UNIVERSIDADE ESTADUAL PAULISTA
"JÚLIO DE MESQUITA FILHO"
Campus de São José do Rio Preto

Luiz Paulo Liberato

Reconhecimento de padrões em biossequências utilizando sistema imunológico artificial

São José do Rio Preto
2021

Luiz Paulo Liberato

Reconhecimento de padrões em biossequências utilizando sistema imunológico artificial

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação, junto ao Programa de Pós-Graduação em Ciência da Computação, do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista "Júlio de Mesquita Filho", Câmpus de São José do Rio Preto.

Orientador:

Prof. Dr. Geraldo Francisco Donegá Zafalon

São José do Rio Preto
2021

L695r

Liberato, Luiz Paulo
Reconhecimento de padrões em biossequências utilizando sistema imunológico artificial / Luiz Paulo Liberato. -- São José do Rio Preto, 2021
74 p. : il., tabs.

Dissertação (mestrado) - Universidade Estadual Paulista (Unesp), Instituto de Biociências Letras e Ciências Exatas, São José do Rio Preto
Orientador: Geraldo Francisco Donegá Zafalon

1. Reconhecimento de padrões. 2. Alinhamento de sequências (Bioinformática). 3. Algoritmos. I. Título.

Sistema de geração automática de fichas catalográficas da Unesp. Biblioteca do Instituto de Biociências Letras e Ciências Exatas, São José do Rio Preto. Dados fornecidos pelo autor(a).

Essa ficha não pode ser modificada.

Luiz Paulo Liberato

Reconhecimento de padrões em biossequências utilizando sistema imunológico artificial

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação, junto ao Programa de Pós-Graduação em Ciência da Computação, do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista "Júlio de Mesquita Filho", Câmpus de São José do Rio Preto.

Comissão Examinadora

Prof. Dr. Geraldo Francisco Donegá Zafalon
UNESP – Câmpus de São José do Rio Preto
Orientador

Prof. Dr. Carlos Roberto Valêncio
UNESP – Câmpus de São José do Rio Preto

Prof. Dr. Henrique Dezani
FATEC – Câmpus São José do Rio Preto

São José do Rio Preto
8 de setembro de 2021

À Deus e a minha família.

Agradecimentos

Agradeço, primeiramente a meus pais, Francisco Liberato, Marlene Perpétuo Titoto Liberato, ao meu irmão José Francisco Liberato e a minha namorada Julia Alves Sameshima, pelo suporte psicológico durante a minha trajetória acadêmica. Agradeço ao meu orientador Geraldo Francisco Donegá Zafalon pelas instruções e suportes durante a pós-graduação. Ademais, agradeço aos meus amigos Álvaro Magri Nogueira da Cruz e Matheus Lino de Freitas, os quais me alegraram durante as atividades em laboratório. Saliento minha admiração aos professores Adriano Mauro Cansian, Carlos Roberto Valêncio, Geraldo Francisco Donegá Zafalon, Leandro Alves Neves e Rogéria Cristiane Gratão de Souza, os quais forneceram conhecimento científico para que eu obtivesse o necessário para fugir do obscurantismo e do negacionismo, e a reforçar a minha admiração a ciência. Agradeço também a todos os profissionais do IBILCE (Instituto de Biociências, Letras e Ciências Exatas).

Que o conhecimento não se limite a patentes.

(LIBERATO, 2021)

Resumo

Com o avanço nos estudos genômicos foi possível entender melhor a herança genética, a síntese de proteínas e as mutações que ocorrem nos seres vivos. Com o aumento na capacidade de sequenciamento do ADN (Ácido desoxirribonucleico), e seu armazenamento, torna-se possível estudos biológicos avançados. O crescimento dos dados agrega massa de conhecimento para profissionais da área de genética, no entanto, o processamento passa a ser dispendioso quando utilizados métodos determinísticos. Para garantir tempo hábil e maior precisão no processo de reconhecimento de padrões utiliza-se de métodos heurísticos, dado que métodos determinísticos inviabilizam a execução de grandes volumes de dados. Métodos heurísticos possuem a característica de buscar a melhor solução possível dentro do espaço de busca que é explorado. Dentre as heurísticas conhecidas tem-se o Sistema Imunológico Artificial (SIA) que se enquadra na categoria de métodos bioinspirados que simulam um comportamento biológico. No presente trabalho desenvolveu-se a implementação do CLONALG (Algoritmo de Seleção Clonal) da abordagem do SIA com o MMO (Modelo de Markov Oculto) como função de afinidade, afim de obter padrões estocásticos que representem informações genéticas com relevância biológica e um tempo computacional aceitável. Como resultado foi obtido um valor 50% mais relevante em termos de tempo de execução, quando comparado ao CLONALG com a função de afinidade de Hamming. Por se tratar de uma abordagem estocástica é possível armazenar os padrões com maior afinidade para processamentos futuros, e ajustes nos parâmetros do algoritmo podem ser feitos para melhorar ainda mais a qualidade dos padrões encontrados. Finalmente, também validou-se que o CLONALG com a implementação MMO foi capaz de reconhecer os mesmos padrões quando comparado a ferramentas similares.

Palavras-chave: Bioinformática, Sistema Imunológico Artificial, Reconhecimento de Padrões.

Abstract

With the advance on genomics studies was possible to know better the genetic inheritance, protein synthesis and mutations that occurs in living beings. With the increase in the DNA sequencing capacity (Deoxyribonucleic acid), and its storage, advanced biological studies are possible. The growth of data adds mass of knowledge for professionals in the field of genetics, however, processing becomes expensive when using deterministic methods. To ensure timely and greater precision in the pattern recognition process, heuristic methods are used, since deterministic methods make it impossible to execute large volumes of data. Heuristic methods have the characteristic of seeking the best possible solution within the search space that is explored. Among the known heuristics is the Artificial Immune System (AIS), which falls under the category of bioinspired methods that simulate biological behavior. In this work, the CLONALG (Clonal Selection Algorithm) of the AIS approach was implemented with HMM (Hidden Markov Model) as an affinity function, in order to obtain stochastic patterns with biological relevance and an acceptable computational time. As a result, a 50% more relevant value was obtained in terms of execution time, when compared to CLONALG with the Hamming affinity function. As this is a stochastic approach, it is possible to store the patterns with greater affinity for future processing, adjustments in the algorithm parameters can be made to further improve the quality of the patterns found. Finally, it was also validated that CLONALG with the HMM implementation was able to recognize the same patterns when compared to similar tools.

Keywords: Bioinformatics, Artificial Immune System, Pattern Recognition.

Lista de Ilustrações

Lista de Tabelas

Lista de Abreviações

ADN Ácido Desoxirribonucleico

ARN Ácido Ribonucleico

CLONALG Algoritmo de Seleção Clonal

mARN Ácido Ribonucleico Mensageiro

MMO Modelos de Markov Ocultos

NCBI Centro Nacional de Informação Biotecnológica

OG Ontologia Genética

PCSS Pyramidal-based common subsequences search

RNA Redes Neurais Artificiais

SCPS Sliding columns-based pattern search

SIA Sistema Imunológico Artificial

tARN Ácido Ribonucleico Transportador

Sumário

Capítulo 1

Introdução

1.1 Considerações Iniciais

A Bioinformática pode ser definida como o estudo da biologia molecular aliado a estratégias computacionais de alinhamento de sequências e reconhecimento de padrões. As informações são baseadas no dogma central da biologia molecular: as sequências de ADN (Ácido Desoxirribonucleico) são transcritas em sequências de mARN (Ácido Ribonucleico Mensageiro), as sequências de mARN são traduzidas em sequências de proteínas. As aplicações de Bioinformática, portanto, podem abordar a transferência de informações em qualquer estágio do dogma central, incluindo a organização e controle de genes na sequência do ADN, a identificação de unidades transcricionais no ADN, a previsão da estrutura da proteína a partir da sequência e a análise da função molecular (ALTMAN; DUGAN, 2003).

Com a descoberta da estrutura básica do ADN (WATSON; CRICK et al., 1953) e com avanços tecnológicos, a quantidade de dados biológicos disponíveis passou a crescer consideravelmente, de modo que se tornou infactível a análise manual dessa quantidade de informações (ZAFALON et al., 2015). Com o surgimento dos sequenciadores automáticos de ADN, houve um aumento significativo na quantidade de sequências a serem armazenadas. Além do armazenamento é necessário a análise desses dados,

o que torna indispensável a utilização de estratégias computacionais eficientes para a interpretação dos resultados obtidos (PROSDOCIMI et al., 2002). Portanto, fez-se necessária a busca por estratégias computacionais que auxiliassem os trabalhos de análise e tomada de decisão por parte dos biólogos, para que dessa forma posteriores inferências possam ser feitas de maneira satisfatória. Estes fatores foram pontos importantes que originaram a Bioinformática (KHURI, 2008).

A Bioinformática é a junção de várias linhas de conhecimento: engenharia de software, matemática, estatística, ciência da computação e biologia molecular. Uma das vertentes de pesquisa é o reconhecimento de padrões que será explorado nesse trabalho, com exemplos de aplicações e otimizações de métodos.

1.2 Motivação e Justificativa

Encontrar padrões em biossequências é uma tarefa desafiadora. Ao longo dos anos, inúmeros pesquisadores têm dedicado esforços para tornar esse objetivo alcançável, tanto do ponto de vista computacional quanto do ponto de vista biológico (KUCHEROV, 2019). A estratégia escolhida deve combinar ambos aspectos, pois dessa forma é possível obter um padrão com relevância biológica em um custo computacional aceitável.

A busca por padrões em ADN, ARN e proteínas consomem recursos computacionais, e em alguns casos o tempo de execução torna inviável encontrar a solução ótima. Para atenuar tais problemas várias heurísticas foram criadas e estendidas, porém para determinadas situações são necessárias combinações de várias estratégias, para executar a tarefa em um tempo aceitável e obter um valor ótimo ou próximo ao valor ótimo. Dessa forma, a solução proposta no presente trabalho equadra-se nesta necessidade.

1.3 Objetivos

O presente trabalho tem por objetivo a hibridização de métodos para o reconhecimento de padrões em Bioinformática. Assim, no presente trabalho alia-se dois métodos, o CLONALG (Algoritmo de seleção clonal) da classe de algoritmos do SIA (Sistema Imunológico Artificial) e MMO (Modelos de Markov Ocultos). A junção de ambos tem por objetivo oferecer variabilidade e qualidade biológica. A sua hibridização contribui para o contexto de reconhecimento de padrões e serve de inspiração para várias outras combinações de abordagens.

1.4 Organização do Trabalho

A divisão do presente trabalho foi realizada da seguinte forma: no Capítulo 2 é realizada uma revisão bibliográfica referente aos assuntos pertinentes ao escopo da dissertação e alguns trabalhos correlatos, os quais utilizam abordagens semelhantes a proposta deste, no Capítulo 3 é explicada a metodologia utilizada na elaboração do algoritmo proposto, no Capítulo 4 são exibidos os testes executados e o conjunto de dados utilizado. Por fim, no Capítulo 5, apresentam-se as conclusões e os trabalhos futuros.

Capítulo 2

Fundamentação Teórica

Nesta seção são introduzidos conceitos fundamentais da biologia molecular, que são necessários para entendimento de alguns componentes da célula, moléculas pequenas e macromoléculas de extrema importância para os seres vivos.

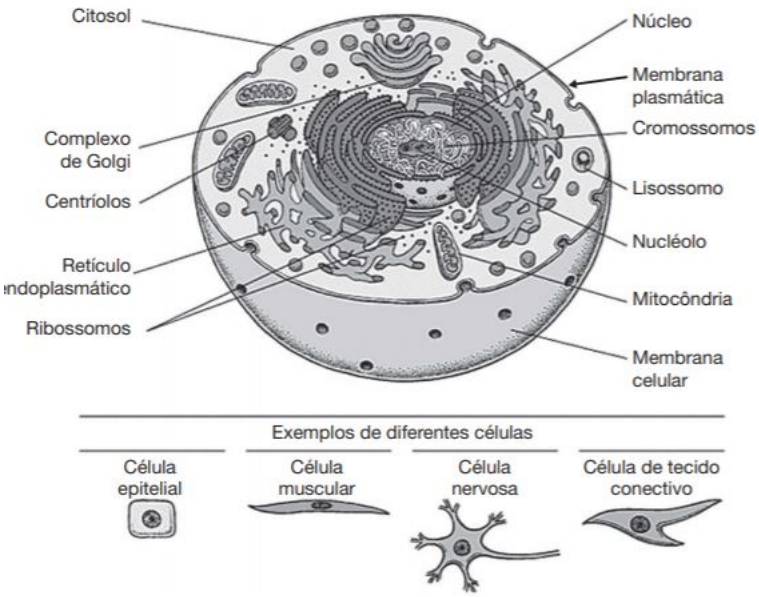
2.1 Organização Celular

As células são pequenas unidades que carregam informações importantes para os seres vivos, composta por membranas, diversos produtos químicos e com habilidades para geração de cópias de si mesmas. Cada espécie existente na terra se reproduz gerando descendentes com características de seus ancestrais, por meio do fenômeno da *hereditariedade*. As informações hereditárias de uma célula são armazenadas na forma de moléculas de ADN de fita dupla - longas cadeias de nucleotídeos representados pelo alfabeto de quatro letras - A (adenina), T (timina), C (citosina) e G (Guanina) - que estão ligadas umas as outras formando uma longa cadeia que codifica a informação genética (ALBERTS et al., 2010).

Exitem dois grupos de organismos vivos baseados na estrutura celular: os **eucariotos** e os **procariotos** (Figuras 2.1 e 2.2). Os eucariotos possuem envoltório nuclear, ou seja, mantém seu ADN em um compartimento intracelular. Os procariotos não

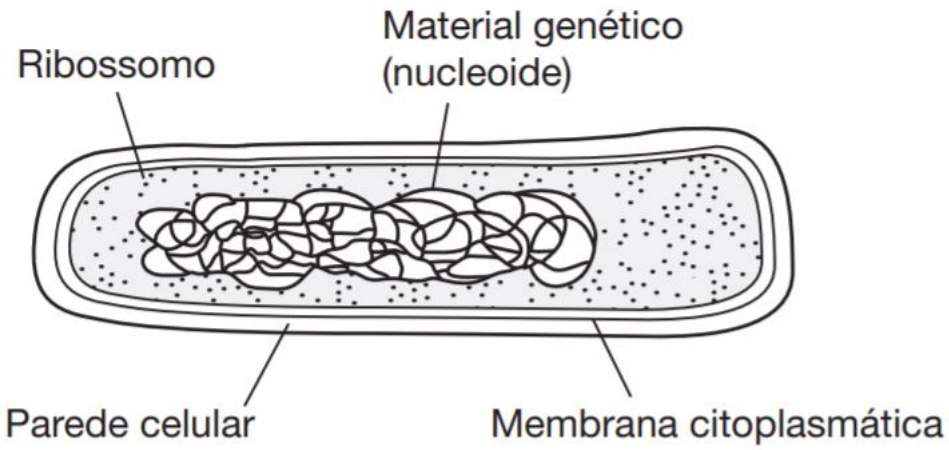
possuem envoltório nuclear para abrigar seu ADN (ALBERTS et al., 2010).

Figura 2.1: Representação esquemática da organização de uma célula de eucarioto.



Fonte: ZAHA; FERREIRA; PASSAGLIA (2014).

Figura 2.2: Representação esquemática da organização de uma célula de procarioto.



Fonte: ZAHA; FERREIRA; PASSAGLIA (2014).

É evidente a importância do ADN na vida do indivíduo, pois é a partir dele que as características são criadas. Além disso, o ADN diz muito sobre antepassados, assim

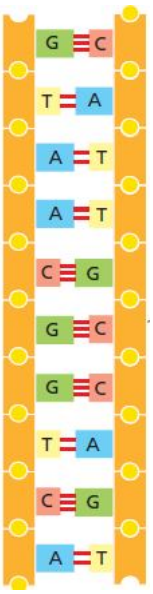
como mutações que podem ocorrer ao longo do tempo, aumentando a variabilidade genética (ALBERTS et al., 2010). Com intuito de compreender melhor tais aspectos, a ciência tem evoluído muito e dedicado esforços para criar tecnologias que facilitem as interpretações sobre o conteúdo biológico. A subseção 2.1.1 aborda conceitos relevantes para a compreensão do ADN e os processos realizados para sintetizar uma proteína.

2.1.1 ADN, ARN e Proteínas

O ADN carrega as informações genéticas do indivíduo, esforços são destinados a entender melhor essa estrutura. Com a extração de informações de uma sequência de ADN é possível realizar descobertas relevantes para humanidade. No início, muitos acreditavam que a estrutura do ADN seria algo complexo e diferente entre si, porém com avanços nas pesquisas, constatou-se que a diferença de um gene para outro é a quantidade e a ordem que são organizadas as suas bases. A forma do ADN é uma dupla hélice composta por duas cadeias polinucleotídicas, que são formadas por bases classificadas em dois tipos, **purinas** e **pirimidinas**. As purinas são **adenina** e a **guanina**, e as pirimidinas são a **citosina** e a **timina** (WATSON et al., 2015).

O ARN possui algumas coisas em comum com o ADN, porém é possível observar características bem distintas entre eles (WATSON et al., 2015). No ARN, a cadeia principal é formada por um açúcar ligeiramente diferente do açúcar do ADN – a ribose em vez da desoxirribose –, e uma das quatro bases é ligeiramente diferente – a uracila (U) no lugar da timina (T). Mas as outras três bases – A, C e G – são as mesmas, e todas as quatro bases pareiam com suas contrapartes complementares no ADN – os A, U, C e G do ARN com os T, A, G e C do ADN (ALBERTS et al., 2010). Na Figura 2.3 é representada a fita dupla de ADN.

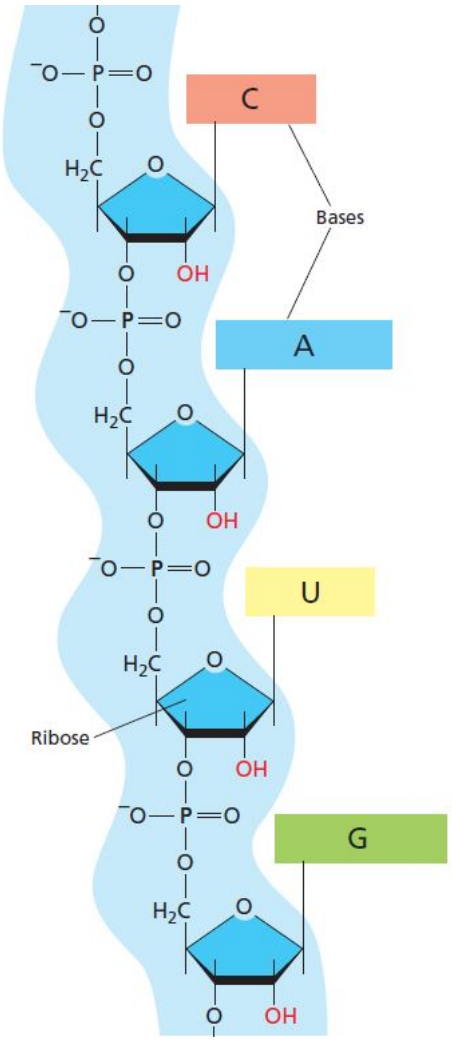
Figura 2.3: Fita dupla de ADN.



Fonte: adaptado de ALBERTS et al. (2010).

Para gerar uma proteína é necessário encontrar a sequência de nucleotídeos específicos para ela. Quando a sequência é encontrada o ARN copia a esta por meio do processo chamado *transcrição* (Figura 2.4). Essa cópia de ARN de segmentos de ADN é utilizada para a síntese da proteína no processo chamado *tradução* (ALBERTS et al., 2010). O processo completo de transcrição e tradução está representado na Figura 2.5.

Figura 2.4: Fragmento de ARN.

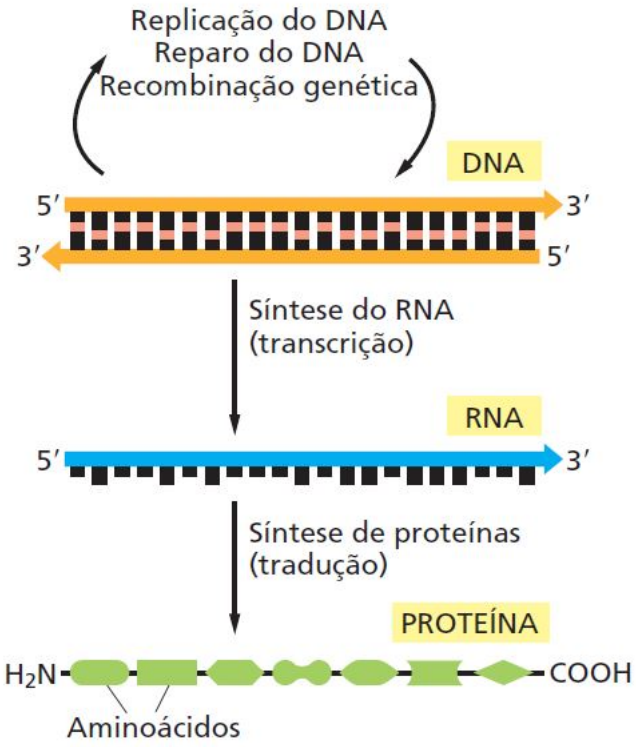


Fonte: adaptado de ALBERTS et al. (2010).

Na síntese de proteínas (ou tradução, Figura 2.5), ocorre interações entre os **códons** (sequência de três nucleotídeos que corresponde a um aminoácido) das moléculas do ARN mensageiro (mARN) com os **anticódons** (sequência de três nucleotídeos presente no tARN) dos ARNs transportadores (tARN) por meio do pareamento de bases, gerando por meio do tARN a sequência de aminoácidos da proteína nascente. Para que esse processo ocorra de fato são necessárias além das interações do mARN-tARN, a presença dos ribossomos que são considerados como “pequenas fábricas”, responsáveis por toda atividade necessária para a síntese de proteínas (ZAHA; FERREIRA;

PASSAGLIA, 2014).

Figura 2.5: Do ADN à proteína.

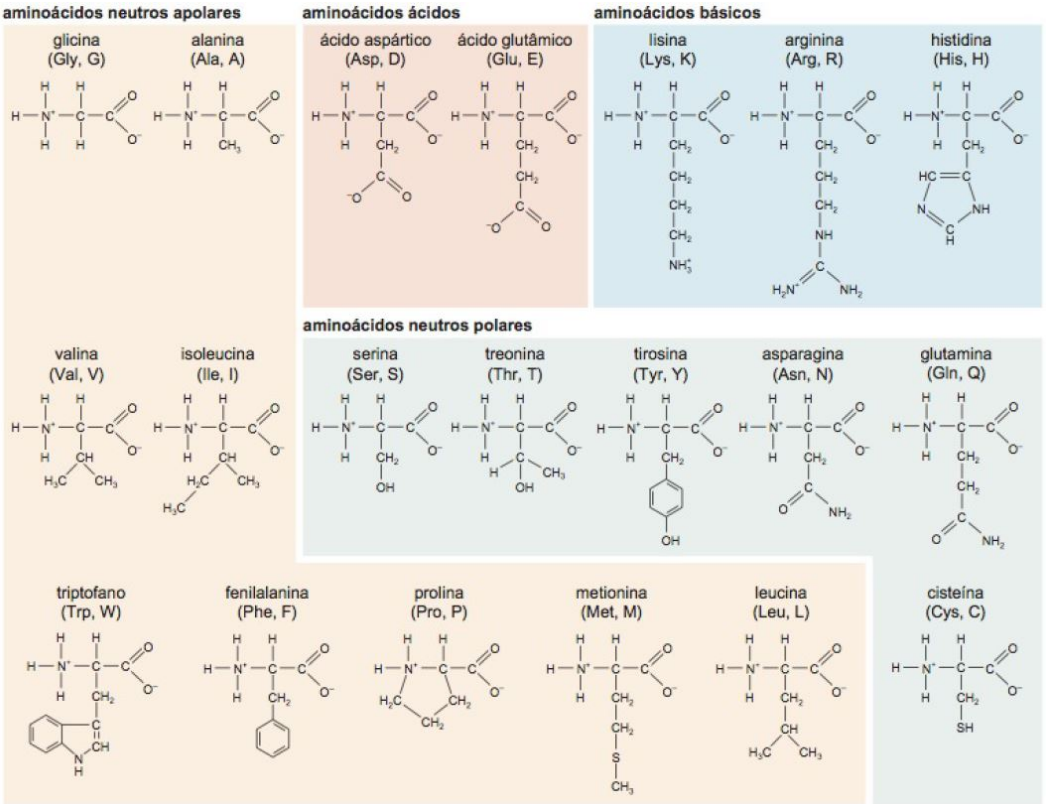


Fonte: adaptado de ALBERTS et al. (2010).

Proteínas são cadeias lineares de aminoácidos, ou seja, um arranjo de α-aminoácidos dos quais 20 ocorrem de maneira regular nos organismos vivos e são especificados pelo código genético (Figura 2.6). Entretanto, esses aminoácidos podem sofrer alterações mesmo após a integração à uma proteína, com isso aumenta-se a variabilidade real (WATSON et al., 2015).

Os aminoácidos realizam ligações peptídicas que são ligações covalentes, formadas por uma reação de condensação com a liberação de uma molécula de água. Cada aminoácido pode fazer duas ligações desse tipo, de modo que sucessivas ligações do mesmo tipo podem formar uma cadeia polipeptídica. Devido ao fato da ligação peptídica liberar uma molécula de água, os componentes da cadeia são conhecidos como **resíduos de aminoácidos** ou, às vezes, apenas resíduos.

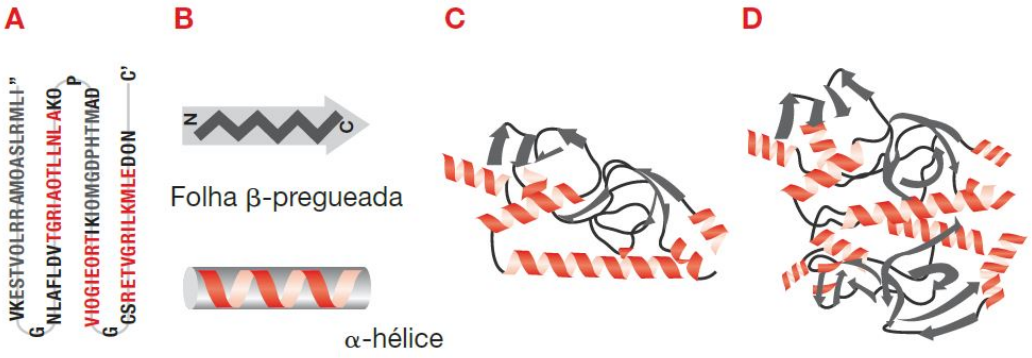
Figura 2.6: Os 20 diferentes aminoácidos encontrados nas proteínas.



Fonte: WATSON et al. (2015).

As proteínas possuem quatro níveis estruturais bem definidos (Figura 2.7). O primeiro nível, a **estrutura primária**, é uma fita unidimensional (1D) onde são arranjados os aminoácidos em uma cadeia polipeptídica, a **estrutura secundária** possui conformação local da sua cadeia polipeptídica e arranjo 3D de um trecho curto de resíduos de aminoácidos, a **estrutura terciária** possui montagem tridimensional enovelada, e por fim a **estrutura quaternária** onde há associação de cadeias polipeptídicas enoveladas em um arranjo de múltiplas subunidades. (WATSON et al., 2015).

Figura 2.7: Estrutura tridimensional de uma proteína com os diferentes níveis organizacionais. (A) estrutura primária; (B) estruturas secundárias;(C) estrutura terciária; (D) estrutura quaternária.



Fonte: ZAHA; FERREIRA; PASSAGLIA (2014).

- A **estrutura primária** é a sequência de aminoácidos que estão ligados para formar uma cadeia peptídica (Figura 2.7A). A ordem dos aminoácidos na cadeia é muito importante, pois a alteração de um aminácido pode provocar doenças como por exemplo, a anemia falciforme. Nessa doença ocorre a substituição de um único aminoácido na molécula da hemoglobina (ZAHA; FERREIRA; PASSAGLIA, 2014).

- A **estrutura secundária** (Figura 2.7B) se refere a um conjunto de arranjos espaciais de aminoácidos próximos na cadeia peptídica central, que provocam dobramentos. Os dobramentos são denominados estruturas secundárias e os arranjos podem apresentar uma organização que se repete em intervalos regulares (ZAHA; FERREIRA; PASSAGLIA, 2014).

- Segundo ZAHA; FERREIRA; PASSAGLIA (2014) a **estrutura terciária** (Figura 2.7C) refere-se à forma como a cadeia polipeptídica está enovelada, incluindo o arranjo tridimensional de todos os átomos da molécula. Este nível estrutural é estabelecido quando diferentes estruturas secundárias se dispõem entre si, a estabilidade da estrutura é mantida por pontes de hidrogênio entre

grupos peptídicos não envolvidos na estrutura secundária.

- A **estrutura quaternária** (Figura 2.7D) é encontrada em proteínas que possuem mais de um polipeptídeo, formando subunidades chamadas de proteínas multiméricas. Essa estrutura refere-se à disposição das subunidades proteicas que formam a molécula. O número de subunidades pode variar e a união entre elas ocorre de forma não covalente, por meio de interações eletrostáticas (ZAHA; FERREIRA; PASSAGLIA, 2014).

A seção 2.2 trata de aspectos ligados a computação que são importantes para compreensão da extração de padrões de proteínas, e diferentes tipos de abordagens utilizadas na literatura.

2.2 Reconhecimento de Padrões

Segundo PAOLANTI; FRONTONI (2020) o reconhecimento de padrões é um processo de classificação que tem por objetivo extrair padrões de um conjunto de dados e categorizá-los em diferentes classes. Pode contribuir para o desenvolvimento de sistemas capazes de resolver determinados problemas como, agrupamento, classificação, combinação ou seleção de características para obtenção de uma representação mais clara de um determinado problema (RIDDER; RIDDER; REINDERS, 2013). Uma classe é caracterizada por um conjunto de padrões. Um padrão normalmente, consiste em um conjunto de características que são essenciais para o reconhecimento exclusivo do padrão. Por exemplo, algumas das medidas podem ser utilizadas para o reconhecimento de padrões: comprimento, largura, altura, média, desvio padrão, entre outras (PAL; RAY; GANIVADA, 2017).

Um sistema de reconhecimento de padrão consiste em três fases: aquisição de dados, seleção e classificação. A fase de aquisição de dados inclui a coleta de dados por meio de um conjunto de sensores, dependendo do ambiente em que os objetos devem

ser classificados (MAJI; PAUL, 2016). Em seguida, os dados são passados para a fase de seleção de características onde a dimensionalidade dos dados é reduzida. Na fase de classificação os dados selecionados são passados para o sistema de classificação em que as informações são avaliadas e uma decisão final é tomada em relação a sua classe (PAL; RAY; GANIVADA, 2017).

2.2.1 Aquisição de Dados

O reconhecimento de padrões pode ser aplicado a diversos tipos de domínios, nos quais os dados podem ser representados quantitativamente, qualitativamente ou ambos; eles podem ser numéricos, linguísticos, pictóricos ou qualquer combinação dos mesmos. Geralmente, a estrutura de dados utilizada no processo de reconhecimento de padrões são de dois tipos: vetores de dados de objetos e dados relacionais (PAL; RAY; GANIVADA, 2017).

2.2.2 Seleção de Características

O objetivo principal dessa etapa é gerar características ótimas necessárias para o reconhecimento e reduzir a dimensionalidade do espaço de medição, para que algoritmos efetivos e facilmente computáveis possam ser criados para uma classificação eficiente. Existem dois problemas inerentes ao processo de seleção: o primeiro é a formulação de um critério adequado para avaliar a qualidade de um conjunto de recursos e o segundo é pesquisar o conjunto ideal em termos de critério. Uma boa característica é imutável com qualquer outra variação possível dentro de uma classe, enfatiza diferenças importantes na discriminação entre padrões de tipos diferentes (PAL; RAY; GANIVADA, 2017).

2.2.3 Classificação

Segundo PAL; RAY; GANIVADA (2017) o problema da classificação é basicamente o de dividir o espaço de busca em regiões, uma região para cada categoria de entrada. O objetivo é tentar atribuir todos os pontos de dados em todo o espaço de busca a uma das possíveis classes. Geralmente, o que se tem disponível são amostras que oferecem informações parciais para a construção do seletor de recursos ou do sistema de classificação. Supõe-se que tais amostras são representantes das classes, esse conjunto de amostras são chamados de conjunto de treinamento. Sendo assim, são fornecidos diferentes valores de parâmetros para o sistema de reconhecimento de padrões. O design de um esquema de classificação pode ser feito com dados rotulados ou não rotulados. Dados rotulados são aqueles obtidos através de objetos com classificações conhecidas; caso contrário, é chamado de aprendizado não supervisionado. O aprendizado supervisionado é usado para classificar objetos diferentes, enquanto o agrupamento é realizado através do aprendizado não supervisionado.

A classificação de padrões, admite muitas abordagens, às vezes complementares, às vezes concorrentes, para fornecer solução à um determinado problema. Isso inclui abordagem teórica da decisão (determinística e probabilística), abordagem conexionista, abordagem teórica de conjuntos difusos e aproximados e abordagem híbrida ou computacional (PAL; RAY; GANIVADA, 2017).

ABDO; GOLDING (2007) aplica a abordagem teórica da decisão para avaliação estatística, para associar sequências de indivíduos recém-sequenciados ou amostrados a grupos pré-identificados.

2.2.4 Agrupamento

Agrupamento é um método de classificação não supervisionado que não necessita de informações de classes dos padrões para determinar os grupos (PAL; RAY; GANIVADA, 2017). Os métodos de separação em grupos são classificados da seguinte

maneira:

- **Métodos de particionamento** consiste em particionar dados de acordo com a quantidade de objetos e o número de grupos a serem formados. Os grupos são criados para otimizar um critério de particionamento, que mede a similaridade, como distância entre os objetos, para realçar a diferença entre objetos de grupos distintos. Inicialmente, é criada uma partição que ao longo das iterações é melhorada com uma técnica de refinamento iterativo, que visa mover objetos de um grupo para outro de acordo com as suas características. Os métodos de particionamento mais conhecidos e comumente usados são k-means, k-medoids e suas variações (PAL; RAY; GANIVADA, 2017). ROZAS et al. (2017) utilizaram métodos de particionamento para análises genéticas populacionais exaustivas em alinhamentos de múltiplas sequências;

- **Métodos de agrupamento hierárquico** constituem um dendograma que pode utilizar qualquer uma das métricas de distância entre ligação única, ligação completa e ligação média. A distância entre dois grupos é obtida pelo caminho mais curto entre dois padrões em dois grupos diferentes (ligação única), ou pelo valor do caminho mais longo entre dois padrões em grupos diferentes (ligação completa) ou pela média das distâncias entre todos objetos no primeiro grupo e todos os objetos no segundo grupo (ligação média). Existem dois tipos de agrupamento hierárquico: (i) Aglomerativo (ii) Divisivo. O primeiro utiliza a abordagem de baixo para cima, considera todos os padrões como um único grupo e mescla um par de grupos em um grupo, de acordo com a menor distância entre o par. O processo de mesclagem dos grupos é finalizada quando todos os grupos são combinados em um único grupo. Utiliza-se no segundo método a abordagem de cima para baixo, que começa com todos os padrões em um grupo e divide o grupo único em dois grupos, de forma que um grupo seja diferente do outro, e assim sucessivamente até que um padrão forme um

único grupo (PAL; RAY; GANIVADA, 2017). Um exemplo de aplicação do método de agrupamento hierárquico pode ser encontrado em (LANGFELDER; ZHANG; HORVATH, 2007);

- **Métodos baseados em densidade** tem por objetivo aumentar o grupo fornecido até que o número de pontos na vizinhança exceda algum limite. Esse método pode ser utilizado para descobrir grupos de formas arbitrárias. Para execução do método são necessários alguns parâmetros como: raio, número mínimo de padrões e um padrão central. Para um padrão ser considerado núcleo é necessário que ele seja maior ou igual ao número mínimo de padrões, conforme especificado anteriormente, dentro de sua vizinhança. Diz-se que um padrão está dentro do raio de um padrão de núcleo se a distância do padrão de núcleo ao seu padrão de vizinhança for menor ou igual ao raio epsilon, definido pelo usuário. O princípio da densidade alcançável foi aplicado aos métodos a partir de um padrão r, quando r é um núcleo e s está dentro da vizinhança epsilon de r. DBSCAN (ESTER et al., 1996), OPTICS (ANKERST et al., 1999) e DENCLUE (HINNEBURG; KEIM et al., 1998) são exemplos típicos nesta classe (PAL; RAY; GANIVADA, 2017);

  Uma aplicação do método DBSCAN pode ser encontrado em (FRANCIS; VILLAGRASA; CLAIRAND, 2011), com adaptação para cálculos de agrupamento de danos no ADN em estágio inicial.

- **Métodos baseados em grade** consistem na divisão do espaço de um objeto em um número finito de células que formam uma estrutura de grade. A grande vantagem dessa abordagem é o tempo de processamento, que normalmente não depende da quantidade de objetos de dados. Além das abordagens mencionadas existem também algoritmos de agrupamento baseados em redes neurais e redes neurais granulares (PAL; RAY; GANIVADA, 2017). Exemplo de agrupamento na estrutura de redes neurais granulares são encontrados em HERBERT; YAO

(2009) , onde a granulação é desenvolvida usando conjuntos difusos.

2.2.5 Tipos de Padrões

Podem existir vários tipos de padrões em proteínas, porém é necessário saber que existem padrões em sequências de proteína e padrões de estrutura. Padrões de estruturas referem-se a estrutura tridimensional das proteínas. Padrões de sequências descrevem características da sequência propriamente dita. Nesse trabalho são abordados apenas os padrões em sequências. Os padrões em sequências podem ser classificados em dois tipos: padrões determinísticos e padrões probabilísticos (também chamados de padrões estocásticos). Por exemplo, C-x(2,4)-[DE] é um padrão de sequência que corresponde a qualquer sequência que contenha uma subcadeia começando com C, seguida por dois e quatro símbolos arbitrários, seguidos por D ou E. Esse padrão é um exemplo de padrão determinístico (um padrão determinístico corresponde ou não a uma determinada sequência). O padrão probabilístico atribui uma probabilidade de ocorrência de um padrão em uma determinada sequência. Um exemplo de padrões probabilísticos são perfis e modelos de Markov ocultos (BRAZMA et al., 1998).

O objetivo do reconhecimento de padrões em biossequências é, dado um conjunto de sequências, encontrar padrões comuns a todas as sequências ou à maioria delas. Com essa condição atendida é possível inferir que, dada uma nova sequência que contenha esse padrão, ela provavelmente compartilha características inerentes a família que o padrão foi extraído, ou até que o padrão encontrado desempenhe um papel importante na determinação da função biológica das proteínas correspondentes (BRAZMA et al., 1998).

Para alcançar o objetivo de encontrar um padrão é necessário escolher uma abordagem ou uma combinação de abordagens à serem utilizadas. As subseções 2.2.6 e 2.2.7 especificam algoritmos determinísticos e estocásticos para melhor compreensão dessa distinção.

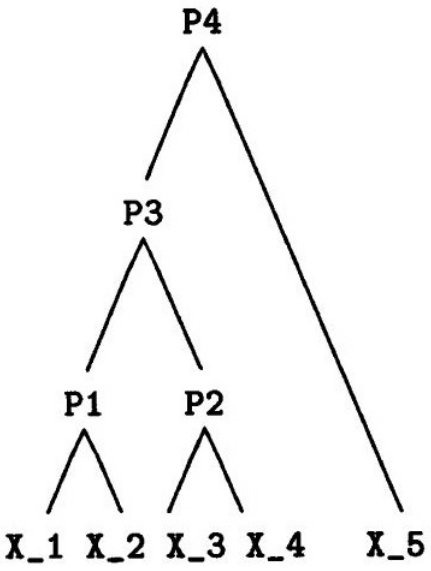
2.2.6 Modelos Determinísticos

Nessa seção são mostrados alguns algoritmos que utilizam abordagem determinística e algumas aplicações.

Método de Comparação de Pares: SMITH; SMITH (1990) apresenta um algoritmo para reconhecimento de padrões em biossequências baseado em programação dinâmica. O algoritmo explora o fato de que os caracteres de um alfabeto α podem ser organizados em grupos hierárquicos parcialmente ordenados.

Inicialmente é construído um dendograma que usa distâncias relativas estimadas entre as sequências. Por exemplo, um possível dendograma das sequências X_1, X_2, X_3, X_4, X_5 onde os pares X_1, X_2, e X_3, X_4 são sequencias com maior similaridade entre si, mas a sequência X_5 é a mais diferente entre elas, como demonstrado na Figura 2.8. Os pares de sequências são alinhados em cada nó do dendrograma que começa de baixo para cima, e um padrão comum é obtido de cada par por meio de programação dinâmica.

Figura 2.8: Exemplo de dendograma para as sequências X_1, X_2, X_3, X_4, X_5.



Fonte: BRAZMA et al. (1998).

Os alinhamentos aos pares garantem um padrão ótimo comum às duas sequências

alinhadas, mas isso não garante a otimização dos padrões mais altos no dendrograma em relação a todas as sequências especificadas. O algoritmo também pode ser usado para classificação no processo de aprendizado não supervisionado (BRAZMA et al., 1998).

Método de Poda Durante a Pesquisa: essa abordagem poda o espaço de pesquisa, as subárvores com baixa aptidão são desconsideradas baseado em algum limite. Pode ser mais econômico na prática, porque a remoção da árvore de pesquisa pode ser mais eficiente do que a pesquisa em profundidade (BRAZMA et al., 1998).

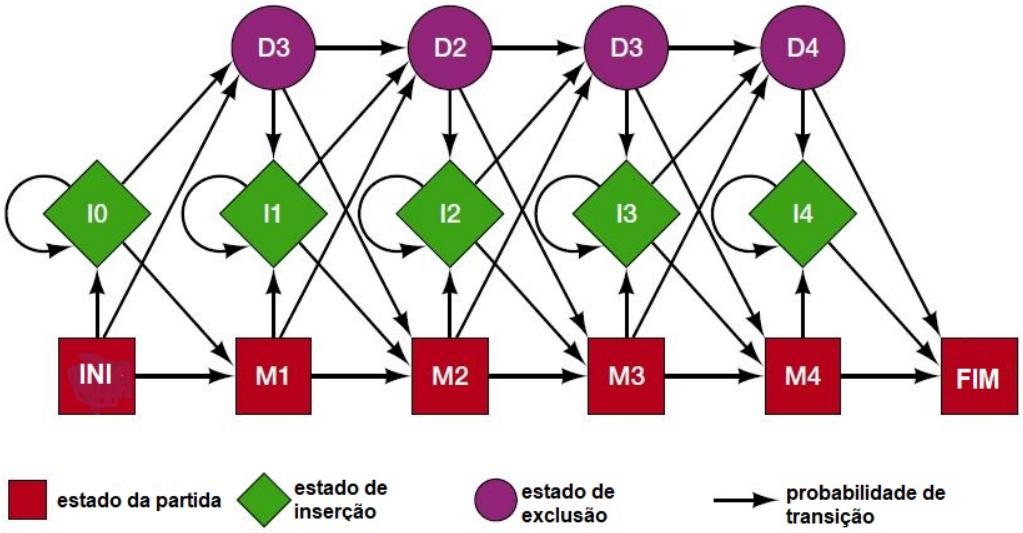
BEGUM et al. (2015) propõe uma nova estratégia de poda que explora tanto limites superiores quanto inferiores para remover uma grande fração dos cálculos de distância.

2.2.7 Modelos Estocásticos

Nessa seção são elencados alguns métodos baseados em probabilidade, amplamente utilizados em bioinfomática.

Modelos de Markov Ocultos (MMO): segundo EDDY (1996) MMO são uma técnica geral de modelagem estatística para problemas "lineares", como sequências ou séries temporais. A ideia principal é que um MMO é um modelo finito que descreve uma distribuição de probabilidade em um número infinito de sequências possíveis, e é constituído por um número de estados. A sequência de estados não é observável e, portanto, é chamada oculta, ela influencia outro processo estocástico que produz uma sequência de observações. Cada estado emite símbolos de acordo com a distribuição de probabilidade de emissão desse estado (Figura 2.9), criando uma sequência observável de símbolos (YU, 2010; EDDY, 1996).

Figura 2.9: Modelo de Markov Oculto.

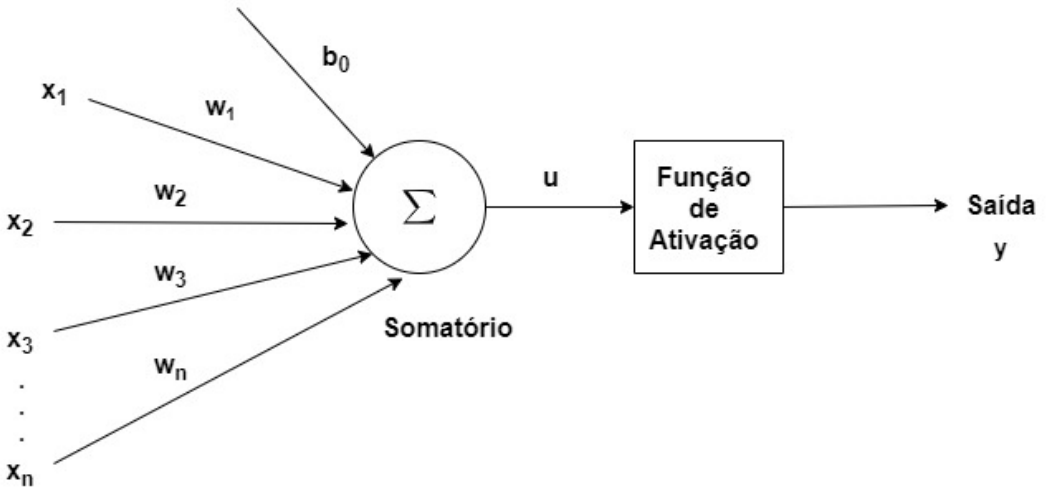


Fonte: adaptado de MOUNT (2004).

PANUCCIO; BICEGO; MURINO (2002) utiliza MMO para agrupar sequências de dados. O principal aspecto desse trabalho é o uso dessa abordagem para derivar novas distâncias de proximidade.

Rede Neural Artificial (RNA): é um sistema composto por vários componentes que são abstrações de células e ligações entre neurônios (sinapse), que ocorrem no cérebro humano e que apresentam alto grau de discriminação, dedução e generalização. A inteligência humana e o poder discriminador são atribuídos principalmente à rede massivamente conectada de neurônios biológicos no cérebro humano. Rede neural artificial é um sistema composto por muitos neurônios artificiais. As RNAs tentam simular o esquema de representação de informações e capacidade de discriminação de neurônios no cérebro humano. Os pesos, chamados de sinapses, são atribuídos a links que conectam os neurônios e comunicam informações entre os nós (Figura 2.10). As características das redes neurais incluem: não linearidade, mapeamento de saída e entrada, adaptabilidade, tolerância a falhas, robustez e otimização (PAL; RAY; GANIVADA, 2017).

Figura 2.10: Arquitetura de um neurônio.



Fonte: adaptado de PAL; RAY; GANIVADA (2017).

VIDAKI et al. (2017) utilizou RNA para previsão de idade forense baseada em metilação de ADN.

Sistema Imunológico Artificial (SIA): são metodologias de manipulação, classificação e representação de dados que seguem um paradigma biologicamente plausível, o do sistema imunológico humano. O SIA pode ser definido como sistemas adaptativos, inspirados na imunologia teórica e nas funções, princípios e modelos observados, aplicados à solução de problemas. A maioria dos SIA exploram apenas um número limitado de ideias e abstrações de alto nível do sistema imunológico. Segundo SOTIROPOULOS; TSIHRINTZIS (2016) é considerado um SIA todo sistema que:

- Utilize no mínimo um modelo básico de um componente imune, como célula, molécula ou órgão;
- É elaborado com base em ideias decorrentes da imunologia teórica e experimental;
- Sua existência é destinada exclusivamente à solução de problemas.

De acordo com SOTIROPOULOS; TSIHRINTZIS (2016) as propriedades de pro-

cessamento de informações mais importantes do sistema imunológico adaptativo que devem ser incorporadas, até certo ponto, algumas características relevantes que qualquer artefato computacional deve ter são: reconhecimento de padrões, exclusividade, identidade própria, diversidade, descartabilidade, autonomia, aprendizagem imune e memória.

Devido a grande quantidade de características computacionais do sistema imunológico adaptativo, esse paradigma é utilizado em diversas áreas como:

- Agrupamento e Classificação;
- Detecção de intrusão;
- Otimização;
- Bioinformática;
- Recuperação de informações e mineração de dados da Web;
- Processamento de imagem.

Como o foco desse trabalho é reconhecimento de padrões em Bioinformática, o escopo é limitado para direcionar ao objetivo desejado. Portanto são abordados algoritmos e métodos para tal finalidade.

Existem vários tipos de células e moléculas imunes que compõem o sistema imunológico. O primeiro passo para projetar um sistema imunológico artificial é conceber um esquema para criar modelos abstratos dessas células e moléculas imunes. Segundo CASTRO (2006) qualquer resposta imune requer o reconhecimento de um antígeno por um receptor celular.

O reconhecimento de padrões pelo SIA é feito pela complementaridade, porém em alguns casos é necessário avaliar por medidas de similaridade entre as estruturas de dados. A estrutura de dados mais comum é uma sequência de atributos, que pode

ser um vetor com valor real, uma sequência inteira, uma sequência binária ou uma sequência simbólica (CASTRO, 2006). Isso resulta em diversos espaços de busca.

- Espaço de busca com valor real: as sequências de atributos são vetores com valor real;
- Espaço de busca com valor inteiro: as sequências são números inteiros;
- Espaço de busca de Hamming: alfabeto finito de comprimento k;
- Espaço de forma simbólico: geralmente composto por diferentes tipos de atributos strings em que pelo menos um deles é simbólico, como “idade”, “altura” etc.

Como existem vários tipos de cadeias de atributos que representam as formas generalizadas de moléculas no sistema imunológico, cada um desses tipos exigirá uma classe específica de medida de afinidade (CASTRO, 2006).

Para espaços de formas com valor real, as medidas de afinidade mais comuns são: distância Euclidiana e Manhattan, dadas pelas Equações 2.1 e Equação 2.2, respectivamente:

$$D = \sqrt[2]{\sum_{i=1}^{L} (Ab_i - Ag_i)^2} \tag{2.1}$$

$$D = \sum_{i=1}^{L} |Ab_i - Ag_i| \tag{2.2}$$

Para espaços de forma de Hamming, a distância de Hamming pode ser usada para avaliar a afinidade entre duas células. Nesse caso, as moléculas são representadas como sequências de símbolos sobre um alfabeto finito de comprimento k. A equação 2.3 mostra a distância de Hamming usada para avaliar a afinidade entre duas sequências de atributos de comprimento L, onde é levada em consideração a complementaridade (CASTRO, 2006):

$$D = \sum_{i=1}^{L} \sigma_i, \text{ onde } \sigma_i = \begin{cases} 1 & \text{se } Ab_i \neq Ag_i \\ 0 & \text{caso contrário} \end{cases} \tag{2.3}$$

A equação 2.4 mostra a fórmula de aplicação para quando a similaridade é levada em consideração:

$$D = \sum_{i=1}^{L} \sigma_i, \text{ onde } \sigma_i = \begin{cases} 1 & \text{se } Ab_i = Ag_i \\ 0 & \text{caso contrário} \end{cases} \tag{2.4}$$

A classe do algoritmo utilizado para reconhecimento de padrões da imunologia é modelada a partir do processo de seleção clonal, usado para gerar população de células imunes acionadas pelos antígenos (CASTRO, 2006).

Seleção Clonal: o princípio da seleção clonal é usado para descrever as características básicas de uma resposta imune adaptativa aos antígenos. Estabelece a ideia de que apenas as células que reconhecem os antígenos proliferam, sendo selecionadas contra aquelas que não o fazem. As células selecionadas estão sujeitas a um processo de maturação por afinidade, o que melhora sua afinidade com os antígenos (CASTRO, 2006).

CLONALG: esse algoritmo pertence ao conjunto de algoritmos baseados em seleção clonal, foi inicialmente criado para reconhecimento de padrões, em seguida, adaptado para resolver tarefas de otimização de funções multimodais (CASTRO, 2006). Consiste nas seguintes etapas.

1. **Inicialização**: criação de uma população aleatória de indivíduos (P);
2. **Apresentação antigênica** para cada padrão de S, faça:
   (a) **Avaliação de afinidade**: para cada indivíduo da população P determine sua afinidade com cada padrão de S;
   (b) **Seleção e expansão clonal**: selecionar n1 elementos de maior afinidade

de P e gerar clones desses indivíduos proporcionalmente à sua afinidade com o antígeno, quanto maior a afinidade, maior o número de cópias e vice-versa;

(c) **Maturação por afinidade**: modificar todas cópias com uma taxa inversamente proporcional à afinidade com o padrão de entrada: quanto maior a afinidade, menor a taxa de mutação e vice-versa;

(d) **Meta-dinâmica**: substituir um número n2 de indivíduos de baixa afinidade por novos (gerados aleatoriamente).

3. **Ciclo**: repetir a etapa 2 até que um determinado critério de parada seja atendido.

Esse algoritmo foi utilizado por ALVES; DELGADO; FREITAS (2010) para predizer funções proteicas descritas na Ontologia Genética (OG) e apresenta um sistema para extração de conhecimento em bancos de dados de proteínas, baseado em um sistema imunológico artificial.

2.3 Trabalhos Correlatos

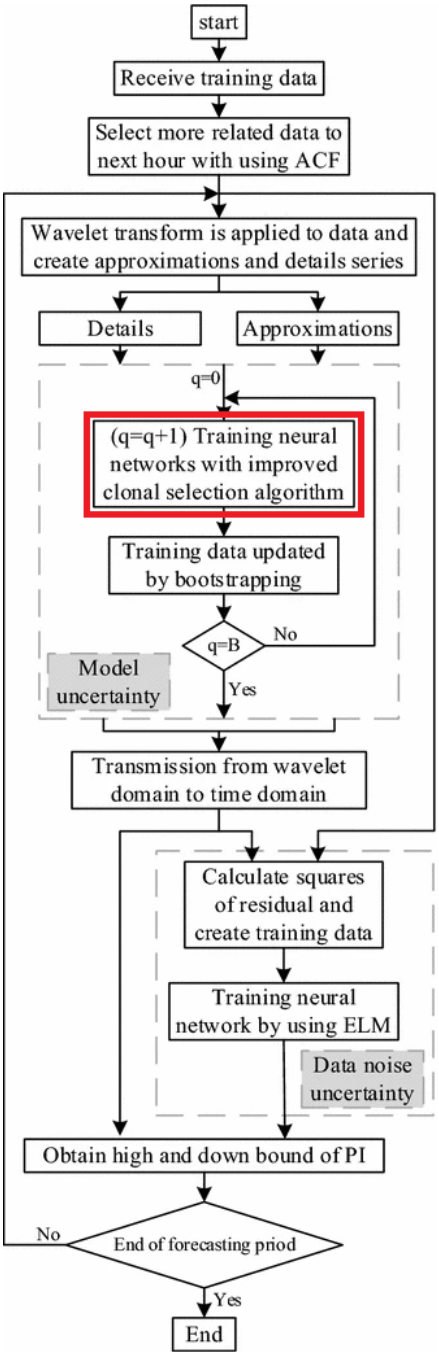
Nesta seção são apresentados trabalhos que utilizaram o CLONALG (Algoritmo de seleção clonal) em diversos tipos de problemas, os quais apresentaram resultados satisfatórios, e também é mostrado na seção 2.3.5 um algoritmo de reconhecimento de padrão que serviu de base para testes e validações. Dessa forma é possível observar características importantes do ponto de vista computacional que serão abordadas mais adiante neste trabalho.

2.3.1 Probabilistic electricity price forecasting by improved clonal selection algorithm and wavelet preprocessing

RAFIEI; NIKNAM; KHOOBAN (2017) salienta neste trabalho as transformações que atravessam a indústria de energia elétrica no mundo, e a caminhada para um mercado desregulado, que precisa cada vez mais de estruturas de previsões. É abordado então a dificuldade de previsão do preço da energia elétrica para planejamento estratégico de mercado. Nesse contexto se faz necessária uma estratégia probabilística para simular a variação dos preços e facilitar a tomada de decisão.

Neste trabalho foi utilizado o CLONALG, transfomada wavelet, e redes neurais artificiais. O CLONALG gera um conjunto de anticorpos que serão utilizados para treinar duas camadas da rede neural artificial, e passam por um processo substituição de anticorpos com baixa afinidade, o que permite escapar do ótimo local, o erro quadrático é considerado como função objetivo (RAFIEI; NIKNAM; KHOOBAN, 2017). É proposto então um método híbrido constituído da combinação dos métodos citados, que está representado na Figura 2.11.

Figura 2.11: Estrutura do modelo algoritmo de seleção clonal e wavelet



Fonte: RAFIEI; NIKNAM; KHOOBAN (2017).

No fluxograma da Figura 2.11 é possível observar mais claramente em qual etapa é aplicado o algoritmo de seleção clonal.

2.3.2 A clonal selection algorithm for dynamic multimodal function optimization

A otimização de funções multimodais é uma tarefa desafiadora que consiste basicamente em encontrar os pontos de ótimos globais, vários estudos no campo da computação inspirada pela natureza tem sido amplamente aplicados nesse contexto. CLONALG é aplicado para a solução de problemas dessa natureza como citado por DASGUPTA; YU; NINO (2011).

Neste estudo LUO et al. (2019) utiliza o métodos de niching (HORN; GOLDBERG; DEB, 1994) para dividir a população em subpopulações, que buscam convergir para um ótimo global. A medida que as subpopulações convergem uma pequena população é gerada. Um operador de mutação da evolução diferencial (DE) é incorporado ao algoritmo para acelerar a convergência, o que leva a um maior desempenho.

Existe uma característica interessante no método que é a aplicação de dois tipos de operadores de mutação, mutação gaussiana e operador de mutação da evolução diferencial (DE) em diferentes etapas do algoritmo.

2.3.3 Prediction of Protein Secondary Structure With Clonal Selection Algorithm and Multilayer Perceptron

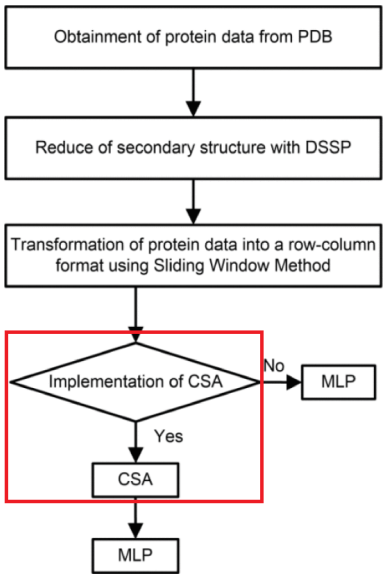
As proteínas podem ser encontradas nas estruturas primária, secundária, terciária e quaternária. A estrutura primária das proteínas se originou das sequências de 20 aminoácidos diferentes em diferentes ordens e comprimentos. Cada um desses 20 aminoácidos é representado por uma letra do alfabeto (A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y).

YAVUZ; YURTAY; OZKAN (2018) implementou um método de predição da estrutura terciária de proteínas, o dicionário de estrutura secundária de proteína (DSSP) foi utilizado como guia. DSSP representa diferentes estruturas secundárias representa-

das por H, G, I, E, B, T, S, e C. Para reduzir a complexidade na predição as estruturas foram reduzidas com a transformação de {H, G} em {H}, {E, B} em {E} e o restante em {C}.

O treinamento inicialmente foi realizado com método de deep learning MLP, em seguida os mesmos dados foram melhorados com CLONALG no processo de clonagem e mutação, antes do processo de classificação do MLP, o que permitiu obter melhor resultado na classificação. A Figura 2.12 exibe o fluxograma do algoritmo.

Figura 2.12: Fluxograma algoritmo de seleção clonal e multilayer perceptron



Fonte: YAVUZ; YURTAY; OZKAN (2018).

2.3.4 Predicting the Protein Tertiary Structure by Hybrid Clonal Selection Algorithms on 3D Square Lattice

A tarefa de predição da estrutura terciária de proteínas é complexa e extremamente necessária, pois a conformação da proteína a partir da interação entre os aminoácidos determina sua função biológica (DILL et al., 1995). Existem várias pesquisas referentes a este contexto, que utilizam a estrutura primária para inferir a estrutura terciária.

FEFELOVA et al. (2020) utiliza o algoritmo de seleção clonal (CLONALG), com

duas estratégias de mutação para aumentar a variabilidade da população, algoritmo de evolução diferencial (DE) e operador de mutação de evolução diferencial trigonométrica (TDE), com intuito de fugir de ótimos locais. Foi elaborado então um algoritmo híbrido para a predição da estrutura terciária.

De acordo com *HP Dill model* (HIRST, 1999) os aminoácidos são divididos em dois tipos: hidrofílicos e hidrofóbicos, solúveis em água e insolúveis, respectivamente (AFTABUDDIN; KUNDU, 2007). Os hidrofílicos são denotados por P e os hidrofóbicos por H. A sequência de aminoácidos pode ser representada por S = (s_1, s_2, s_3,..., s_n), $s_i \in \{H, P\}$, $i = \overline{1,n}$.

Segundo FEFELOVA et al. (2020), utilizar estratégias de mutação possibilitaram melhorar significativamente a busca e acelerar a convergência do algoritmo. Os experimentos mostraram um aumento geral na qualidade do resultado do algoritmo proposto em comparação com métodos existentes.

2.3.5 Discovering genomic patterns in SARS-CoV-2 variants

Segundo D'ANGELO; PALMIERI (2020) em 31 Dezembro de 2019 em Wuhan (província de Hubei, China) foi identificado um surto de casos de pneumonia de etiologia desconhecida. Em 9 de janeiro de 2020, o Centro Chinês para Controle e Prevenção de Doenças (CDC) reconheceu um novo coronavírus SARS-CoV-2. Em 11 de março de 2020, a Organização Mundial da Saúde (OMS) declarou a COVID-19 como a doença causada pelo SARS-CoV-2, e um alerta ao mundo de uma pandemia global.

A partir de então se inicia uma corrida contra o tempo para obtenção do código génetico sequenciado do coronavírus, que serviria para a manipulação de vacinas. Com a elevada taxa de mutação do vírus se torna ainda mais urgente encontrar em seu código genético o local responsável pela infecção das células do hospedeiro. No decorrer dos estudos foi descoberto que o coronavírus sintetiza a proteína Spike, responsável pelas infecções. Portanto, os cientistas iniciam exaustivos testes para encontrar padrões na

região responsável pela síntese dessa proteína.

D'ANGELO; PALMIERI (2020) propõe a criação de dois algoritmos para reconhecimento de padrões. O primeiro é responsável por descobrir subsequências comuns chamado de PCSS (pyramidal-based common subsequences search), enquanto o último SCPS (Sliding columns-based pattern search) é usado para encontrar múltiplas combinações dessas substrings.

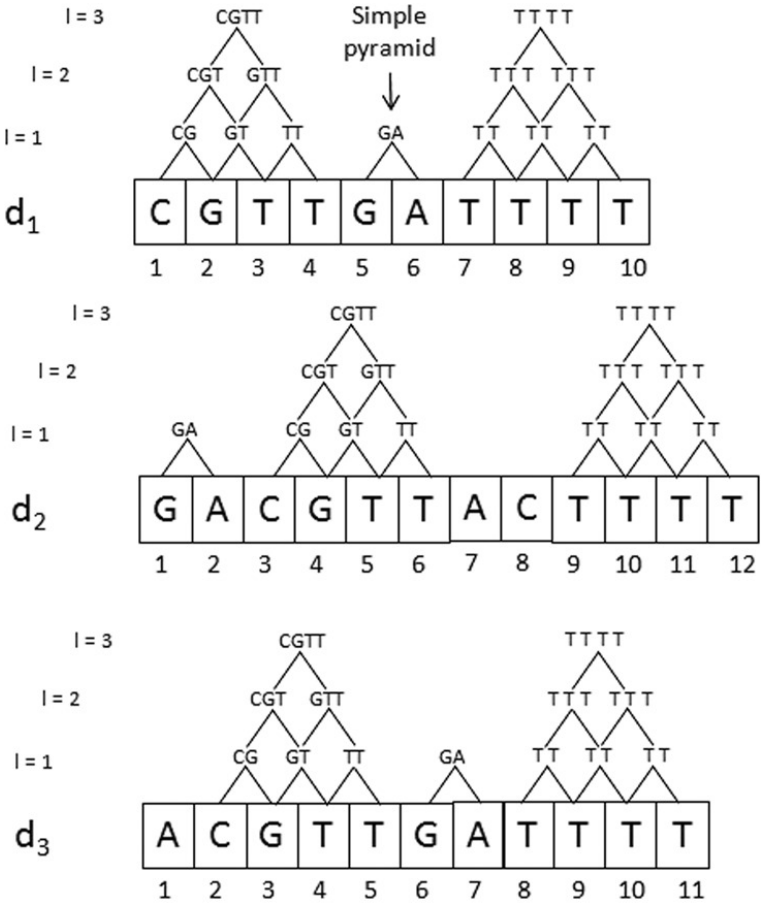
A ideia do algoritmo PCSS é baseada no processo de junção e poda, que é responsável por juntar subsequências comuns consecutivas para formar subsequências mais longas e podar subsequências que não são consecutivas. As subsequências são criadas na forma de pirâmide como demonstrado na Figura (2.13), baseadas nos genomas a seguir:

$$d_1 = CGTTGATTTT, \tag{2.5}$$

$$d_2 = GACGTTACTTT, \tag{2.6}$$

$$d_3 = ACGTTGATTTT. \tag{2.7}$$

Figura 2.13: Representação esquemática do algoritmo de busca de subsequências comuns com base piramidal (PCSS).



Fonte: D'ANGELO; PALMIERI (2020).

No primeiro nível são criadas algumas pirâmides candidatas ao segundo nível, para isso é necessário que haja uma correspondência com todos os demais genomas, caso contrário o processo de poda é executado. Assim, as pirâmides consecutivas ***CG*** e ***GT*** do d_1 representam as bases da nova pirâmide ***CGT*** no $l = 2$ conforme representado. Repetindo essas etapas para todas as outras pirâmides consecutivas de $l = 1$, as novas pirâmides ***GTT*** e ***TTT*** também são implementadas. Já ***GA*** no Nível 1 não está envolvida na construção de nenhuma pirâmide no Nível 2, porque não é adjacente a nenhuma outra. Por fim, no nível 3, as pirâmides ***CGTT*** e ***TTTT*** construídas a partir dos pares ***CGT - GTT*** e ***TTT - TTT*** do Nível 2, respectivamente.

O algoritmo SCPS é dedicado a descobrir padrões repetitivos de substrings co-

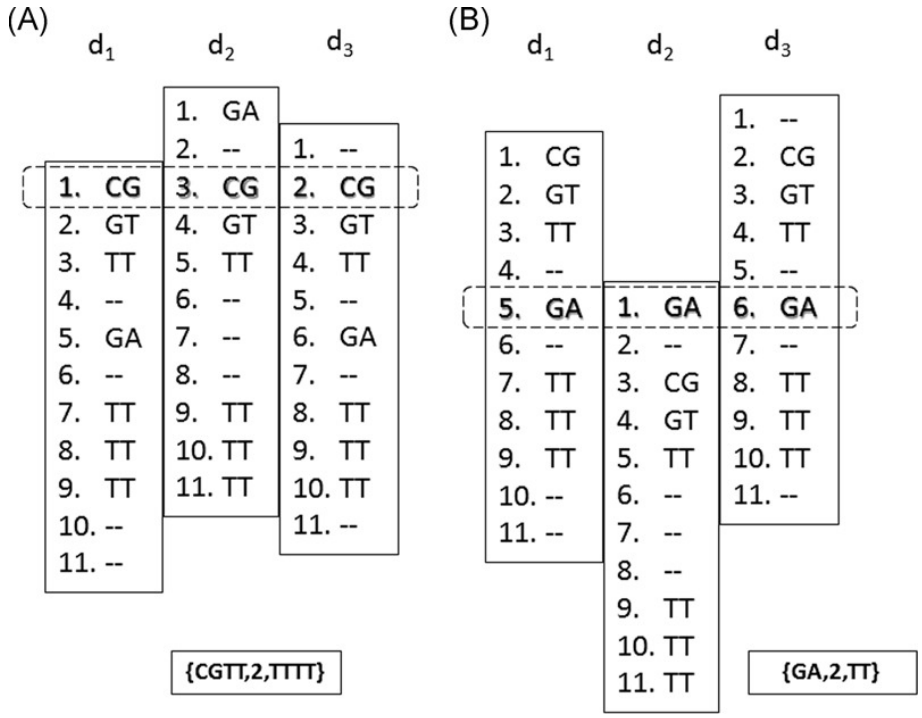
muns. Para tanto, um processo de alinhamento de colunas da matriz M^l (Figura 2.14) é usado. Mais precisamente, para um determinado nível $\boldsymbol{l}$ uma determinada substring comum do genoma de referência, as colunas dos genomas restantes são deslocadas para serem alinhadas com a substring considerada, como demonstrado na Figura 2.15.

Figura 2.14: Matriz M^l para $\boldsymbol{l}$ = 1 , 2 , 3.

	d_1			d_2			d_3		
idx	$l = 1$	$l = 2$	$l = 3$	$l = 1$	$l = 2$	$l = 3$	$l = 1$	$l = 2$	$l = 3$
1	CG	CGT	CGTT	GA	-	-	-	-	-
2	GT	GTT	-	-	-	-	CG	CGT	CGTT
3	TT	-	-	CG	CGT	CGTT	GT	GTT	-
4	-	-	-	GT	GTT	-	TT	-	-
5	GA	-	-	TT	-	-	-	-	-
6	-	-	-	-	-	-	GA	-	-
7	TT	TTT	TTTT	-	-	-	-	-	-
8	TT	TTT	-	-	-	-	TT	TTT	TTTT
9	TT	-	-	TT	TTT	TTTT	TT	TTT	-
10	-	-	-	TT	TTT	-	TT	-	-
11	-	-	-	TT	-	-	-	-	-

Fonte: D'ANGELO; PALMIERI (2020).

Figura 2.15: Algoritmo de pesquisa de padrão baseado em colunas deslizantes (SCPS).



Fonte: D'ANGELO; PALMIERI (2020).

Ao final desse processo é obtido um padrão que servirá como assinatura da proteína Spike que pode ser útil para manipulação de drogas e vacinas para o combate a COVID-19. O padrão pode possuir alguns números entre as subsequências que significa a quantidade de caracteres irrelevantes na construção do padrão. O padrão encontrado é exibido a seguir:

$$\Psi = \{TAAA, 6, ATG, 5, TTT, 79, TCT, 37, GAT, 88, ...$$
$$..., TGG, 1, ACT, 43, ATTTG, 23, ATA, 62, CGCT, 30, CAA\}$$

2.4 Considerações Parciais

Neste capítulo foram abordados conceitos para compreensão da biologia molecular, assim como métodos para reconhecimento de padrões utilizados na ciência para resolução de diversos problemas, ambos utilizados em Bioinformática. Além disso, foram elencadas algumas aplicações do CLONALG e suas contribuições, assim como, os desafios encontrados na implementação, a exploração das características intrínsecas da abordagem de SIA como: mutação, seleção, clones e memória imunológica. Estes trabalhos contribuem para o entendimento do algoritmo e as diversas formas que pode ser aplicado, e também como pode ser combinado com outras abordagens, afim de melhorar alguns aspectos como variabilidade e maior probalidade de encontrar ótimos globais. Também foi explorado o algoritmo descrito na seção 2.3.5 que foi utilizado como base para avaliação do método proposto neste trabalho, afim de validar os resultados e servir de inspiração para a utilização do CLONALG-MMO (descrito na seção 3) como alternativa para abordagens evolutivas, como citado por D'ANGELO; PALMIERI (2020) em suas atividades futuras. Assim, serão apresentadas, nos capítulos subsequentes, as melhorias desenvolvidas no presente trabalho, quando comparadas a estas abordagens anteriormente dispostas.

Capítulo 3

Metodologia

Neste capítulo serão descritas em detalhes quais técnicas, conjunto de sequências e medidas de distância o algoritmo utiliza, afim de elucidar seu funcionamento e contribuições para o processo de reconhecimento de padrões.

3.1 CLONALG e MMO

O algoritmo CLONALG é baseado na abordagem de SIA que se enquadra no conjunto de algoritmos evolutivos bioinspirados como descrito na seção 2.2.7. Optou-se pelo Modelo de Markov Oculto (MMO) como medida de distância, ao invés das medidas de distância padrão, distância de Hamming e distância Euclidiana por exemplo. A justificativa para adoção de tal abordagem como já descrito anteriormente é sua característica estocástica, que oferece uma solução satisfatória em um tempo de execução aceitável e um resultado com relevânica biológica.

Foram realizados testes com a distância de Hamming, os resultados obtidos do ponto de vista computacional não foram muito satisfatórios, pois abordagens determinísticas para uma grande quantidade de sequências torna o processo infactível, tendo em vista que o algoritmo necessita de algumas iterações para aumentar a qualidade do padrão. Assim, o MMO foi implementado, por ser uma abordagem amplamente uti-

lizada no contexto de reconhecimento de padrões como descrito por SUN; BUHLER (2007).

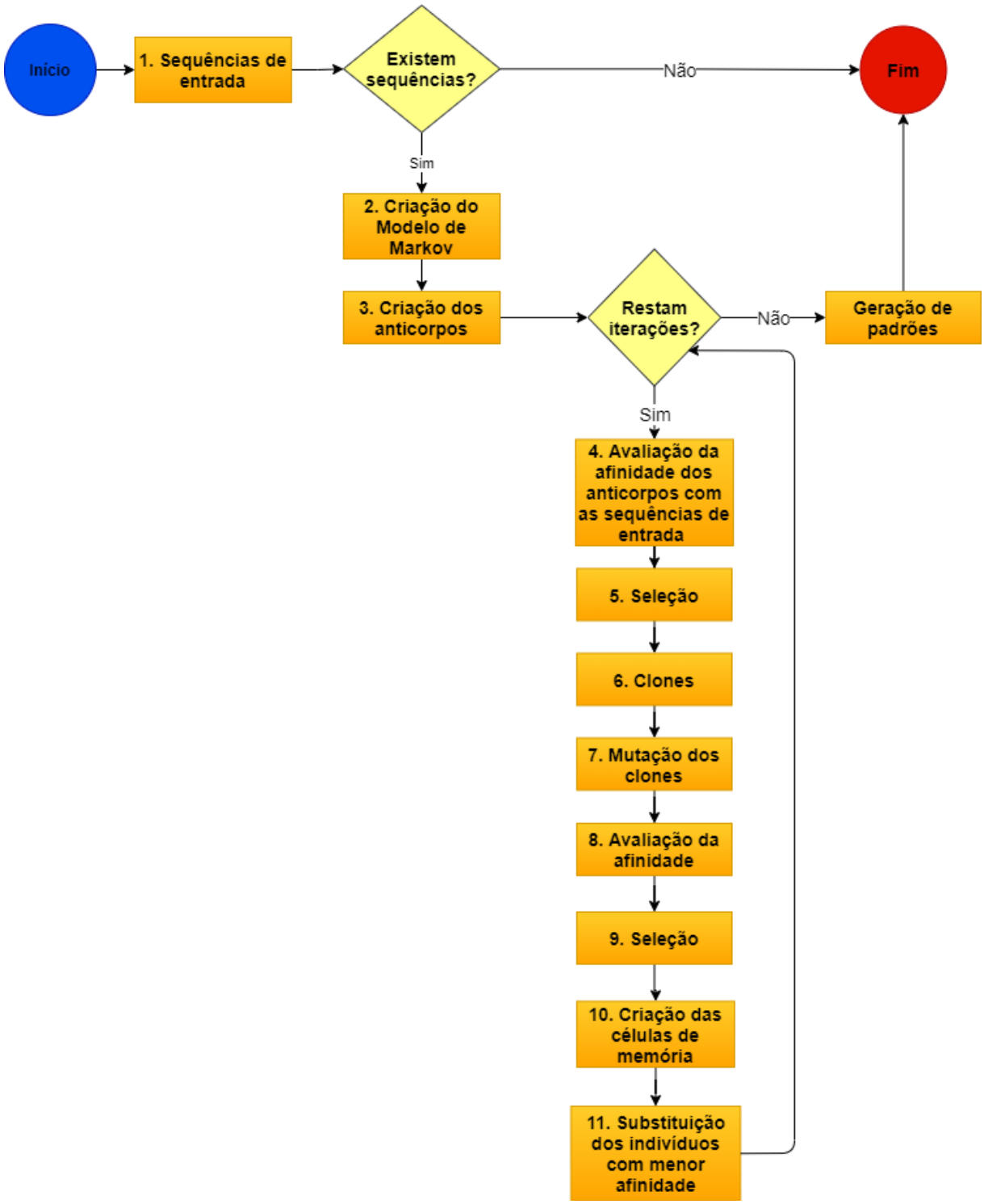
Para avaliar a afinidade dos anticorpos (subsequências) com os antigênos que são as sequências de entrada, o algoritmo utiliza a probabilidade de uma subsequência ocorrer no conjunto de sequências de entrada. Com essa medida de afinidade é possível aplicar uma mutação inversamente proporcional à afinidade, ou seja, quanto maior a afinidade do anticorpo com os antígenos (sequências de entrada) menor a taxa de mutação, e uma taxa para geração de clones diretamente proporcional a afinidade do anticorpo, como descrito por CASTRO (2006).

Os parâmetros fornecidos pelo usuário para o algoritmo são:

- `S`: conjunto de biossequências;
- `max_it`: número de iterações a serem executadas;
- `n1`: percentual de anticorpos com alta afinidade;
- `n2`: percentual de anticorpos com baixa afinidade.

Na figura 3.1 exibe-se o fluxo principal do algoritmo.

Figura 3.1: Fluxograma do algoritmo CLONALG-MMO.



Fonte: Próprio autor.

O algoritmo recebe um conjunto de sequências que são utilizadas para o treina-

mento do MMO, posteriormente são geradas as subsequências que são os possíveis padrões que variam de 3 a 9 nucleotídeos, como utilizado por D'ANGELO; PALMIERI (2020). Cada nucleotídeo será gerado de maneira aleatória, afim de formar uma subsequência, que será avaliada através da probalidade de pertencer ao conjunto de sequências de entrada. São selecionadas para a próxima iteração as subsequências com maior probabilidade, de acordo com o parâmetro de afinidade máxima `n1` e são descartadas as subsequências com baixa afinidade conforme o parâmetro de afinidade mínima `n2`, ambos informados pelo usuário.

O fluxo de processamento é descrito a seguir, com as etapas do funcionamento do algoritmo CLONALG com o MMO.

- **Etapa 1:** sequências de ADN são fornecidas ao algoritmo para a extração dos padrões. Neste trabalho foram utilizadas sequências de ADN do SARS-CoV-2;
- **Etapa 2:** é criado um Modelo de Markov para representar essas sequências, a ser descrito na subseção 3.1.1;
- **Etapa 3:** são gerados os anticorpos (subsequências) de maneira aleatória de tamanhos entre 3 e 9 nucleotídeos. Primeiramente é sorteado um número entre 3 e 9 considerando ambos, que é o tamanho da subsequência, supondo que esse número seja 3, serão feitos 3 sorteios de índices em um vetor de quatro posições, cada posição representa uma letra do alfabeto {A, C, T, G};
- **Etapa 4:** são avaliadas as afinidades dos anticorpos com os antígenos (sequências de entrada). Essa avaliação é feita para todas as sequências e todos os padrões, ou seja, a primeira sequência será avaliada com todos os anticorpos, assim todos os anticorpos terão um valor de afinidade com essa sequência e esse valor será usado na etapa 5;
- **Etapa 5:** são selecionados os anticorpos com maior afinidade, baseado em uma porcentagem informada pelo usuário no parâmetro `n1`. A seleção é feita de

acordo com o valor da afinidade definida na etapa 4;

- **Etapa 6:** os anticorpos com maior afinidade são clonados com uma taxa diretamente proporcional a sua afinidade, dessa forma é possível aumentar a chance de ter maior número de anticorpos com alta afinidade;
- **Etapa 7:** os clones são mutados a uma taxa inversamente proporcional a sua afinidade, no intuito de aumentar ainda mais a afinidade dos mesmos;
- **Etapa 8:** são avaliadas as afinidades dos clones em relação aos antígenos, da mesma forma que na etapa 4;
- **Etapa 9:** são selecionados os clones com as maiores afinidades, de acordo com o parâmetro `n1`;
- **Etapa 10:** são criadas as células de memórias, que armazenam os anticorpos, durante a execução do algoritmo essas células de memória podem ser substituídas por outras com maior afinidade;
- **Etapa 11:** os anticorpos com as menores afinidades são substituídos por outros, de acordo com o parâmetro `n2` informado pelo usuário, são gerados novos anticorpos de forma aleatória;
- **Etapa 12:** os anticorpos (subsequências) gerados ao final das iterações são passados como parâmetro para o método que gera os padrões.

Ao final da execução do algoritmo são extraídos os padrões das sequências de entrada. Para validação desses padrões foram realizadas comparações com os padrões encontrados por D'ANGELO; PALMIERI (2020) e serão mostrados na seção 4.

3.1.1 Modelo de Markov Oculto como medida de afinidade

O Modelo de Markov Oculto é construído a partir das sequências de entrada, chamadas de antígenos na abstração do algoritmo CLONALG. O modelo é criado baseado

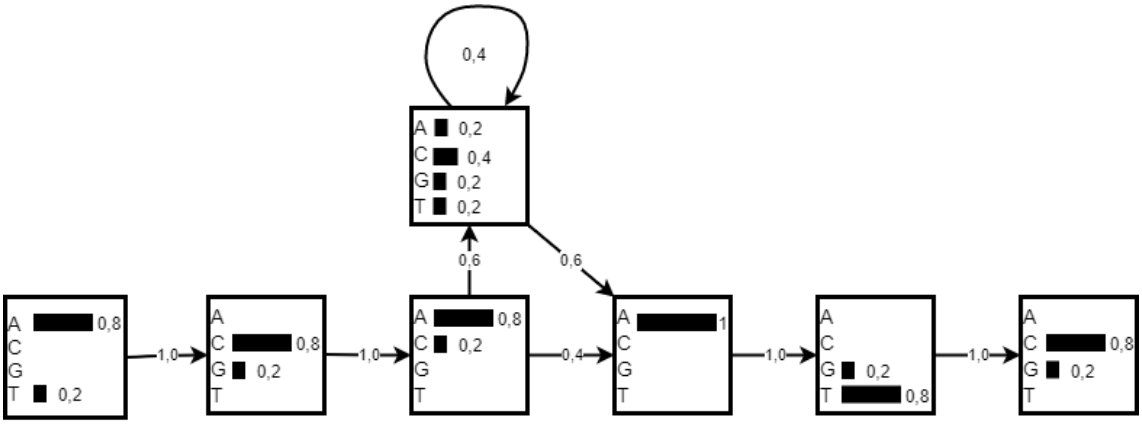
na contagem de todos os caracteres do alfabeto (A, C, T, G) das sequências na posição ***i***, onde ***i*** é o índice da coluna na linha ***j***. Suponha as seguintes sequêcias de ADN:

Tabela 3.1: Exemplos de sequências de ADN

	1	2	3	4	5	6	7	8	9
1	A	C	A	-	-	-	A	T	G
2	T	C	A	A	C	T	A	T	C
3	A	C	A	C	-	-	A	G	C
4	A	G	A	-	-	-	A	T	C
5	A	C	C	G	-	-	A	T	C

A partir das sequências da Tabela 3.1 uma pontuação é gerada com 4/5=0,8 para um A na primeira posição e 1/5=0,2 para um T porque se observa que das 5 letras, 4 são As e 1 é T. Similarmente na segunda posição a probabilidade do C é 4/5 e do G 1/5, e assim por diante. Depois da terceira posição no alinhamento, 3 das 5 sequências têm 'inserções' de comprimentos diferentes, então é dito que a probabilidade de se ter uma inserção é de 3/5 e, consequentemente, 2/5 de não ter (que correspondem às sequências que possuem 3 buracos nas posições 3, 4 e 5). O diagrama seguinte mostra estas probabilidades.

Figura 3.2: Modelo de Markov Oculto.



Fonte: Próprio autor.

Supondo que se deseja avaliar a pontuação da sequência ACACATC, tem-se a seguinte Equação 3.1:

$$P(ACACATC) = 0{,}8 * 1 * 0{,}8 * 1 * 0{,}8 * 0{,}6 * 0{,}4 * 0{,}6 * 1 * 1 * 0{,}8 * 1 * 0{,}8 \approx 4{,}7 * 10^{-2} \tag{3.1}$$

Desta forma é possível avaliar a pontuação das subsequências geradas pelo algoritmo CLONALG, e assim utilizar o mecanismo de seleção das subsequências com maior afinidade com o MMO. Isso permite gerar uma população de forma aleatória, e no decorrer das iterações o aumento da afinidade. Na seção 4 são apresentados quais dados foram utilizados nos testes e quais resultados foram obtidos.

Capítulo 4

Testes e Resultados

O objetivo dos testes é encontrar padrões na região dos genomas do SARS-CoV-2 responsável por sintetizar a proteína Spike, que segundo D'ANGELO; PALMIERI (2020) se encontra na posição 21300 a 25400. O conjunto de dados utilizado nos testes foi baixado do banco de dados Genbank[1], 100 sequências foram utilizadas e selecionadas no período de 01/01/2020 a 03/06/2020 (coluna "Release Date") e ordernadas pelo (coluna "Length") de forma decrescente. Com essas sequências foi possível confrontar os padrões obtidos com os padrões encontrados por D'ANGELO; PALMIERI (2020).

Todos os testes foram executados em ambiente C# .NET Core 2.2 em um PC Windows 10 de 64 bits, com CPU Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz com 8,00 GB de RAM. O projeto está disponível no Github[2] para consulta e possíveis atualizações.

Primeiramente é necessário provar que o algoritmo é capaz de extrair padrões, para tanto foi executado com ***n1 = 0,6***, que significa que 60% dos anticorpos com alta afinidade serão selecionados para a próxima geração, ***n2 = 0,4*** que significa 40% dos anticorpos com baixa afinidade serão substituidos por outros gerados aleatóriamente,

[1] <https://www.ncbi.nlm.nih.gov/genbank/sars-cov-2-seqs/#reference-genome>
[2] <https://github.com/lpliberato/CLONALG-HMM>

e por fim o parametro ***max_it = 100***, que é a quantidade de iterações do algoritmo. A Tabela 4.1 exibe os padrões encontrados.

Tabela 4.1: Padrões encontrados para ***n1 = 0,6; n2 = 0,4*** e ***max_it = 100***

	Padrões	
	CLONALG-MMO	D'ANGELO; PALMIERI
1	TTT	TTT
2	GAT	GAT
3	TCT	TCT
4	ATA	ATA
5	ATG	ATG
6	CAA	CAA
7	ACT	ACT
8	TGG	TGG
9	TAAA	TAAA
10	CGCT	CGCT
11	-	ATTTTG

Com os resultados exibidos na Tabela 4.1 é possível verificar que o algoritmo de fato é capaz de encontrar padrões. Mas somente com uma execução do mesmo, não é adequado para se ter uma visão mais ampla do comportamento dos dados, visto que, se trata de um algoritmo evolutivo que pode oferecer soluções aproximadas. Devido a esse motivo, seguem algumas tabelas que possuem dados referentes a 10 execuções para cada iteração (max_it), com médias de quantidade de padrões, tempo médio e desvios-padrão.

Ao análisar as Tabelas 4.2 à 4.17, pode-se ter uma visão mais ampla acerca do comportamento dos dados. Nota-se que a quantidade média de padrões encontrados variam entre 8 e 9, sendo que o resultado obtido por D'ANGELO; PALMIERI (2020) foi de 11. É natural que a média de padrões encontrados seja menor que 11 por se tratar de um modelo probabilístico, o que não quer dizer que em alguma execução do algoritmo não tenha sido encontrado os 11 padrões. O algoritmo CLONALG-MMO utiliza como medida de afinidade o Modelo de Markov Oculto que é uma abordagem estocástica, para avaliar a sua eficiência foram realizadas várias comparações com a

medida de afinidade de Hamming, que é uma medida determinística, e foi comprovado que CLONALG-MMO oferece resultados melhores (maior quantidade de padrões e menor tempo de execução) como demonstrado nas tabelas 4.2 à 4.17.

Tabela 4.2: Qtd. Padrões com ***n1 = 0,6*** e ***n2 = 0,4***

		CLONALG-MMO	
Execução	max_it	Padrões (média)	Desvio Padrão
1	10	9	1,33333
2	50	9	0,94281
3	100	≈ 9	0,73786
4	500	≈ 9	0,63246
5	1000	≈ 9	0,82327
6	5000	≈ 9	0,97183

Tabela 4.3: Tempo de execução (em segundos) com ***n1 = 0,6*** e ***n2 = 0,4***

		CLONALG-MMO	
Execução	max_it	Tempo em segundos	Desvio Padrão
1	10	0,11678	0,06630
2	50	0,10760	0,02421
3	100	0,09012	0,02051
4	500	0,11686	0,06416
5	1000	0,11008	0,02843
6	5000	0,14025	0,04547

Tabela 4.4: Qtd. Padrões com ***n1 = 0,6*** e ***n2 = 0,4***

		CLONALG-HAMMING	
Execução	max_it	Padrões (média)	Desvio Padrão
1	10	≈ 8	1,42984
2	50	≈ 8	0,91894
3	100	≈ 8	0,94868
4	500	≈ 9	0,87560
5	1000	≈ 9	0,67495
6	5000	9	0,00000

Tabela 4.5: Tempo de execução (em segundos) com ***n1 = 0,6*** e ***n2 = 0,4***

		CLONALG-HAMMING	
Execução	max_it	Tempo em segundos	Desvio Padrão
1	10	0,24764	0,06673
2	50	0,25940	0,03678
3	100	0,23959	0,02531
4	500	0,25510	0,04029
5	1000	0,23930	0,06151
6	5000	0,30113	0,03530

Tabela 4.6: Qtd. Padrões com ***n1 = 0,7*** e ***n2 = 0,3***

		CLONALG-MMO	
Execução	max_it	Padrões (média)	Desvio Padrão
1	10	≈ 9	1,03280
2	50	≈ 9	0,82327
3	100	9	0,66667
4	500	≈ 9	0,78881
5	1000	≈ 9	0,78881
6	5000	≈ 9	0,91894

Tabela 4.7: Tempo de execução (em segundos) com ***n1 = 0,7*** e ***n2 = 0,3***

		CLONALG-MMO	
Execução	max_it	Tempo em segundos	Desvio Padrão
1	10	0,10391	0,08034
2	50	0,09341	0,01493
3	100	0,10182	0,01858
4	500	0,10060	0,02985
5	1000	0,11550	0,01852
6	5000	0,13532	0,02013

Tabela 4.8: Qtd. Padrões com ***n1 = 0,7*** e ***n2 = 0,3***

		CLONALG-HAMMING	
Execução	max_it	Padrões (média)	Desvio Padrão
1	10	≈ 9	0,82327
2	50	≈ 9	0,84984
3	100	≈ 8	1,13529
4	500	≈ 9	0,84327
5	1000	≈ 8	0,96609
6	5000	≈ 8	1,28668

Tabela 4.9: Tempo de execução (em segundos) com ***n1 = 0,7*** e ***n2 = 0,3***

		CLONALG-HAMMING	
Execução	max_it	Tempo em segundos	Desvio Padrão
1	10	0,28211	0,16348
2	50	0,26065	0,05568
3	100	0,23601	0,04286
4	500	0,25395	0,05403
5	1000	0,26503	0,02161
6	5000	0,31255	0,09295

Tabela 4.10: Qtd. Padrões com ***n1 = 0,8*** e ***n2 = 0,2***

		CLONALG-MMO	
Execução	max_it	Padrões (média)	Desvio Padrão
1	10	≈ 8	1,71270
2	50	≈ 9	0,73786
3	100	≈ 9	0,70711
4	500	≈ 9	1,17851
5	1000	≈ 9	1,05935
6	5000	≈ 9	1,22927

Tabela 4.11: Tempo de execução (em segundos) com ***n1 = 0,8*** e ***n2 = 0,2***

		CLONALG-MMO	
Execução	max_it	Tempo em segundos	Desvio Padrão
1	10	0,13173	0,13575
2	50	0,10288	0,01637
3	100	0,08619	0,01572
4	500	0,11110	0,02962
5	1000	0,10138	0,01722
6	5000	0,14287	0,02120

Tabela 4.12: Qtd. Padrões com ***n1 = 0,8*** e ***n2 = 0,2***

		CLONALG-HAMMING	
Execução	max_it	Padrões (média)	Desvio Padrão
1	10	≈ 8	0,69921
2	50	≈ 8	1,07497
3	100	≈ 9	0,78881
4	500	9	0,81650
5	1000	≈ 9	1,19722
6	5000	≈ 8	1,33749

Tabela 4.13: Tempo de execução (em segundos) com ***n1 = 0,8*** e ***n2 = 0,2***

		CLONALG-HAMMING	
Execução	max_it	Tempo em segundos	Desvio Padrão
1	10	0,26571	0,17477
2	50	0,23452	0,04880
3	100	0,26310	0,04446
4	500	0,24691	0,04951
5	1000	0,26617	0,03496
6	5000	0,28276	0,03237

Tabela 4.14: Qtd. Padrões com ***n1 = 0,9*** e ***n2 = 0,1***

		CLONALG-MMO	
Execução	max_it	Padrões (média)	Desvio Padrão
1	10	≈ 8	1,42984
2	50	≈ 9	0,69921
3	100	9	1,24722
4	500	≈ 9	0,82327
5	1000	≈ 9	1,31656
6	5000	≈ 9	0,69921

Tabela 4.15: Tempo de execução (em segundos) com ***n1 = 0,9*** e ***n2 = 0,1***

		CLONALG-MMO	
Execução	max_it	Tempo em segundos	Desvio Padrão
1	10	0,16282	0,22475
2	50	0,10084	0,01539
3	100	0,09575	0,02305
4	500	0,09095	0,01779
5	1000	0,11448	0,03287
6	5000	0,12337	0,02006

Tabela 4.16: Qtd. Padrões com ***n1 = 0,9*** e ***n2 = 0,1***

		CLONALG-HAMMING	
Execução	max_it	Padrões (média)	Desvio Padrão
1	10	≈ 8	0,78881
2	50	≈ 9	0,84984
3	100	≈ 9	0,91894
4	500	≈ 8	0,63246
5	1000	≈ 9	0,94868
6	5000	≈ 8	0,96609

Tabela 4.17: Tempo de execução (em segundos) com ***n1 = 0,9*** e ***n2 = 0,1***

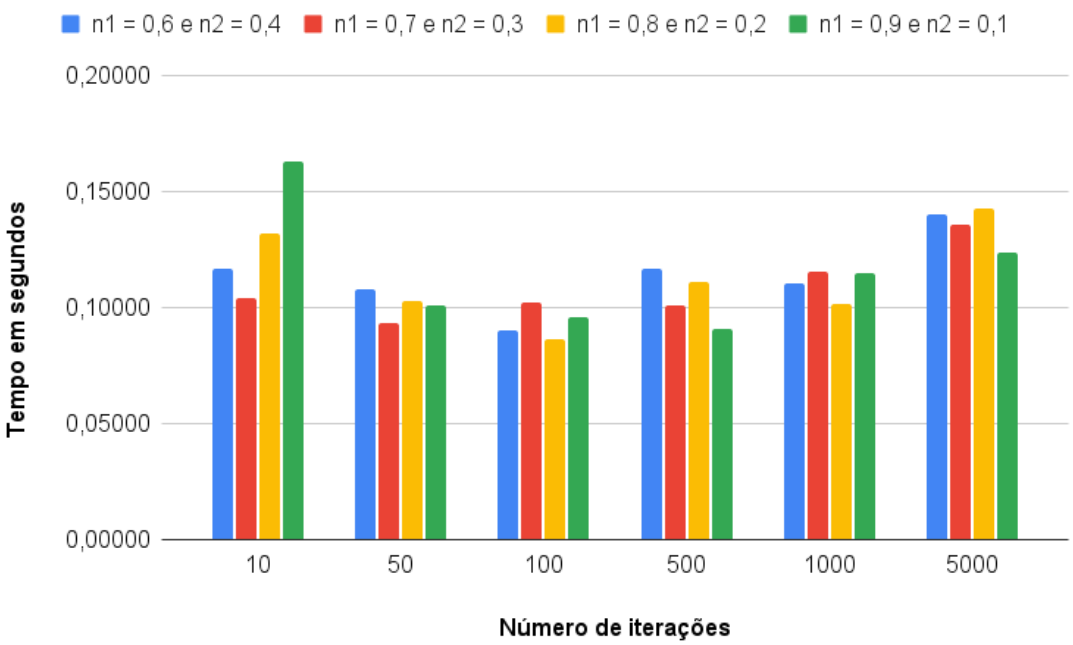
		CLONALG-HAMMING	
Execução	max_it	Tempo em segundos	Desvio Padrão
1	10	0,22105	0,07671
2	50	0,27023	0,05026
3	100	0,25076	0,03600
4	500	0,25832	0,02078
5	1000	0,24824	0,05242
6	5000	0,27329	0,05595

O intuito é oferecer uma abordagem evolutiva para a solução do reconhecimento de padrões, para corroborar com os trabalhos futuros citados por D'ANGELO; PALMIERI (2020). Portanto, este trabalho pode contribuir com o avanço das pesquisas neste tema, e a utilização da abordagem CLONALG em Bioinformática.

O gráfico da Figura 4.1 traz o tempo de execução de acordo com as iterações, em que é possível observar que quando ***n1 = 0,6*** e ***n2 = 0,4*** para 10 iterações o tempo de execução foi um dos menores, e a medida que as iterações aumentaram o tempo também aumentou, quando analisado as amostras em conjunto. A mesma análise aplicada a ***n1 = 0,7*** e ***n2 = 0,3*** permite afirmar que o tempo para 10 iterações foi menor que o primeiro caso, já na iteração 1000 foi superior. Repetindo esta avaliação para todas amostras, tomando a iteração 10 e 1000 como referência, temos um comportamento interessante, que é uma leve tendência de a medida que o ***n2*** diminui e as iterações aumentam, o tempo de execução diminui. Não é objetivo deste trabalho avaliar esse

comportamento, uma vez que esforços foram dedicados a encontrar padrões com a combinação dos dois algoritmos já demonstrados (CLONALG-MMO) de forma até então inédita em Bioinformática, o que trouxe bons resultados, e uma gama de novas opções de estratégias de programação, porém, em trabalhos futuros testes poderão ser realizados para aferir tal comportamento.

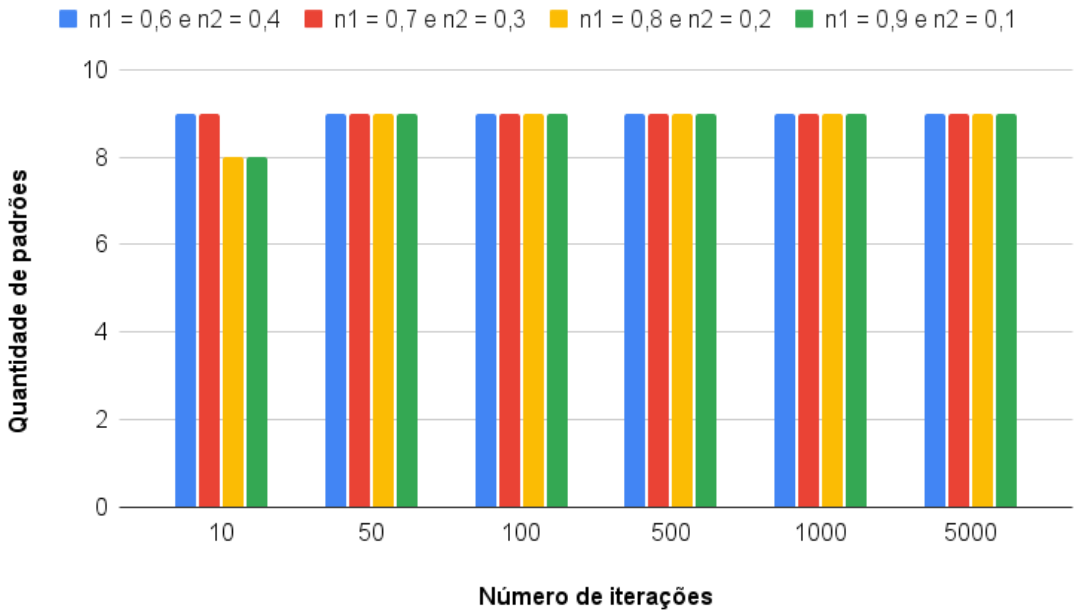
Figura 4.1: Gráfico de tempo por iterações CLONALG-MMO.



Fonte: Próprio autor.

No gráfico da Figura 4.2 é mostrado a quantidade de padrões encontrados em relação às iterações, que possui um comportamento parecido a partir da iteração 50. Algumas otimizações no método de mutação por exemplo, podem trazer melhorias na descoberta de padrões, pois podem aumentar a chance de gerar um anticorpo com maior afinidade. Na seção 5 serão elencados os pontos de possíveis melhorias em trabalhos futuros.

Figura 4.2: Gráfico de quantidade de padrões por iterações CLONALG-MMO.



Fonte: Próprio autor.

Capítulo 5

Conclusão

Neste trabalho foram abordados conceitos relevantes à biologia e a Bioinformática, além de apresentar aplicações de abordagens relacionadas ao reconhecimento de padrões. Nesse contexto foram analisados alguns dos métodos de reconhecimento de padrões determinísticos e estocásticos com exemplos de ambos.

De acordo com os trabalhos analisados é possível notar combinações de várias estratégias que visam melhorar a qualidade no reconhecimento de padrões. No desenvolvimento deste trabalho foi possível observar o esforço em oferecer estratégias computacionais com alto desempenho que atendam as expectivas dos biólogos. Portanto, acredita-se que com as estratégias utilizadas no presente trabalho foi possível oferecer uma hibridização capaz de criar padrões com qualidade biológica.

Com a junção do algoritmo bioinspirado (CLONALG) com o algoritmo estocástico (MMO), foi possível extrair o que há de melhor em ambos, os padrões gerados correspondem aos padrões encontrados por D'ANGELO; PALMIERI (2020). Por fim, fica evidente que as estratégias utilizadas servem de fonte de estudos para trabalhos futuros, que agora podem contar com a abordagem CLONALG, ainda pouco explorada no contexto de Bioinformática, mas que por meio deste trabalho verificou-se a sua eficiência.

5.1 Trabalhos Futuros

Pretende-se validar a hipótese de utilizar uma outra abordagem de mutação, para tentar aumentar a convergência do algoritmo e consequentemente diminuir o tempo de execução. Um outro objetivo é testar o algoritmo com o parâmetro n2 inversamente proporcional à quantidade de iterações, visto que na seção 4 ao avaliar o gráfico da Figura 4.1 é possível notar um comportamento interessante dos dados, em que há uma leve tendência de diminuição do tempo de execução.

Outro ponto que será abordado é a paralelização do processo de reconhecimento de padrões, pois não há dependência de dados durante as iterações, ou seja, é possível isolar cada subsequência com o conjunto de sequências de entrada, e dessa forma é possível avaliar várias subsequências ao mesmo tempo.

Referências

ABDO, Z.; GOLDING, G. B. A step toward barcoding life: a model-based, decision-theoretic method to assign genes to preexisting species groups. *Systematic biology*, Oxford University Press, v. 56, n. 1, p. 44–56, 2007.

AFTABUDDIN, M.; KUNDU, S. Hydrophobic, hydrophilic, and charged amino acid networks within protein. *Biophysical journal*, Elsevier, v. 93, n. 1, p. 225–231, 2007.

ALBERTS, B.; JOHNSON, A.; LEWIS, J.; MORGAN, D.; RAFF, M.; ROBERTS, K.; WALTER, P.; WILSON, J.; HUNT, T. *Biologia molecular da célula*. [S.l.]: Artmed Editora, 2010.

ALTMAN, R. B.; DUGAN, J. M. Defining bioinformatics and structural bioinformatics. *Structural Bioinformatics*, Wiley Online Library, v. 44, p. 1–14, 2003.

ALVES, R. T.; DELGADO, M.; FREITAS, A. A. Knowledge discovery with artificial immune systems for hierarchical multi-label classification of protein functions. In: IEEE. *International Conference on Fuzzy Systems*. [S.l.], 2010. p. 1–8.

ANKERST, M.; BREUNIG, M. M.; KRIEGEL, H.-P.; SANDER, J. Optics: ordering points to identify the clustering structure. v. 28, n. 2, p. 49–60, 1999.

BEGUM, N.; ULANOVA, L.; WANG, J.; KEOGH, E. Accelerating dynamic time warping clustering with a novel admissible pruning strategy. In: ACM. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. [S.l.], 2015. p. 49–58.

BRAZMA, A.; JONASSEN, I.; EIDHAMMER, I.; GILBERT, D. Approaches to the automatic discovery of patterns in biosequences. *Journal of computational biology*, v. 5, n. 2, p. 279–305, 1998.

CASTRO, L. N. D. *Fundamentals of natural computing: basic concepts, algorithms, and applications*. [S.l.]: Chapman and Hall/CRC, 2006.

D'ANGELO, G.; PALMIERI, F. Discovering genomic patterns in sars-cov-2 variants. *International Journal of Intelligent Systems*, Wiley Online Library, v. 35, n. 11, p. 1680–1698, 2020.

DASGUPTA, D.; YU, S.; NINO, F. Recent advances in artificial immune systems: models and applications. *Applied Soft Computing*, Elsevier, v. 11, n. 2, p. 1574–1587, 2011.

DILL, K. A.; BROMBERG, S.; YUE, K.; CHAN, H. S.; FTEBIG, K. M.; YEE, D. P.; THOMAS, P. D. Principles of protein folding—a perspective from simple exact models. *Protein science*, Wiley Online Library, v. 4, n. 4, p. 561–602, 1995.

EDDY, S. R. Hidden markov models. *Current opinion in structural biology*, Elsevier, v. 6, n. 3, p. 361–365, 1996.

ESTER, M.; KRIEGEL, H.-P.; SANDER, J.; XU, X. et al. A density-based algorithm for discovering clusters in large spatial databases with noise. v. 96, n. 34, p. 226–231, 1996.

FEFELOVA, I.; FEFELOV, A.; VORONENKO, M.; KORNELYUK, A.; SACHENKO, A.; RYZHKOV, E.; LYTVYNENKO, V. Predicting the protein tertiary structure by hybrid clonal selection algorithms on 3d square lattice. In: IEEE. *2020 IEEE 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET)*. [S.l.], 2020. p. 965–968.

FRANCIS, Z.; VILLAGRASA, C.; CLAIRAND, I. Simulation of dna damage clustering after proton irradiation using an adapted dbscan algorithm. *Computer methods and programs in biomedicine*, Elsevier, v. 101, n. 3, p. 265–270, 2011.

HERBERT, J. P.; YAO, J. A granular computing framework for self-organizing maps. *Neurocomputing*, Elsevier, v. 72, n. 13-15, p. 2865–2872, 2009.

HINNEBURG, A.; KEIM, D. A. et al. An efficient approach to clustering in large multimedia databases with noise. v. 98, p. 58–65, 1998.

HIRST, J. D. The evolutionary landscape of functional model proteins. *Protein Engineering*, Oxford University Press, v. 12, n. 9, p. 721–726, 1999.

HORN, J.; GOLDBERG, D. E.; DEB, K. Implicit niching in a learning classifier system: Nature's way. *Evolutionary Computation*, MIT Press, v. 2, n. 1, p. 37–66, 1994.

KHURI, S. A bioinformatics track in computer science. In: ACM. *ACM SIGCSE Bulletin*. [S.l.], 2008. v. 40, n. 1, p. 508–512.

KUCHEROV, G. Evolution of biosequence search algorithms: a brief survey. *Bioinformatics*, Oxford University Press, v. 35, n. 19, p. 3547–3552, 2019.

LANGFELDER, P.; ZHANG, B.; HORVATH, S. Defining clusters from a hierarchical cluster tree: the dynamic tree cut package for r. *Bioinformatics*, Oxford University Press, v. 24, n. 5, p. 719–720, 2007.

LIBERATO, L. P. *Reconhecimento de padrões embiossequências utilizando sistema imunológico artificial*. [S.l.], 2021. Disponível em: <http://www.abntex.net.br/>.

LUO, W.; LIN, X.; ZHU, T.; XU, P. A clonal selection algorithm for dynamic multimodal function optimization. *Swarm and Evolutionary Computation*, Elsevier, v. 50, p. 100459, 2019.

MAJI, P.; PAUL, S. *Scalable Pattern Recognition Algorithms*. [S.l.]: Springer, 2016.

MOUNT, D. W. *Bioinformatics: sequence and genome analysis. 2nd*. [S.l.]: Cold Spring Harbor, NY: Cold Spring Harbor Laboratory Press. xii, 2004. v. 692.

PAL, S. K.; RAY, S. S.; GANIVADA, A. *Granular neural networks, pattern recognition and bioinformatics*. [S.l.]: Springer, 2017.

PANUCCIO, A.; BICEGO, M.; MURINO, V. A hidden markov model-based approach to sequential data clustering. In: SPRINGER. *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*. [S.l.], 2002. p. 734–743.

PAOLANTI, M.; FRONTONI, E. Multidisciplinary pattern recognition applications: A review. *Computer Science Review*, Elsevier, v. 37, p. 100276, 2020.

PROSDOCIMI, F.; COUTINHO, G.; NINNECW, E.; SILVA, A. F.; REIS, A. N. dos; MARTINS, A. C.; SANTOS, A. C. F. dos; JÚNIOR, A. N.; FILHO, F. C. Bioinformática: manual do usuário. *Biotecnologia Ciência & Desenvolvimento*, v. 29, p. 12–25, 2002.

RAFIEI, M.; NIKNAM, T.; KHOOBAN, M. H. Probabilistic electricity price forecasting by improved clonal selection algorithm and wavelet preprocessing. *Neural Computing and Applications*, Springer, v. 28, n. 12, p. 3889–3901, 2017.

RIDDER, D. de; RIDDER, J. de; REINDERS, M. J. Pattern recognition in bioinformatics. *Briefings in bioinformatics*, Oxford University Press, v. 14, n. 5, p. 633–647, 2013.

ROZAS, J.; FERRER-MATA, A.; SÁNCHEZ-DELBARRIO, J. C.; GUIRAO-RICO, S.; LIBRADO, P.; RAMOS-ONSINS, S. E.; SÁNCHEZ-GRACIA, A. Dnasp 6: Dna sequence polymorphism analysis of large data sets. *Molecular biology and evolution*, Oxford University Press, v. 34, n. 12, p. 3299–3302, 2017.

SMITH, R. F.; SMITH, T. F. Automatic generation of primary sequence patterns from sets of related protein sequences. *Proceedings of the National Academy of Sciences*, National Acad Sciences, v. 87, n. 1, p. 118–122, 1990.

SOTIROPOULOS, D. N.; TSIHRINTZIS, G. A. *Machine Learning Paradigms: Artificial Immune Systems and Their Applications in Software Personalization*. [S.l.]: Springer, 2016. v. 118.

SUN, Y.; BUHLER, J. Designing patterns for profile hmm search. *Bioinformatics*, Oxford University Press, v. 23, n. 2, p. e36–e43, 2007.

VIDAKI, A.; BALLARD, D.; ALIFERI, A.; MILLER, T. H.; BARRON, L. P.; COURT, D. S. Dna methylation-based forensic age prediction using artificial neural networks and next generation sequencing. *Forensic Science International: Genetics*, Elsevier, v. 28, p. 225–236, 2017.

WATSON, J. D.; BAKER, T. A.; BELL, S. P.; GANN, A.; LEVINE, M.; LOSICKE, R. *Biologia molecular do gene*. [S.l.]: Artmed Editora, 2015.

WATSON, J. D.; CRICK, F. H. et al. Molecular structure of nucleic acids. *Nature*, v. 171, n. 4356, p. 737–738, 1953.

YAVUZ, B. Ç.; YURTAY, N.; OZKAN, O. Prediction of protein secondary structure with clonal selection algorithm and multilayer perceptron. *IEEE Access*, IEEE, v. 6, p. 45256–45261, 2018.

YU, S.-Z. Hidden semi-markov models. *Artificial intelligence*, Elsevier, v. 174, n. 2, p. 215–243, 2010.

ZAFALON, G.; VISOTAKY, J.; AMORIM, A.; VALÊNCIO, C.; NEVES, L.; SOUZA, R. D.; MACHADO, J. A parallel approach of coffee objective function to multiple sequence alignment. In: IOP PUBLISHING. *Journal of Physics: Conference Series*. [S.l.], 2015. v. 633, n. 1, p. 012084.

ZAHA, A.; FERREIRA, H. B.; PASSAGLIA, L. M. *Biologia Molecular Básica-5*. [S.l.]: Artmed Editora, 2014.