfield, Dublin 4, Ireland, v. 7, p. 539, 2011.

SILVA, L. et al. Indice de seleÇÃo para escolha de progÊnies em acessos de coffea arabica. In: XIV CURSO DE INVERNO DE GENÉTICA – FCAV/UNESP. [S.l.], 2018.

SNUSTAD, D.; SIMMONS, M. **Fundamentos de Genética**. 6th. ed. Rio de Janeiro: Guanabara Koogan, 2013.

SOUZA, M. A. L. de. **Alinhamento múltiplo de proteínas**. Dissertação (Mestrado) — Instituto de Computação - Universidade Estadual de Campinas, Campinas, Brasil, 2010.

TAKOU, M. et al. Linking genes with ecological strategies in Arabidopsis thaliana. **Journal of Experimental Botany**, v. 70, n. 4, p. 1141–1151, 12 2018. ISSN 0022-0957. Disponível em: <https://doi.org/10.1093/jxb/ery447>.

TAN, Z.-D. et al. The complete mitochondrial genome sequence of changbai mountains wild boar (cetartiodactyla: Suidae). **Conservation Genetics Resources**, v. 10, n. 1, p. 99–102, Mar 2018. ISSN 1877-7260. Disponível em: <https://doi.org/10.1007/s12686-017-0773-6>.

TICONA, W. **Algoritmos Evolutivos multi-objetivo para a reconstrução de árvores filogenéticas**. Dissertação (Mestrado) — Tese apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, São Carlos - SP, 2008.

TORTORA, G.; FUNKE, B.; CASE, C. **Microbiologia**. 10th. ed. Porto Alegre: Artmed, 2012.

UNICAMP. **Princípios de Sistemática Molecular**. Campinas - SP, 2004. <ftp://ftp.dca.fee.unicamp.br/pub/docs/vonzuben/bp590_1s04/Criterio_DistanciasII.pdf>, acessado em 05/09/2018.

VERLI, H. **Bioinformática da Biologia à flexibilidade molecular**. São Paulo, SP - Brasil: The MIT Press, 2014.

WANG, L.; JIANG, T. On the complexity of multiple sequence alignment. **Journal of computational biology**, v. 1, p. 337–48, 1994.

ZAHA, A.; FERREIRA, H.; PASSAGLIA, L. **Biologia Molecular Básica**. 5th. ed. Porto Alegre: Artmed, 2014.

ZOU, D. et al. Biological databases for human research. **Genomics Proteomics Bioinformatics**, v. 13, p. 55–63, 2015.