UNIVERSIDADE FEDERAL DE MINAS GERAIS
Instituto de Ciências Biológicas
Programa de Pós-Graduação em Bioinformática

Lucas Moraes dos Santos

REDES CONVOLUCIONAIS NA IDENTIFICAÇÃO DE ALTERAÇÕES SUTIS EM MAPAS DE DISTÂNCIAS DE CONFORMAÇÕES DE DINÂMICA

BELO HORIZONTE
2022

Lucas Moraes dos Santos

REDES CONVOLUCIONAIS NA IDENTIFICAÇÃO DE ALTERAÇÕES SUTIS EM MAPAS DE DISTÂNCIAS DE CONFORMAÇÕES DE DINÂMICA

Dissertação apresentada ao Programa de Pós-Graduação em Bioinformática do Instituto de Ciências Biológicas da Universidade Federal de Minas Gerais como requisito parcial à obtenção do grau de Mestre em Bioinformática.

Orientadora: Profa. Dra. Raquel Cardoso de Melo Minardi

Belo Horizonte

2022

043 Santos, Lucas Moraes dos.
Redes convolucionais na identificação de alterações sutis em mapas de distâncias de conformações de dinâmica [manuscrito] / Lucas Moraes dos Santos. – 2022.
95 f. : il. ; 29,5 cm.

Orientadora: Profa. Dra. Raquel Cardoso de Melo Minardi.
Dissertação (mestrado) – Universidade Federal de Minas Gerais, Instituto de Ciências Biológicas. Programa de Pós-Graduação em Bioinformática.

1. Bioinformática. 2. Convolucões (Matemática). 3. Conformação Molecular. 4. Redes Neurais (Computação). 5. Aprendizado Profundo. I. Minardi, Raquel Cardoso de Melo. II. Universidade Federal de Minas Gerais. Instituto de Ciências Biológicas. III. Título.

CDU: 573.42

Ficha catalográfica elaborada pela bibliotecária Fabiane C M Reis – CRB 6 - 2680

UNIVERSIDADE FEDERAL DE MINAS GERAIS
Instituto de Ciências Biológicas
Programa Interunidades de Pós-Graduação em Bioinformática da UFMG

FOLHA DE APROVAÇÃO

"Redes convolucionais na identificação de alterações sutis em mapas de distâncias de conformações de dinâmica"

Lucas Moraes dos Santos

Dissertação aprovada pela banca examinadora constituída pelos Professores:

Profa. Raquel Cardoso de Melo Minardi - Orientadora
Universidade Federal de Minas Gerais

Prof. Marcelo da Silva Reis
Universidade Estadual de Campinas

Prof. Lucas Bleicher
Universidade Federal de Minas Gerais

Belo Horizonte, 20 de outubro de 2022.

Documento assinado eletronicamente por **Marcelo da Silva Reis**, **Usuário Externo**, em 20/10/2022, às 19:30, conforme horário oficial de Brasília, com fundamento no art. 5º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Lucas Bleicher**, **Professor do Magistério Superior**, em 24/10/2022, às 17:06, conforme horário oficial de Brasília, com fundamento no art. 5º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Raquel Cardoso de Melo Minardi**, **Professora do Magistério Superior**, em 27/10/2022, às 12:11, conforme horário oficial de Brasília, com fundamento no art. 5º do Decreto nº 10.543, de 13 de novembro de 2020.

A autenticidade deste documento pode ser conferida no site https://sei.ufmg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **1842829** e o código CRC **69E1DD87**.

Referência: Processo nº 23072.262435/2022-11 SEI nº 1842829

Agradecimentos

Primeiramente, agradeço a Deus pela vida, proteção e saúde, permitindo que meus sonhos se tornassem conquistas, ao longo de toda minha trajetória de vida.

A minha mãe Deolinda, meu pai Joceli, minha avó Carlinda (*in memoriam*) e a meus irmãos, pelo apoio e incentivo incondicionais.

A Universidade Federal de Minas Gerais (UFMG), instituição na qual sempre sonhei estudar, pela excelência no ensino e infraestrutura de qualidade à pesquisa.

Às agências de fomento: Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG) e Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

Aos Programas de Pós-Graduação em Bioinformática e Ciência da Computação da UFMG, por me proporcionarem uma formação de caráter multidisciplinar. Estendo o agradecimento a meus saudosos professores, pelo conhecimento transmitido a mim.

Ao Prof. Dr. Leonardo Henrique Franca de Lima da Universidade Federal de São João del-Rei (UFSJ), por contribuir com as análises de impacto produzido pelas mutações na proteína alvo, e compreensão dos dados de dinâmica molecular.

A José Gutembergue Mendonça e Yan Gerônimo, do Laboratório de Química Quântica Computacional (LQQC), coordenado pelo Prof. Dr. Gerd Rocha da Universidade Federal da Paraíba (UFPB), pelo compartilhamento das trajetórias de DM.

Aos estimados colegas do Laboratório de Bioinformática e Sistemas (LBS) da UFMG, grupo de pesquisa que tenho prazer em integrar, pelas presenças em apresentações e prévias de qualificação e defesa, sempre contribuindo com sugestões relevantes ao aprimoramento do trabalho.

Por fim, agradeço à minha orientadora Profa. Dra. Raquel Cardoso de Melo Minardi, pela confiança e oportunidade em poder aprender com uma das principais referências da pesquisa em Bioinformática e IA no Brasil; por cada ideia e instigação propostas, possibilitando a evolução do trabalho e, o aprimoramento da minha formação como pesquisador; e, não menos importante, pela *didática* ao ensinar e *empenho* à pesquisa, os quais servem de modelo à minha presente, e futura, carreira acadêmica.

"Se eu vi mais longe,
foi por estar sobre ombros de gigantes."
(Isaac Newton)

Resumo

Uma forma de compreender os efeitos de mutações na mobilidade natural de uma estrutura, é a partir de simulações de dinâmica molecular. Essa abordagem tem sido empregada no estudo de variantes do SARS-CoV-2, em especial, àquelas que tendem a desenvolver uma maior afinidade de ligação ao receptor humano.

Devido ao volume de informação gerada, as trajetórias de DM são consideradas como dados em larga escala, sendo sua aquisição (e posterior análise) comumente associada a um elevado custo computacional. Ainda assim, essas trajetórias revelam mudanças conformacionais sutis inerentes às estruturas, criando um cenário propício às técnicas de aprendizagem profunda.

Nesse sentido, o presente trabalho propõe avaliar o uso de redes convolucionais na previsão do impacto de mutações, através de dados obtidos de simulações de dinâmica molecular representados como mapas de distâncias. Esses dados correspondem a região da proteína S conhecida como Domínio de Ligação ao Receptor, onde é comum a ocorrência de mutações características a variantes do SARS-CoV-2. Desse modo, o modelo é capaz de predizer se o produto de uma nova simulação, referente à uma estrutura mutada, seria mais afim ao receptor humano e, consequentemente, mais infectante.

Com base nos resultados da precisão do modelo, nos diferentes sistemas da base de dados, foi possível correlacionar às mudanças de energia livre de ligação para diferentes mutações, sendo estimado um ρ igual a 0,77. Esse valor indica que as predições do modelo acompanham resultados recentemente publicados, no que tange a afinidade, ou neutralidade, de mutações encontradas no RBD da proteína S. Por fim, também é possível observar quais regiões do mapa de distâncias, estariam contribuindo ao aprendizado e à decisão do modelo, através do mapa de recursos correspondente.

Palavras-chave: Mudanças Conformacionais, Mapas de Distâncias, Redes Neurais Convolucionais.

Abstract

One way to understand the effects of mutations on the natural mobility of a structure is through molecular dynamics simulations. This approach has been used in the study of SARS-CoV-2 variants, especially those that tend to develop a higher binding affinity for the human receptor.

Due to the volume of information generated, DM trajectories are considered as large-scale data, and their acquisition (and subsequent analysis) is commonly associated with a high computational cost. Even so, these trajectories reveal subtle conformational changes inherent to the structures, creating a favorable scenario for deep learning techniques.

In this sense, the present work proposes to evaluate the use of convolutional networks in predicting the impact of mutations, through data obtained from molecular dynamics simulations represented as distance maps. These data correspond to the region of the S protein known as Receptor Binding Domain, where mutations characteristic of SARS-CoV-2 variants are common. In this way, the model is able to predict whether the product of a new simulation, referring to a mutated structure, would have a greater affinity with the human receptor and, consequently, be more infective.

Based on the results of the model's precision, to the different systems of the database, it was possible to correlate the changes in binding free energy for different mutations, with an estimated ρ equal to 0.77. This value indicates that the model's predictions follow recently published results, regarding the affinity, or neutrality, of mutations found in the RBD of the S protein. Finally, it is also possible to observe which regions of the distance map would be contributing to the learning and to the decision of the model, through the corresponding resource map.

Keywords: Conformational Changes, Distance Maps, Convolutional Networks.

Lista de Figuras

Lista de Tabelas

Lista de Acrônimos

2019-nCoV *2019 novel coronavirus*.

Acc Acurácia.

ACE2 *Angiotensin-Converting Enzyme 2*.

BFE *Binding Free Energy*.

BN *Batch Normalization*.

BS *Batch Size*.

CNN *Convolutional Neural Networks*.

ConvNet *Convolutional Networks*.

COVID-19 *Coronavirus Disease*.

DL *Deep Learning*.

DM Dinâmica Molecular.

DNN *Deep Neural Networks*.

Esp Especificidade.

EV *Emerging Variants*.

FC *Fully Connected*.

FPR *False Positive Rate*.

hACE2 *human Angiotensin-Converting Enzyme 2*.

ILSVRC *ImageNet Large Scale Visual Recognition Challenge*.

MD Mapas de Distâncias.

MLP *Multilayer Perceptron*.

MM Mecânica Molecular.

Prec Precisão.

RBD *Receptor Binding Domain*.

RBM *Receptor Binding Motif*.

Rec Revocação.

ReLU *Rectified Linear Unit*.

RLROP *Reduce Learn Rate On Plateau*.

RMSD *Root Mean Square Deviation*.

RMSF *Root Mean Square Fluctuation*.

RNA *Ribonucleic Acid*.

ROC *Receiver Operating Characteristic Curve*.

S ***S****pike* protein.

SARS-CoV-2 *Severe Acute Respiratory Syndrome Coronavirus-2*.

SBC Sociedade Brasileira de Computação.

TNR *True Negative Rate*.

TPR *True Positive Rate*.

VGGNet *Visual Geometry Group Network*.

VOC *Variants of Concern*.

VOI *Variants of Interest*.

WT *Wild Type*.

Sumário

Capítulo 1

Introdução

As proteínas desempenham um papel fundamental no metabolismo celular, sendo responsáveis por diversas funções, como: catálise de processos metabólicos, respiração, expressão gênica, comunicação celular, formação de estruturas intra e extracelulares, reconhecimento molecular, defesa do organismo, e transdução de sinais intracelulares [Kessel & Ben-Tal, 2018, p. 6–20]. Esses processos encontram-se presentes na maioria dos organismos, sendo possível traçar uma relação direta com a diversidade de proteínas na natureza. Essa relação implica que a evolução das proteínas acompanhou a evolução de espécies na Terra ao longo dos anos [Stryer et al., 2002, p. 194].

1.1 SARS-CoV-2

A *pandemia de COVID-19*, declarada pela Organização Mundial de Saúde (OMS) em 11 de março de 2020, corresponde a uma pandemia em curso de uma doença respiratória, causada pelo *coronavírus da síndrome respiratória aguda grave 2* (SARS-CoV-2). O SARS-CoV-2 é um betacoronavírus (β-CoV) da família *Coronaviridae* com genoma viral de RNA [Guruprasad, 2021]. A infecção pelo SARS-CoV-2 ocorre através do reconhecimento do receptor humano pela glicoproteína espicular S (do inglês, *spike* S), localizada no envelope viral [Wu et al., 2020; Zhang et al., 2021]. Sua função é permitir a infecção do vírus, possibilitada devido a afinidade de seu Domínio de Ligação ao Receptor (*Receptor Binding Domain* - RBD) com a Enzima Conversora de Angiotensina II (*Angiotensin-Converting Enzyme 2* - ACE2/hACE2), encontrada nas células epiteliais alveolares tipo II [Lan et al., 2020].

As interações intermoleculares que formam o complexo RBD-ACE2 estão relacionadas à afinidade de ligação das proteínas virais ao receptor humano. Assim, a região do RBD da proteína S, que contém os resíduos de contato com o hACE2, torna-se propícia ao

surgimento de mutações críticas, com o potencial de aumentar a afinidade ligante-receptor [Sanches et al., 2021; McCallum et al., 2022], ou ainda a evasão do sistema imune [Harvey et al., 2021]. Trabalhos recentes sugerem uma relação dessas mutações com infectividade do SARS-CoV-2, uma vez que elas ocorrem em regiões de interação da proteína S com o receptor hACE2 às quais, substituições de aminoácidos tendem a produzir mudanças na mobilidade natural da estrutura, bem como uma maior afinidade com o receptor humano [Luan et al., 2021; Verma & Subbarao, 2021; Tian et al., 2021; Lupala et al., 2022].

Em seu trabalho, Verma & Subbarao [2021] analisam o impacto de mutações no RBD da proteína S e seus efeitos na estabilidade do complexo ligante-receptor. Algumas mutações, como a S:N501Y, estão associadas a mudanças conformacionais na estrutura do complexo [Verma & Subbarao, 2021]. Nesse sentido, a compreensão dessas interações tem sido alicerce para o desenvolvimento de pesquisas focadas no desenvolvimento racional de vacinas à COVID-19, capazes de induzir anticorpos que neutralizem a ligação do RBD ao receptor humano [Lan et al., 2020].

1.2 Simulações de Dinâmica Molecular

A adoção de métodos de simulação computacional, como a Dinâmica Molecular (DM), tem proporcinado uma maior compreenssão dessas interações em nível intermolecular [Alder & Wainwright, 1957]. A DM caracteriza-se por um método que consiste em resolver equações de movimento, de forma simultânea, para centenas de partículas pertencentes a um sistema. Como resultado, obtém-se uma trajetória que especifica como as posições das partículas do sistema variam no tempo [Leach, 2001, p. 353–355]. Assim, simulações de DM têm se destacado na análise das propriedades conformacionais de sistemas moleculares [Alder & Wainwright, 1957; Leach, 2001, p. 392–394], sendo recentemente empregadas no entendimento das interações intermoleculares no complexo RBD-ACE2 [Gobeil et al., 2021; Verma & Subbarao, 2021].

1.3 Mapas de Distâncias

Uma representação utilizada para descrever a estrutura de uma proteína, ou ainda a conformação de uma molécula, é a *matriz de distâncias* [Kloczkowski et al., 2009; Leach, 2001, p. 467–474]. Uma matriz de distâncias, $\mathbf{d} = (d_{ij})$, é obtida a partir do cálculo da distância entre o i-ésimo e o j-ésimo resíduo. Usualmente, a distância é medida entre os átomos de $C\alpha$ (carbono-α) dos resíduos [Kloczkowski et al., 2009; Wang et al., 2017]. Soluções recentes em predição de estrutura de proteínas têm utilizado representações 2D

(imagens) dessas matrizes, conhecidas por *mapas de distâncias* [Anishchenko et al., 2021; Jumper et al., 2021]. Na maioria das aplicações, esses mapas são empregados como uma *feature*, na construção de modelos à predição de estrutura de proteínas [Gao et al., 2020; Torrisi et al., 2020].

1.4 Inteligência Artificial

Embora o termo *inteligência artificial* (IA) tenha sido cunhado pela primeira vez em 1955, pelo matemático americano John McCarthy, a história revela que as diferentes definições do termo, buscavam mensurar o sucesso de um sistema na execução de tarefas, em termos de fidelidade a *performance humana* ou ainda, a *racionalidade* [Russell & Norvig, 2010]. Kurzweil [1990] define a inteligência artificial como "*a arte de criar máquinas que executam funções que exigem inteligência quando executadas por pessoas*".

No campo da *biologia molecular* a IA tem sido empregada há mais de uma década, impulsionada pelo crescente volume de dados biológicos advindos das ômicas, ou ainda de imagens e sinais biomédicos [Min et al., 2017]. Abordagens recentes baseadas em aprendizado profundo (do inglês, *deep learning*, DL), têm se destacado na solução de problemas em aberto em biologia estrutural, como o enovelamento de proteínas [Alquraishi, 2019; Senior et al., 2020; Anishchenko et al., 2021]. Na mesma linha, aplicações como o Rosetta [Yang et al., 2020] e, mais recentemente, o AlphaFold2 [Jumper et al., 2021] tornaram-se o estado da arte no que tange ao design *de novo* [Huang et al., 2016b], ou ainda a modelagem por homologia, obtendo performances significativas na *modelagem* de estruturas de proteínas [Alquraishi, 2019]. Assim, verifica-se que a IA não se limita apenas ao entendimento, mas também a *construção* de entidades inteligentes, conforme observado por Russell & Norvig [2010].

Um aspecto que tem contribuído ao desempenho dessas aplicações é o uso de algoritmos de DL, como as *Redes Convolucionais* [LeCun et al., 1989]. Nesse caso, sucessivas camadas convolucionais formam redes neurais densas capazes de identificar padrões complexos em imagens. Em alusão a seu nome, redes convolucionais utilizam uma operação linear denominada *convolução* na obtenção dos pesos da rede, ao invés da multiplicação matricial, usada em abordagens tradicionais de redes neurais [Goodfellow et al., 2016, p. 326–328]. A arquitetura de uma rede convolucional consiste em camadas de extração de características da imagem e classificação [LeCun et al., 1989].

As redes convolucionais têm obtido melhorias graduais e contínuas na predição de proteínas, tornando-se um dos algoritmos de DL mais aplicados ao problema, comparado a outras técnicas em DL [Torrisi et al., 2020; Gao et al., 2020]. Isso deve-se a sua capacidade

de analisar informações espaciais [Min et al., 2017], o que possui relação com propriedades específicas de sua arquitetura.

1.5 Questão de Pesquisa

Conforme descrito na subseção 1.1, trabalhos recentes sugerem que a ocorrência de mutações na proteína S, com destaque à região de interação entre o RBD e o receptor humano (ou a anticorpos), tendem a produzir *mudanças conformacionais sutis* e locais. Como consequência, é possível observar variações na estabilidade do complexo RBD-hACE2 ou ainda, o ganho de afinidade com o receptor. Uma maneira de analisar essas mudanças nos padrões de movimento intrínsecas às estruturas, é a partir de *simulações de dinâmica molecular*.

Devido ao volume de informação gerada, as trajetórias provenientes de simulações de dinâmica molecular podem ser consideradas como dados em *larga escala*, estando sua obtenção comumente associada a um alto custo computacional [Durrant & McCammon, 2011]. A título de exemplo, em um cenário onde é necessário prever o impacto de mutações na afinidade de ligação em complexos proteína-ligante, a princípio, cada mutação requisitaria uma nova simulação do sistema. Isso tornaria o processo custoso computacionalmente, devido a combinação de mutações possíveis.

Contudo, essas trajetórias revelam propriedades conformacionais intrínsecas as estruturas, criando um cenário propício ao uso de aprendizado profundo. Os dados que compõem as trajetórias de dinâmica molecular, podem ser transformados em representações 2D análogas a estrutura, como *mapas de distâncias*. Desse modo, pretende-se avaliar se *redes convolucionais* (uma classe de rede neural profunda especializada no processamento de imagens) são capazes de prever o impacto de mutações, a partir de mudanças conformacionais sutis em dados de dinâmica molecular, representadas por mapas de distâncias. Nesse caso, o modelo obtido seria capaz de prever se, o produto de uma nova simulação, seria mais afim ao receptor,de acordo com a mutação realizada.

1.6 Objetivos

1.6.1 Objetivos gerais

O objetivo do presente estudo é o desenvolvimento de um modelo baseado em aprendizado profundo, para a identificação de mudanças conformacionais relacionadas a mutações pontuais características a variantes do SARS-CoV-2.

1.6.2 Objetivos específicos

- Revisão de literatura com foco em fundamentos e trabalhos relacionados
- Coletar e integrar as trajetórias obtidas através de simulações de dinâmica molecular com a proteína alvo
- Gerar os mapas de distâncias [Kloczkowski et al., 2009], que serão entrada para a rede neural convolucional
- Desenvolver o modelo de aprendizado profundo observando práticas metodológicas consolidadas como, a análise de modelos de *baseline*, seleção e otimização dos hiperparâmetros, regularização, técnicas de validação e, métricas de avaliação [Goodfellow et al., 2016, p. 409]
- Implementar o modelo de aprendizado profundo utilizando bibliotecas de redes neurais atuais e arquiteturas constantes na literatura
- Avaliar a capacidade preditiva dos modelos bem como discutir a semântica dos resultados obtidos em termos dos padrões de interação molecular
- Desenvolver estratégias capazes de identificar qual(is) regiões/resíduos da estrutura que contribuem à decisão do modelo

1.7 Estrutura do Trabalho

O presente trabalho encontra-se organizado da seguinte forma:

- O capítulo 1 tem por objetivo introduzir uma breve contextualização para apresentação da questão de pesquisa.
- O capítulo 2 discorre sobre a fundamentação teórica da pesquisa, sendo abordados conceitos relacionados à estrutura das proteínas (seção 2.1), dinâmica molecular (seção 2.2), SARS-CoV-2 (seção 2.3), processamento de imagens (seção 2.4) e redes convolucionais (seção 2.5).
- No capítulo 3 é abordada a metodologia utilizada no desenvolvimento da pesquisa. Serão descritos aspectos relacionados à representação das trajetórias de dinâmica molecular e, implementação do modelo baseado em aprendizado profundo.
- O capítulo 4 apresenta os resultados alcançados e discussões sobre o significado biológico desses resultados.

- Por fim, no capítulo 5, são extraídas conclusões sobre os resultados apresentados, com base na questão de pesquisa, além das principais contribuições e perspectivas do trabalho.

Capítulo 2

Fundamentação Teórica

2.1 Estrutura de Proteínas

2.1.1 Composição hierárquica

As características constitutivas das proteínas revelam uma complexidade estrutural que costuma ser organizada hierárquicamente em três níveis principais: estrutura *primária*, *secundária*, e *terciária*. A estrutura primária corresponde ao arranjo de aminoácidos[1] que compõe a cadeia polipeptídica [Kessel & Ben-Tal, 2018, p. 67]. A partir da estrutura primária é possível o entendimento de propriedades biológicas relevantes, pois a sequência de aminoácidos contém toda a informação necessária à proteína enovelar-se em uma estrutura tridimensional única [Kessel & Ben-Tal, 2018, p. 103], determinando a forma final da proteína e, a sua função [Kessel & Ben-Tal, 2018, p. 73].

Ao longo da cadeia polipeptídica podem ser encontrados padrões regulares, caracterizados por segmentos de aminoácidos locais que enovelam-se assumindo formas simples semelhantes a folhas, *loops*, hélices, etc. Esses padrões caracterizam a estrutura secundária cujos elementos integram a disposição final da proteína, sendo que os mais comuns são as alfa-hélices (α-hélices) e folhas-beta (folhas β) [Nelson & Cox, 2004, p. 126]. Ainda assim, esses elementos são muito simples à execução de funções complexas, o que é alcançado com o enovelamento de toda a cadeia proteica [Kessel & Ben-Tal, 2018, p. 127]. Essa conformação é denominada estrutura terciária, comumente referida como a forma nativa da proteína, devido a sua estabilidade que possibilita a proteína ser biologicamente

[1] Cada aminoácido é composto por um átomo de carbono central (conhecido como carbono-α, $C\alpha$), um átomo de hidrogênio (H), um grupamento amino (NH_2^+), um grupamento carboxila ($COOH^-$), e uma cadeia lateral que distingue os aminoácidos [Kessel & Ben-Tal, 2018, p. 73]. Assim, quando incorporados a cadeia proteica, os aminoácidos são frequentemente referidos como *resíduos*.

ativa[2] [Kessel & Ben-Tal, 2018, p. 67].

2.1.2 Arquitetura de proteínas

De modo geral, a arquitetura das proteínas pode ser dividida em três níveis básicos: *motivos proteicos*, *enovelamentos complexos*, e *domínios*. Os motivos proteicos caracterizam-se por combinações de elementos secundários (na maioria α-hélices e folhas β) formando arranjos de tamanhos não maiores que 20 resíduos. É possível encontrar motivos formados apenas por α-hélices, ou folhas β, embora também hajam arranjos onde ambos elementos coexistam, conhecidos como *mistos* [Kessel & Ben-Tal, 2018, p. 132–139]. De modo análogo aos elementos secundários, também é possível encontrar combinações de motivos (especialmente os motivos β-α-β) constituindo enovelamentos mais complexos [Kessel & Ben-Tal, 2018, p. 139].

Por sua vez, os domínios são subestruturas encontradas na cadeia polipeptídica em tamanhos que variam de 100 a 250 resíduos, as quais podem enovelar-se de forma independente produzindo uma conformação compacta e estável [Alberts et al., 2002, p. 136]. Atualmente, essas estruturas têm sido consideradas como as unidades funcionais e evolucionárias básicas das proteínas, pois muitos domínios que possuem uma forma característica também têm uma função específica. Assim, diferentes domínios em uma proteína podem estar associados a múltiplas funções relacionadas à sinalização, regulação e ao reconhecimento molecular [Kessel & Ben-Tal, 2018, p. 147].

2.1.3 Relacionamento entre estrutura e função de uma proteína

Uma característica que está correlacionada à função de uma proteína é a sua estrutura. A natureza não-covalente das interações que ocorrem a partir das cadeias laterais dos aminoácidos possui papel relevante na conformação natural adquirida pela proteína, uma vez que são responsáveis por proporcionar estabilidade à estrutura. Assim, a cadeia polipeptídica ajusta-se no espaço assumindo uma forma que acaba por caracterizar sua configuração funcional, adquirindo a flexibilidade necessária ao exercício da função biológica. Esse processo químico é denominado como *enovelamento proteico* [Alberts et al., 2002, p. 109–116].

Até o final da década de 70, acreditava-se que o estado nativo de uma proteína era alcançado a partir das interações intermoleculares dos resíduos pertencentes a cadeia

[2]Uma vez que a estrutura tridimensional de uma proteína seja alterada de modo que ocorra a perda de sua função, costuma-se dizer que a proteína encontra-se em um *estado desnaturado*. Nesse caso, a estrutura não encontra-se totalmente desenovelada, podendo adquirir conformações parciais que não possuem capacidade de desempenhar qualquer função [Nelson & Cox, 2004, p. 50]

polipeptídica (hipótese termodinâmica). Consequentemente, isso implicaria na exploração do conjunto de todos as conformações possíveis à estrutura, dentre elas a nativa. Contudo, em 1968, Cyrus Levinthal apresenta uma objeção a essa abordagem, que ficaria conhecida como *Paradoxo de Levinthal*[3] [Kessel & Ben-Tal, 2018, p. 306], na qual ele sugere que o tempo à proteína atingir sua conformação natural, com base na hipótese termodinâmica, seria inexequível, considerando o período de tempo extremamente curto em que a maioria das proteínas presentes na natureza enovela-se.

Como alternativa a esse problema, surge a teoria da Superfície de Energia (do inglês *Energy Landscape Theory*) a qual fornece um entendimento ao processo de enovelamento com base no afunilamento da energia livre. De acordo com a hipótese termodinâmica, a cadeia polipeptídica pode encontrar-se em dois estados: não-enovelada e nativo. Porém, a teoria do funil de enovelamento propõe que as proteínas se enovelam de maneira cooperativa, transitando por *estados intermediários* de estabilidade estrutural onde cada estado limita as possíveis conformações futuras. À medida que a proteína enovela-se ocorre o decrescimento simultâneo de energia e entropia, até que a energia de conformação mínima da proteína seja alcançada (estado nativo) [Kessel & Ben-Tal, 2018, p. 314–315], como ilustrado na figura 2.1.

Cada proteína enovela-se de forma única, segundo requisitos básicos de sua estrutura terciária, possibilitando assim a execução de funções sofisticadas como: catálise de processos metabólicos, transdução de sinais, defesa, transferência de energia, reconhecimento molecular, etc. Além da *estabilidade*, mencionada acima, a *solubilidade* em meio aquoso e a habilidade de constituir *sítios ativos, ou de ligação* compõem esses requisitos estruturais [Kessel & Ben-Tal, 2018, p. 131]. A título de exemplo, a formação de complexos ligante-receptor apresenta grande especificidade uma vez que as proteínas tendem a ligar-se a apenas uma ou algumas moléculas, dentre milhares de tipos que encontra seu ambiente biológico [Alberts et al., 2002, p. 153].

Embora as proteínas alcancem o estado nativo em um curto intervalo de tempo, a estrutura enovelada ainda apresenta uma estabilidade relativa. Assim, a interação com outras moléculas, como os *ligantes*, tende a impactar a conformação natural e, inclusive,

[3]Com base na hipótese termodinâmica, a busca pelo estado nativo da proteína envolveria a exploração de todas as conformações possíveis da cadeia polipeptídica, considerando cada resíduo independentemente. Uma vez que o número mínimo de conformações por resíduo seja 2 (dois), então uma proteína contendo 100 resíduos possuiria 2^{100} conformações possíveis, dentre elas o estado nativo. Assumindo que o tempo à mudança conformacional seja de 1 picossegundo ($1ps = 10^{-12}$ s), seriam necessários 2^{200} ps, ou ainda, $1,3 \times 10^{10}$ anos, para a proteína explorar todo o espaço conformacional e encontrar o estado nativo. Porém, essa escala de tempo é da ordem de grandeza da idade do Universo, estimada em $1,4 \times 10^{10}$ anos. Assim, esse processo é inexequível considerando o período de tempo extremamente curto que a maioria das proteínas leva para se enovelar na vida real (entre milissegundos a 1(um) segundo) [Kessel & Ben-Tal, 2018, p. 306]

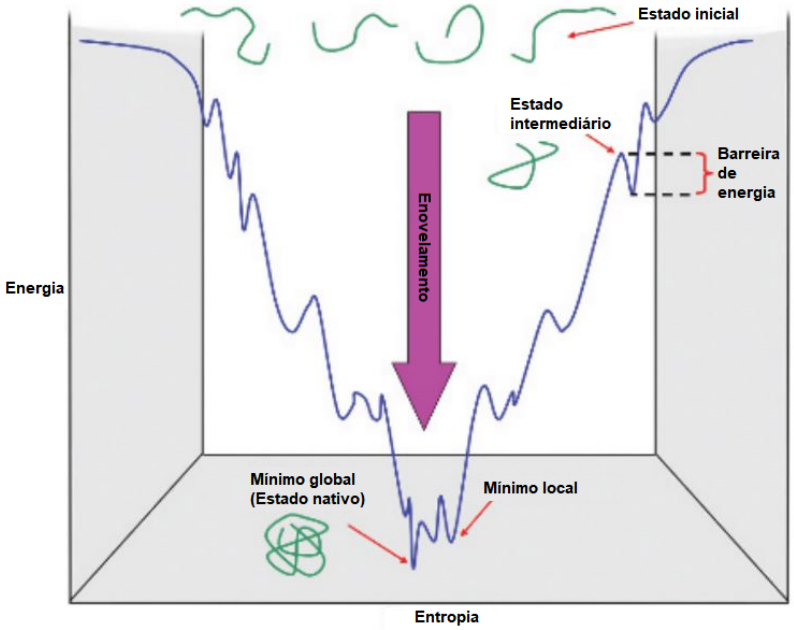


Figura 2.1. As curvas em *verde* representam a cadeia polipeptídica da proteína enovelada. A forma de funil da curva, em *azul*, reflete o decrescimento simultâneo de energia e entropia da proteína durante o processo de enovelamento, até alcançar o estado nativo. Fonte: Kessel & Ben-Tal [2018]

a função desempenhada pela proteína [Nelson & Cox, 2004, p. 159]. Quando os resíduos de uma proteína interagem com determinado ligante, a energia livre da estrutura tende a ser alterada. Isso implica em mudanças conformacionais capazes de impactar características físico-químicas inerentes ao sítio ativo, e com consequências diretas a função desempenhada pela proteína [Kessel & Ben-Tal, 2018, p. 343–345].

Por vezes, as mudanças conformacionais são significativas possuindo uma abrangência em nível estrutural, como a abertura, ou fechamento, de trímeros. Nesse caso, são comuns movimentos com larga amplitude atingindo alguns nanometros (nm), em alguns casos [Nelson & Cox, 2004, p. 159]. Porém, também são comuns mudanças conformacionais sutis refletindo em vibrações moleculares e pequenos movimentos de resíduos ao longo de um determinado segmento da cadeia polipeptídica. Ainda assim, essas conformações sutis são significativas o suficiente para alterar a afinidade de ligação da proteína [Kessel & Ben-Tal, 2018, p. 345].

Como mencionado anteriormente, os resíduos participam ativamente das interações entre as proteínas e as demais moléculas. Alguns resíduos, como os aromáticos, são prevalentes na ligação das proteínas com determinados ligantes, uma vez que realizam interações eletrostáticas conhecidas como interações π-π. Os anéis aromáticos da tirosina podem participar de interações cátion-π, onde suas nuvens de elétrons parcialmente negativas interagem eletrostaticamente com as cadeias laterais catiônicas de lisina ou ar-

ginina [Kessel & Ben-Tal, 2018, p. 97–98]. A planaridade dos anéis aromáticos tornam as interações *geometricamente dependentes*, o que confere especificidade. Outro exemplo desse tipo de interação são as ligações de hidrogênio, determinantes da especificidade da interação proteína-ligante [Kessel & Ben-Tal, 2018, p. 526].

2.2 Dinâmica Molecular

2.2.1 Conceitualização

Biofísicos tem buscado nos últimos anos formas de compreender o comportamento dinâmico de sistemas moleculares. Dentre algumas abordagens, uma, proposta em meados da década de 80, sugere descrever os sistemas moleculares a partir de aproximações das forças reais intrínsecas a eles, aplicando as leis da mecânica clássica newtoniana. Essa abordagem é conhecida como *Mecânica Molecular* (MM), uma vez que são geradas sucessivas configurações do sistema ao longo do tempo, possibilitando o estudo das múltiplas interações interatômicas dos diferentes resíduos que constituem a molécula [Kessel & Ben-Tal, 2018, p. 231–232].

Com o avanço da computação, especialmente no que tange à processamento e recursos físicos, tornou-se possível nos últimos anos a realização das modelagens de MM para sistemas mais complexos, a partir de técnicas de simulação computacional como a *Dinâmica Molecular* (DM). Simulações de DM concentram-se no cálculo dos movimentos das partículas (ex. átomos) pertencentes a um sistema, por um determinado período de tempo [Kessel & Ben-Tal, 2018, p. 238]. Como resultado, obtém-se uma *trajetória* que especifica como as posições e velocidades das partículas no sistema variam com o tempo [Leach, 2001, p. 353]. Essas simulações podem capturar uma ampla variedade de processos biomoleculares relevantes, incluindo mudanças conformacionais [Vijayakumar & Das, 2019; Gobeil et al., 2021; Verma & Subbarao, 2021], interações ligante-receptor [Chen et al., 2019a; Vora et al., 2019; Wingler et al., 2019] e enovelamento de proteínas [Strodel, 2021], revelando as posições de todos os átomos em resolução temporal de nano a femtossegundos [Hollingsworth & Dror, 2018].

A primeira simulação de DM foi realizada por Alder e Wainwright em 1957, que usaram um computador IBM 704 para simular colisões perfeitamente elásticas[4] entre esferas rígidas que representavam as interações atômicas [Alder & Wainwright, 1957]. A figura 2.2 apresenta um sistema de esferas rígidas, semelhante ao proposto por Alder e Wainwright, inicialmente em uma condição de equilíbrio. No momento da colisão entre

[4]Uma colisão elástica caracteriza-se por um encontro entre dois corpos em que a energia cinética e o momento linear do sistema se conservam [Halliday et al., 2013, p. 199]

as esferas, é possível observar a variação da energia do sistema, uma vez que os corpos adquirem velocidade (*princípio da conservação do momento linear*).

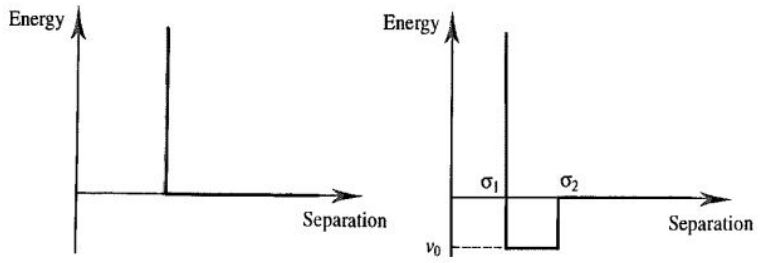


Figura 2.2. A energia do sistema é infinita abaixo da distância de *cutoff* σ_1 e zero acima da distância de *cutoff* σ_2; v_0 é a velocidade inicial das esferas após a colisão dos corpos. Fonte: Leach [2001]

A partir da segunda lei de Newton do movimento, uma partícula i de massa m_i submetida a ação de uma força F_i terá um deslocamento r_i no espaço, sob velocidade (v_i) ao longo do tempo (t) [Leach, 2001, p. 353]. Assim, a trajetória da partícula no sistema pode ser descrita a partir da seguinte equação diferencial:

$$F_i = m_i(\frac{\partial v_i}{\partial t}) = m_i(\frac{\partial^2 r_i}{\partial t^2}) \tag{2.1}$$

Uma vez que essa abordagem pode ser aplicada a todos os átomos do sistema, é possível inferir que as diversas forças interatômicas tendem a contribuir à *energia potencial total* do sistema (U_{total}). Assim, a U_{total} resulta do cálculo das múltiplas energias potenciais, correspondentes as diferentes interações (covalentes, eletrostáticas, van der Waals, etc) que os átomos participam [Kessel & Ben-Tal, 2018, p. 230–239]. Esse conceito, pode ser expresso a partir de um conjunto de equações matemáticas, conhecidas como *campos de forças*, conforme seguem abaixo

$$U_{total} = U_{cov} + U_{elet} + U_{apo} + U_{vdw}, \tag{2.2}$$

onde U_{cov} refere-se a energia potencial resultante das propriedades covalentes, U_{elet} é a energia potencial das interações eletrostáticas, U_{apo} é a energia potencial das interações apolares (não-polares), e U_{vdw} é a energia potencial das interações de van der Waals. A tabela 2.1 apresenta uma relação dos campos de forças para cada uma das interações interatômicas.

A partir da energia potencial total do sistema (U_{total}) é possível derivar a força (F_i) exercida sobre um átomo i, usando a equação de campo de força abaixo:

$$F_i = -\frac{\partial U_{total}}{\partial r_i} \tag{2.3}$$

Tabela 2.1. Equações de campos de força para as interações interatômicas

Interação	Energia Potencial	Parâmetros
Covalente	$\sum_{i=1}^{N_{bonds}} k_{b,i}(b_i - b_{0,i})^2$	• b_i : distância da ligação
Covalente	$\sum_{i=1}^{N_{angles}} k_{\theta,i}(\theta_i - \theta_{0,i})^2$	• θ_i : ângulo de valência
Covalente	$\sum_{i=1}^{N_{diedral}} k_{\phi,i}(\cos(n_i\phi_i - \phi_{0,i}))$	• ϕ_i : ângulo diedral
Covalente	$\sum_{i=1}^{N_{imprp-dbd}} k_{\omega,i}(\omega_i - \omega_{0,i})^2$	• ω_i : ângulo diedral impróprio
Van der Waals	$\sum_{i}^{N_{atoms}} \sum_{j\neq i}^{N_{atoms}} \left(\frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^{6}}\right)$	• r_{ij} : distância entre os átomos
Eletrostática	$\sum_{i}^{N_{atoms}} \sum_{j>i}^{N_{atoms}} \left(\frac{q_i q_j}{D r_{ij}}\right)$	• q : carga dos átomos i e j

A energia potencial de um sistema depende de fatores relacionados às interações covalentes, eletrostáticas, apolares, e de van der Waals. Em todos os casos, a interação covalente é retratada como uma comparação entre um valor associado ao átomo em uma coordenada (i.e., b_i ou θ_i) e um valor teórico em uma condição de equilíbrio (i.e., $b_{0,i}$ ou $\theta_{0,i}$). Os parâmetros $k_{b,i}$,$k_{\theta,i}$,$k_{\phi,i}$,$k_{\omega,i}$ referem-se a forças constantes das interações. A_{ij} e B_{ij} são parâmetros associados com as interações de van der Waals. Quanto à interação eletrostática, o parâmetro D corresponde ao dielétrico local e r_{ij} a distância entre os átomos i e j, com cargas q_i e q_j. Por fim, N_{bonds}, N_{angles}, $N_{imprp-dbd}$, e N_{atoms} correspondem ao total de interações, ângulos, e átomos, respectivamente [Kessel & Ben-Tal, 2018, p. 233].

Com isso, é possível concluir um importante conceito: *o movimento de cada átomo no sistema está diretamente relacionado à sua energia em um dado momento* [Kessel & Ben-Tal, 2018, p. 239]. Ou seja, supondo que seja aplicado um determinado *snapshot* do sistema, incluindo as coordenadas de todos os átomos em uma configuração estática, é possível obter a energia potencial total do sistema nessa configuração [Kessel & Ben-Tal, 2018, p. 232].

2.2.2 Mudanças conformacionais a partir de simulações de dinâmica molecular

Simulações de DM têm sido empregadas na obtenção de *insights* relevantes sobre as propriedades conformacionais dos sistemas moleculares, e como conformações estruturais evoluem no tempo [Chen et al., 2020a; Roy et al., 2020; S et al., 2020; Gobeil et al., 2021; Verma & Subbarao, 2021; Lupala et al., 2022]. Alguns padrões conformacionais revelam-se quando a dinâmica da estrutura é analisada em tempo contínuo. Assim, é comum abstrair a trajetória de DM representando-a como um "filme", onde os conjuntos de coordenadas (também conhecidos como *frames*) da estrutura são armazenados em intervalos de tempo

regulares. Esses *frames* podem ser exibidos em sequência, possibilitando a análise da mobilidade espacial das moléculas com uma resolução temporal muito fina [Leach, 2001, p. 392].

Uma trajetória de DM abrange o conjunto dos estados de um sistema molecular interligados no tempo, o que torna possível o cálculo de propriedades que possam ser correlacionas com o tempo [Leach, 2001, p. 374]. Ainda assim, embora alguns softwares de análise gráfica molecular sejam capazes de apresentar parâmetros estruturais em função do tempo, algumas mudanças estruturais específicas, como as rotações de ligação, criam um desafio à representação cartesiana[5]. Em muitas situações, essas rotações podem ocasionar mudanças conformacionais à estrutura [Leach, 2001, p. 392].

Na maioria dos sistemas reais, o movimento da estrutura molecular é "caótico", em virtude das múltiplas interações realizadas entre os resíduos. Contudo, também ocorrem movimentos simultâneos de baixa frequência, relacionados a mudanças conformacionais sutis mas significativas à função biológica da proteína [Gobeil et al., 2021; Lima et al., 2021; Lupala et al., 2022; Verma & Subbarao, 2021; Leach, 2001, p. 392]. Nesse caso, algumas técnicas como a *transformada de Fourier* podem ser usadas para filtrar os movimentos de alta frequência, de modo que permaneçam apenas as mudanças conformacionais sutis (baixa frequência) [Dauber-Osguthorpe & Osguthorpe, 1993].

Nesse sentido, simulações de DM costumam ser desenvolvidas em dois momentos: um em baixa, e outro de alta resolução. No primeiro, o sistema é inserido em um ambiente aquoso onde o objetivo é encontrar estruturas que desenvolvam interações apolares uma vez que, durante o enovelamento da proteína, o efeito hidrofóbico é responsável pela forma globular da proteína. Em um segundo momento, a análise concentra-se em estruturas isoladas na primeira etapa, com ênfase na busca pela estrutura nativa da proteína. Assim, por envolverem todos os átomos pertencentes ao sistema (especialmente as interações eletrostáticas e de van der Walls), as análises tendem a requisitar mais em relação a recursos computacionais [Kessel & Ben-Tal, 2018, p. 232].

Conforme descrito anteriormente, as simulações de dinâmica molecular focam em reproduzir o comportamento real de moléculas em movimento, o que envolve a análise das múltiplas interações estabelecidas entre os diferentes resíduos dos sistemas [Durrant & McCammon, 2011]. O uso de campos de força reduz a complexidade dos cálculos de interações resíduo-a-resíduo [Kessel & Ben-Tal, 2018, p. 232–236], contudo ainda exigem muito em termos de processamento.

Devido ao volume de informação gerada, as trajetórias provenientes de dinâmica molecular podem ser consideradas como dados em *larga escala*, estando sua obtenção

[5]Isso ocorre em virtude da periodicidade de 2π do ângulo de torção [Leach, 2001, p. 392].

comumente associada a um alto custo computacional [Durrant & McCammon, 2011]. Em um cenário onde é necessário prever o impacto de mutações na afinidade de ligação em complexos proteína-ligante, por exemplo, cada mutação requisitaria uma nova simulação do sistema. Isso tornaria o processo custoso computacionalmente, devido a combinação de mutações possíveis.

Ainda nessa linha, é importante observar que algumas proteínas, ou estruturas, são *homólogas*, compartilhando padrões similares quanto a flexibilidade estrutural. Assim, é possível inferir que para um subconjunto de mutações, os efeitos na conformação da estrutura, ou ainda na função da proteína, tendem a ser similares [Iyer et al., 2022].

Para prever o impacto de determinada mutação nas interações proteína-ligante, é comum a realização de análises, como o cálculo do RMSF [Alaofi & Shahid, 2021; Luan et al., 2021; Ou et al., 2021; Verma & Subbarao, 2021; Lupala et al., 2022]. O RMSF, ou Raiz Quadrada da Flutuação Quadrática Média (do inglês, *Root Mean Square Fluctuation*), é uma métrica que representa o quanto os segmentos de uma estrutura oscilam ao longo do tempo, sendo comumente referida como uma medida de flexibilidade da cadeia. O cálculo baseia-se nas diferenças *individuais* das coordenadas de resíduos que compõem um par de moléculas. Uma das estruturas é definida como referência e a outra varia no tempo. A referência corresponde a mesma estrutura a ser analisada, mas simulada dinamicamente, considerando que ambas estão alinhadas tridimensionalmente.

No contexto de dinâmica molecular, é possível avaliar impacto de mutações na conformação de regiões em proteínas, estimando o RMSF entre as estruturas, sendo uma selvagem (referência) e outra mutada (móvel no tempo). Com o RMSF é possível identificar padrões dinâmicos da estrutura em uma simulação temporal, como resíduos em regiões de loop. O RMSF de uma estrutura também pode ser interpretada como a média temporal do RMSD [6]. Sejam duas estruturas, onde ref indica a referência, x_i, y_i, z_i as coordenadas de um resíduo i no espaço e, t número total de intervalos de tempo considerados, o RMSF pode ser obtido a partir da equação abaixo:

$$RMSF_{residuo} = \sqrt{\frac{1}{t}\sum_{i=1}^{t}(x_i^{ref} - x_i)^2 + (y_i^{ref} - y_i)^2 + (z_i^{ref} - z_i)^2)} \tag{2.4}$$

[6] O RMSD, ou Raiz Quadrada do Desvio Quadrático Médio (do inglês, *Root Mean Square Deviation*) é uma métrica de similaridade entre estruturas de proteínas alinhadas tridimensionalmente. Nesse caso, é necessário que seja estabelecida uma equivalência entre os pares de átomos das moléculas, a partir de uma sobreposição estrutural [Kessel & Ben-Tal, 2018, p. 149].

2.2.3 Energia livre de ligação e $\Delta\Delta G$

A diversidade de ligantes naturais existentes na natureza estabelece características singulares às forças que regem as interações em complexos proteína-ligante. A afinidade, estabilidade, ou flexibilidade na região de interação, são exemplos da relevância das forças estabelecidas entre os resíduos do complexo. Essas forças interatômicas estão associadas ao propósito biológico da proteína, sendo que um fatores que contribuem a isso sao *mudanças na energia livre de ligação* [Kessel & Ben-Tal, 2018, p. 519].

A *energia livre de ligação* (ΔG) corresponde a um balanceamento sensível entre forças que se opõem a formação do complexo proteína-ligante e, forças que favorecem a ligação. Normalmente, as interações proteína-ligante são não-covalentes, sendo a ΔG um valor de pequena magnitude. Isso deve-se a diferença entre a energia livre do complexo proteína-ligante G_{PL} e, as energias livres da proteína (G_P) e do ligante (G_L), respectivamente [Kessel & Ben-Tal, 2018, p. 521]. Essa relação pode ser descrita como

$$\Delta G = G_{PL} - (G_P + G_L). \tag{2.5}$$

A energia livre de ligação possui uma *alta dependência* da conformação da estrutura, o que resulta das múltiplas interações de curto e longo alcance no sistema, além do efeito do dielétrico nas interações eletrostáticas. Contudo, para observar esse comportamento, cálculos precisos requerem tratamento do sistema a partir de simulações de dinâmica molecular. A tendência é que ligantes estrutural/quimicamente diferentes induzem diferentes conformações na proteína [Kessel & Ben-Tal, 2018, p. 522].

Quando a estrutura da proteína e ligante são disponíveis, uma forma precisa de calcular a diferença de energia livre associada é usar um modelo explícito do sistema completo, aplicando as equações de campo de força. Considere uma proteína P e dois ligantes possíveis, L e L'. A partir da equação 2.5, a energia livre de ligação (ΔG) é obtida subtraindo-se a energia livre do complexo (ΔG_{PL}), das energias livres da proteína (ΔG_P) e do ligante (ΔG_L), separadamente [Kessel & Ben-Tal, 2018, p. 523].

$$\Delta G = \Delta G_{PL} - \Delta G_P - \Delta G_L \tag{2.6}$$

$$\Delta G' = \Delta G_{PL'} - \Delta G_P - \Delta G_{L'} \tag{2.7}$$

Assim, a *variação* da energia livre de ligação ($\Delta\Delta G$), pode ser definida como:

$$\Delta\Delta G = \Delta G' - \Delta G. \tag{2.8}$$

2.3 SARS-CoV-2

2.3.1 Contexto pandêmico

O primeiro caso identificado da Síndrome Respiratória Aguda Grave associada ao *novo coronavírus* (do inglês *Severe Acute Respiratory Syndrome Coronavirus-2*, SARS-CoV-2 ou ainda *2019 novel coronavirus*, 2019-nCoV) ocorreu na cidade de Wuhan, pertencente a província de Hubei, na China, em meados de dezembro de 2019. O SARS-Cov-2 possui origem zoonótica, sendo o responsável pelo desenvolvimento da Doença por Coronavírus (*Coronavirus Disease*, COVID-19) [Wu et al., 2020]. Em janeiro de 2020, a Organização Mundial da Saúde (OMS) classificou o surto em Wuhan como Emergência Internacional de Saúde Pública e, em 11 de março de 2020, como pandemia. Em todo o mundo, à data de 17 de junho de 2022, 581.981.235 casos da doença foram registrados, com um total de 6.412.627 mortes causadas por complicações da COVID-19 [Johns Hopkins University (JHU), 2022]. No Brasil, foram confirmados 31.611.769 casos, com um total de 668.693 mortes relacionadas à doença [Ministério da Saúde, 2022].

2.3.2 Estrutura viral

Os vírus são complexos supramoleculares que possuem a capacidade de subverter a maquinaria de células hospedeiras específicas e replicarem-se. O genoma viral, constituído por uma molécula de ácido nucleico (DNA ou RNA), é encapsulado em um revestimento proteico, conhecido como capsídeo. Em alguns vírus, o capsídeo ainda pode ser revestido adicionalmente por uma membrana em bicamada lipídica, ou envelope. Uma vez que o vírus, ou seu genoma, adentra a celula hospedeira, ele torna-se um parasita intracelular [Nelson & Cox, 2004, p. 34]. Alguns vírus de RNA, como os da família *Coronaviridae*, possuem como característica proteínas transmembranares do envelope viral denominadas *peplômeros* [Alberts et al., 2002, p. 1274–1275].

Os peplômeros caracterizam-se por glicoproteínas encontradas na superfície do envelope viral, capazes de reconhecer e ligar-se a moléculas receptoras específicas na superfície das celulas hospedeiras formando complexos proteicos [Alberts et al., 2002, p. 1279–1281]. O processo de fusão com a membrana celular também pode ser mediado por peplômeros, pois a ligação com os receptores provoca mudanças conformacionais na estrutura do ligante, permitindo a entrada viral [Alberts et al., 2002, p. 1279–1281]. Conhecer o mecanismo de infecção dos vírus e suas implicações pode auxiliar no combate a diversos patógenos emergentes vide Síndrome Respiratória Aguda Grave associada ao novo coronavírus (*Severe Acute Respiratory Syndrome Coronavirus-2* - SARS-CoV-2), agente

patogênico responsável pelo desenvolvimento da Doença por Coronavírus (*Coronavirus Disease* - COVID-19) [Wu et al., 2020].

O SARS-CoV-2 é um betacoronavírus (β-CoV) da família *Coronaviridae* com genoma viral de RNA [Guruprasad, 2021], sendo a cepa originária conhecida como 2019-nCoV (*selvagem* ou, do inglês, *wild type*, WT). A infecção pelo SARS-CoV-2 ocorre através do reconhecimento do receptor humano pela glicoproteína espicular S (do inglês, *spike* S), cuja cadeia possui um total de 1273 aminoácidos. Essa proteína encontrada no envelope viral, caracteriza-se por um trímero transmembranar de fusão classe I onde cada monômero possui duas subunidades: S1 de ligação com o receptor da célula hospedeira e S2 de fusão e entrada viral [Huang et al., 2020; Zhang et al., 2021]. Sua função é permitir a infecção viral, possibilitada devido a afinidade de seu Domínio de Ligação ao Receptor (*Receptor-Binding Domain* - RBD) com a Enzima Conversora de Angiotensina II (*Human Angiotensin-Converting Enzyme 2* - hACE2), encontrada nas células epiteliais alveolares tipo II [Lan et al., 2020].

Figura 2.3. Estrutura cristalina do Domínio de Ligação ao Receptor da proteína S, em *ciano*, do SARS-CoV-2 ligado ao receptor hACE2, em *verde* (PDB ID 6M0J, Sehnal et al. [2021]). A região em *vermelho* corresponde ao Motivo de Ligação ao Receptor que interage diretamente com o hACE2. Algumas mutações em resíduos no RBM são críticas, pois impactam na afinidade das interações do complexo RBD-hACE2. Fonte: Elaborada pelos autores.

O RBD é um fragmento imunogênico compreendido na subunidade S1 da proteína S e atua como um componente funcional chave da infecção viral, a partir do reconhecimento do receptor humano [Zhang et al., 2021] (figura 2.3). Isso deve-se principalmente às interações entre um subconjunto de resíduos do RBD, região conhecida como Motivo de Ligação ao Receptor (*Receptor-Binding Motif* - RBM), e o hACE2 [Lan et al., 2020]. Por sua vez, a enzima hACE2 atua como mecanismo contrarregulatório da produção de Angiotensina II (relacionada à vasoconstrição), auxiliando na vasodilatação e diminuição da pressão arterial [Kessel & Ben-Tal, 2018, p. 704]. Uma vez que o SARS-CoV-2 utiliza a hACE2 para infectar a célula, a função dessa proteína fica comprometida (*downregulation*) ocasionando uma expressão exacerbada de Angiotensina II, um dos fatores relacionados ao desenvolvimento do dano pulmonar decorrente da COVID-19.

2.3.3 Variantes de interesse e de preocupação

As interações intermoleculares que formam o complexo RBD-hACE2 estão relacionadas à afinidade da ligação das proteínas virais ao receptor humano. Assim, a região do RBD que contém os resíduos de contato com o hACE2 torna-se propícia ao surgimento de mutações críticas, com o potencial de aumentar a afinidade ligante-receptor [Sanches et al., 2021; McCallum et al., 2022], ou ainda a evasão imune [Harvey et al., 2021]. Em seu trabalho, Verma & Subbarao [2021] analisam mutações no RBD e seus efeitos na estabilidade do complexo ligante-receptor, a partir de simulações de dinâmica molecular. Essas mutações tem sido alvo de pesquisas relacionadas ao desenvolvimento de vacinas capazes de induzir anticorpos capazes de neutralizar a ligação do RBD ao receptor humano [Lan et al., 2020].

Algumas mutações estão relacionadas com o aumento do número de interações entre resíduos no complexo RBD-hACE2 [Verma & Subbarao, 2021], caracterizando *variantes* que possuem maior afinidade ao receptor humano. Dentre essas variantes encontra-se a B.1.1.7, também denominada na literatura como Alfa (α). Os primeiros registros de casos relacionados a essa cepa datam de 14 de Dezembro de 2020, inicialmente, expandido-se no sudeste da Inglaterra. A variante alfa está associada a múltiplas mutações na proteína S, sendo mais notável a substituição da asparagina (Asp) por uma Tirosina (Tyr) no resíduo 501 do RBD (mais especificamente na região conhecida como Motivo de Ligação ao Receptor, do inglês *Receptor Binding Motif* - RBM) [Liu et al., 2021]. Essa mutação tem sido recentemente estudada com o auxílio de simulações de dinâmica molecular [Luan et al., 2021; Verma & Subbarao, 2021; Tian et al., 2021; Lupala et al., 2022] e, está relacionada a mudanças conformacionais na estrutura do RBD que propiciam a ligação com o receptor humano, devido a introdução de interações $\pi - \pi$ [Teruel et al., 2021; Harvey et al., 2021; Laffeber et al., 2021; Sanches et al., 2021; Lupala et al., 2022].

Além da Alfa, a mutação S:N501Y está presente nas seguintes variantes: B.1.351 (Beta, β) [Tegally et al., 2021], B.1.1.28.1 (Gama, γ, ou P.1)[Faria et al., 2021], BA.1, BA.2, BA.4-5 e BA.2.12.1 (Omicron, o)[Saxena et al., 2022]. Essas cepas foram classificadas pela Organização Mundial da Saúde (OMS) como *variantes de preocupação* (do inglês *variants of concern*, VOC) (figura 2.4), pois os primeiros trabalhos relacionados a elas sugerem um aumento na transmissibilidade, gravidade da doença, reinfecção e, redução da proteção conferida por anticorpos neutralizantes [Chen et al., 2021b; Tao et al., 2021; Muik et al., 2021; Wang et al., 2021; Wibmer et al., 2021; Akkız, 2022; Lupala et al., 2022]. Por fim, existem as *variantes de interesse* (do inglês *variants of interest*, VOI) cujo avanço ainda é monitorado pela OMS. Dentre essas, é possível encontrar a mutação S:N501Y na B.1.621 (Mu, μ)[Laiton-Donato et al., 2021] e B.1.1.28.3 (Teta, θ, ou P.3). A figura 2.5 apresenta as relações filogenéticas entre cepas de SARS-CoV-2, surgidas da cepa original. É possível visualizar que algumas variantes podem descender de outras, formando uma linhagem comum.

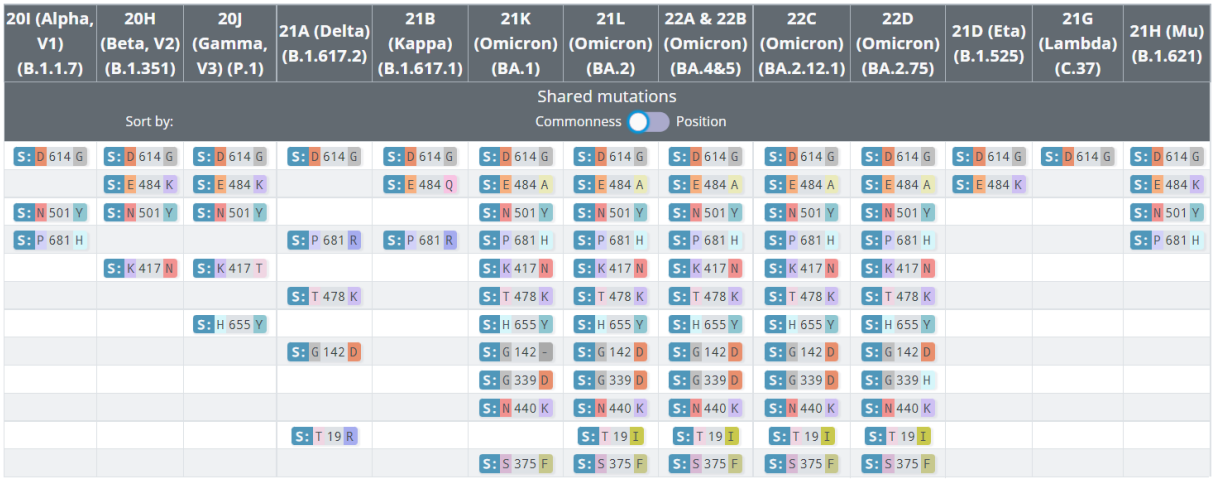


Figura 2.4. Algumas das cepas que surgiram ao final de 2020 e início de 2021 compartilham mutações definidoras de aminoácidos. A figura apresenta algumas das principais variantes de interesse aos cientistas. É possível verificar que mutação N501Y no RBD da proteína S (quarta linha da tabela) se faz presente nas variantes Alfa (B.1.1.7), Beta (B.1.351), Gama (P.1), Omicron (BA.1, BA.2, BA.4-5 e BA.2.12.1) e Mu (B.1.621). Fonte: Hodcroft [2021].

Uma forma de avaliar os efeitos das mutações e compreender a evolução da infectividade do SARS-CoV-2 é a partir da estimativa de variação da energia livre de ligação (do inglês *Binding Free Energy*, BFE) [Teng et al., 2020]. Estudos recentes sugerem que a energia livre de ligação entre o RBD da proteína S e o receptor humano ACE2 é proporcional à infectividade viral [Chen et al., 2021b; Hoffmann et al., 2020; Teng et al., 2020; Walls et al., 2020]. Uma variação mais significativa do $\Delta\Delta G$, induzida por uma mutação específica no RBD, indica um potencial de consolidar a ligação do complexo

proteína S-hACE2, enquanto uma alteração negativa, ou em menor magnitude, do $\Delta\Delta G$ sugere uma provável capacidade de reduzir, ou não afetar, a força de ligação da proteína S-ACE2 [Chen et al., 2021b].

A partir da equação 2.8 (subseção 2.2.3), a variação da energia livre de ligação ($\Delta\Delta G$), após a mutação, pode ser obtida pela subtração entre a energia livre de ligação do tipo selvagem (ΔG_S) e a energia livre de ligação com a mutação (ΔG_M),

$$\Delta\Delta G = \Delta G_S - \Delta G_M. \tag{2.9}$$

Uma mudança significativa no $\Delta\Delta G$ indica que a mutação aumenta a energia livre de ligação do complexo, tornando o vírus mais infeccioso [Chen et al., 2020b].

Teng et al. [2020] ainda sugerem que o $\Delta\Delta G$ pode ser utilizado para distinguir mutações prejudiciais de mutações neutras no RBD da proteína S. No caso das mutações neutras, observa-se que a substituição de resíduos do RBD tem efeito neutro na função da proteína S (ou seja, mutações que documentadamente não aumentam a afinidade, ou a resistência ao sistema imune).

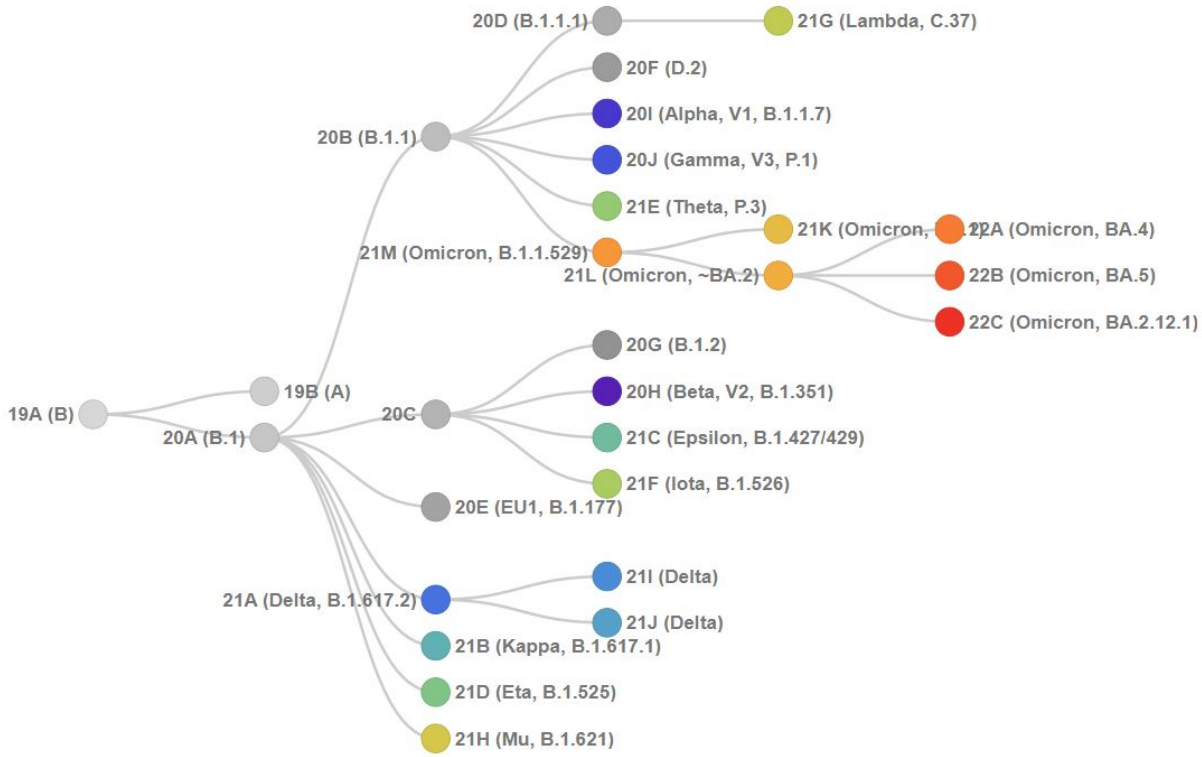


Figura 2.5. A figura apresenta as relações filogenéticas entre cepas de SARS-CoV-2, surgidas da cepa original, ou selvagem. É possível visualizar que algumas variantes podem descender de outras variantes, formando uma linhagem comum. Fonte: Hodcroft [2021].

2.4 Processamento Digital de Imagens

2.4.1 Abordagem tradicional no processamento digital de imagens

O campo do *processamento digital de imagens* refere-se ao processamento de imagens digitais por um computador, tendo origem na década de 1960 para uso em análise de imagens médicas, observações de recursos terrestres remotos e astronomia. O interesse em métodos dessa área decorre das necessidades em algumas aplicações, como: melhoria de informações pictóricas para interpretação humana; e processamento de dados em imagens para tarefas como armazenamento, transmissão e extração de informações pictóricas [Gonzalez & Woods, 2009, p. 17–21].

Ainda hoje, não há um algoritmo genérico capaz de realizar o processamento de imagens de ponta-a-ponta. Normalmente, são empregados diferentes algoritmos em sequência formando um *pipeline*, composto por estágios independentes. Assim, cada estágio é específico à execução de uma determinada tarefa, sendo que, na maioria das aplicações, o processamento digital de imagens engloba as seguintes etapas: filtragem e pré-processamento (preparação), condicionamento e simplificação (segmentação), extração de características, e classificação (reconhecimento de objetos, padrões, ou regiões). Nesse processo, a imagem é o dado de entrada e, ao final do processo, deseja-se obter uma *interpretação* sobre ela [Gonzalez & Woods, 2009, p. 41–43].

2.5 Redes Neurais Convolucionais Profundas

As Redes Neurais Convolucionais (do inglês *Convolutional Neural Networks*, CNNs/ConvNets) [LeCun et al., 1989] representaram uma quebra de paradigma no campo do processamento digital de imagens. Conforme descrito anteriormente, a abordagem tradicional do processamento digital de imagens pode ser entendida como um *pipeline* de etapas, composto por um conjunto de algoritmos independentes, responsáveis por tarefas específicas ao processamento da imagem. Com o advento das ConvNets, no final da década de 80, esse processo passou a ser compreendido de forma holística, onde cada etapa é codificada em um único algoritmo, constituindo um *bloco monolítico*. Assim, o algoritmo recebe como entrada uma imagem e fornece na saída uma interpretação/semântica sobre ela.

De modo a facilitar a compreensão do tema, nas próximas duas subseções, será realizada uma breve explanação sobre conceitos relevantes em redes neurais e aprendizado

profundo. Os aspectos arquiteturais bem como as atuais aplicações de ConvNets, serão abordados em maiores detalhes a partir da subsessão 2.5.3.

2.5.1 Redes neurais

Diversas definições a respeito de *redes neurais* descrevem a técnica como um modelo computacional, capaz de realizar um processo de aprendizagem de máquina. Contudo, essa linha de raciocínio ainda é ampla e, de certa forma, genérica. Isso pois, diversas técnicas, cujo *processo de aprendizagem* não seja equivalente às redes neurais, também consistem na obtenção de um modelo capaz de extrair conhecimento e tomar decisões autônomas (vide Agrupamento, Máquinas de Vetores de Suporte, Modelos de Misturas Gaussianas, etc). Assim, embora possam compartilhar o mesmo objetivo, muitas abordagens possuem formas *distintas* de aprender, o que tende a impactar significativamente no desempenho do modelo em determinada aplicação.

Assim, Haykin [2001] apresenta uma definição de rede neural que considera aspectos específicos à arquitetura, como a densa interconexão de células computacionais simples denominadas *neurônios* ou *unidades de processamento*, sendo vista como uma "máquina adaptativa":

> *Uma rede neural é um processador maciçamente paralelamente distribuído constituído de unidades de processamento simples, que têm a propensão natural para armazenar conhecimento experimental e torná-lo disponível para o uso. Ela se assemelha ao cérebro em dois aspectos:*
>
> 1. *O conhecimento é adquirido pela rede a partir de seu ambiente através de um processo de aprendizagem.*
> 2. *Forças de conexão entre neurônios, conhecidas como pesos sinápticos, são utilizadas para armazenar o conhecimento adquirido.*
>
> [Haykin, 2001, p. 28]

Os neurônios são unidades de processamento de informação fundamentais à operação da rede neural. A figura 2.6 apresenta o modelo não-linear de neurônio proposto por McCulloch & Pitts [1943], composto por três elementos básicos:

1. Sinapses: conexões entre os sinais de entrada $(x_1, x_2, ..., x_n)$ e o neurônio i. Cada sinapse possui um *peso sináptico* associado $(w_{k1}, w_{k2}, ..., w_{km})$. Um sinal x_j, na entrada da sinapse j conectada ao neurônio k, é multiplicado pelo peso (w_{kj}).

2. Somador: efetua uma combinação linear dos sinais de entrada com os respectivos pesos sinápticos.

3. Função de ativação (φ): limita o intervalo de amplitude do sinal de saída (y_k) a um valor finito (geralmente, o intervalo normalizado da amplitude é escrito como o intervalo unitário $[0, 1]$, ou $[-1, 1]$).

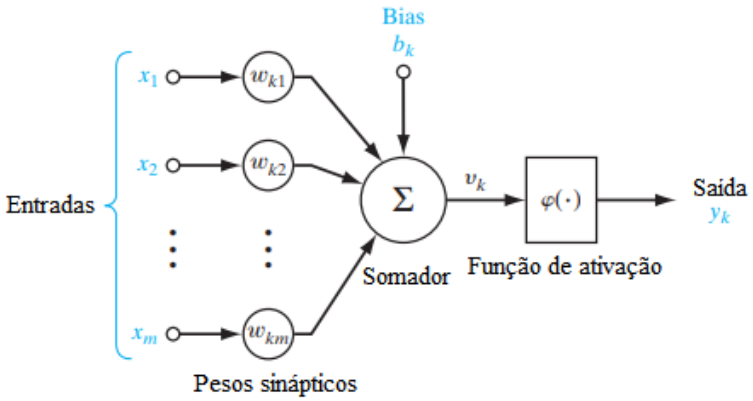


Figura 2.6. O modelo de neurônio de McCulloch & Pitts [1943] é composto pelos seguintes elementos: pesos sinápticos, somador, e a função de ativação. Fonte: Haykin [2001].

Nesse sentido, é possível descrever o neurônio k em função dos seguintes termos matemáticos:

$$u_k = \sum_{j=1}^{m} w_{kj} x_j \tag{2.10}$$

e

$$y_k = \varphi(u_k + b_k). \tag{2.11}$$

O termo b_k refere-se ao *bias* que tem o efeito de realizar uma transformação afim à saída u_k do combinador linear, como mostrado na figura 2.7. Com base no *bias* (positivo, ou negativo), a relação entre o *campo local induzido*, ou *potencial de ativação*, v_k e a saída u_k é modificada Fonte: [Haykin, 2001, p. 37], de acordo com a seguinte relação:

$$v_k = u_k + b_k. \tag{2.12}$$

Dentre algumas funções de ativação existentes, destacam-se duas em especial: ReLU e sigmóide. A *função ReLU* ou, em algumas aplicações de engenharia, também denominada *função rampa*, pode ser descrita matematicamente como:

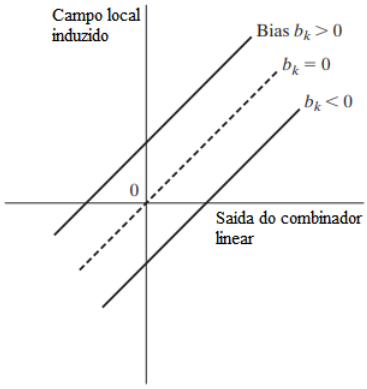
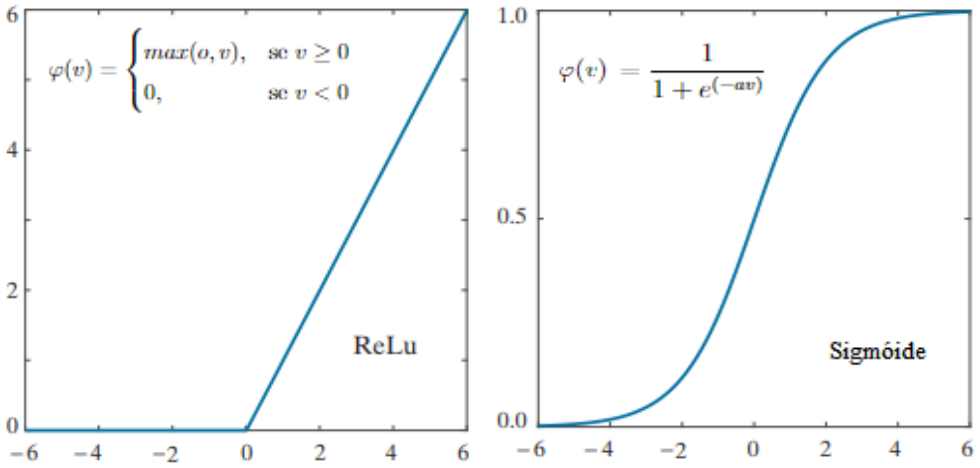


Figura 2.7. A aplicação do *bias* b_k resulta em uma transformação à saída u_k. Assim, dependendo do valor do *bias*, positivo ou negativo, a relação entre o *campo local induzido*, ou *potencial de ativação*, v_k e a saída u_k é modificada. Fonte: Haykin [2001]

$$\varphi(v) = \begin{cases} max(o, v), & \text{se } v \geq 0 \\ 0, & \text{se } v < 0 \end{cases} \tag{2.13}$$

Assim, a saída de um neurônio é igual ao campo local induzido, caso o valor do campo seja não negativo, e 0 caso contrário (figura 2.8a). Unidades neuronais que usam a função de ativação ReLU são denominadas como *Unidades Retificadoras Lineares* [Goodfellow et al., 2016, p. 187].



(a) função ReLU **(b)** função sigmóide

Figura 2.8. Curva característica das funções de ativação ReLU e sigmóide. O valor do coeficiente a (figura 2.8b) determina a inclinação da curva. Fonte: Gonzalez & Woods [2009]

Um outro exemplo de função de ativação empregada no projeto de redes neurais é a *função sigmóide*. Como característica, a sigmóide é uma função exclusivamente crescente e apresenta um balanceamento adequado entre o comportamento linear e não-linear [Hay-

kin, 2001, p. 40]. A equação 2.14 descreve a *curva logística* cujo formato é semelhante a sigmóide.

$$\varphi(v) = \frac{1}{1 + e^{(-av)}}. \tag{2.14}$$

O termo a refere-se ao *parâmetro de inclinação* da função sigmóide. Uma vez que o parâmetro a seja variado, obtém-se curvas características com diferentes inclinações (figura 2.8b) [Haykin, 2001, p. 40].

O modelo neuronal de McCulloch & Pitts [1943] produz uma saída não-linear em função dos sinais nas entradas. Contudo, em 1958, é proposta uma variação desse modelo conhecida na literatura como *perceptron*, ou ainda *perceptron de Rosenblatt* [Rosenblatt, 1958] (figura 2.9).

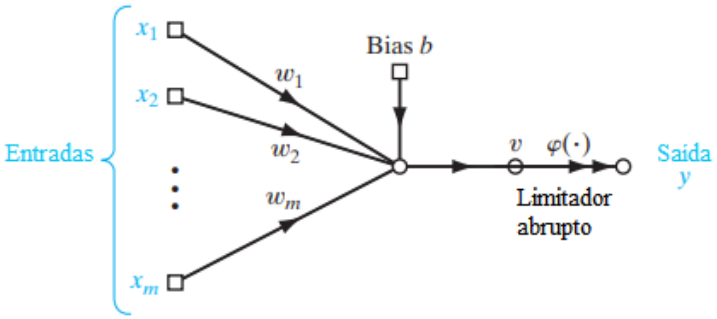


Figura 2.9. A figura apresenta o perceptron Rosenblatt [1958], inspirado no modelo não-linear de McCulloch & Pitts [1943]. A combinação linear entre as entradas $(x_1, x_2, ..., x_m)$ e os pesos sinápticos $(w_1, w_2, ..., w_m)$ resulta no campo local induzido v (considerando $b_k = 0$). Nesse caso, v é a entrada do limitador abrupto (φ) ou seja, a função de ativação que restringe a saída em dois valores. Fonte: Haykin [2001].

O perceptron é um modelo neuronal cujo objetivo é classificar um conjunto de entradas $x_1, x_2, ..., x_m$ em uma das dentre duas classes, C_1 ou C_2. Nesse caso, o campo local induzido (equação 2.15) é a entrada de um *limitador abrupto* ou seja, uma função de ativação que restringe a saída em apenas dois valores possíveis: $+1$ (referente a classe C_1) ou -1 (referente a classe C_2) [Haykin, 2001, p. 161].

$$v = \sum_{i=1}^{m} x_i w_i + b \tag{2.15}$$

Considerando um problema de classificação para duas classes, é possível delimitar a região de decisão no espaço através de um *hiperplano*, definido pela equação 2.16. Assim, cada entrada $x_1, x_2, ..., x_m$ de um sinal m-dimensional (m variáveis de entrada) será atribuída a região compreendida por cada uma das classes, localizando-se acima (classe C_2) ou abaixo (classe C_1) da fronteira de decisão (figura 2.10).

$$\sum_{i=1}^{m} x_i w_i + b = 0 \tag{2.16}$$

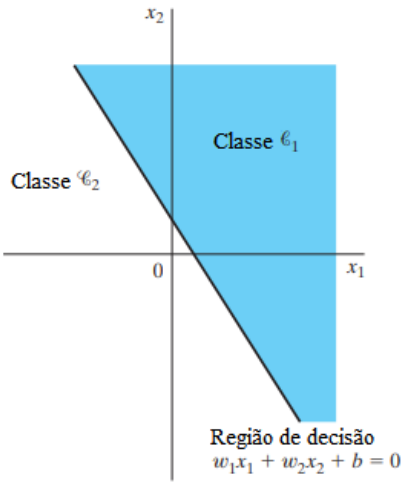


Figura 2.10. O modelo de neurônio de McCulloch & Pitts [1943] é composto pelos seguintes elementos: pesos sinápticos, somador, e a função de ativação. Fonte: Haykin [2001].

Como o objetivo do processo de aprendizagem seja a *minimização do erro* para o perceptron, infere-se que a saída do combinador linear depende dos vetores m-dimensionais correspondentes às entradas $x_1, x_2, ..., x_m$ e aos pesos sinápticos $w_1, w_2, ..., w_m$, para cada *iteração* n do algoritmo. Assim, considerando o bias $b(n)$ como um peso sináptico de entrada fixa igual a $+1$, as entradas e pesos sinápticos podem ser definidas conforme abaixo:

$$\mathbf{x}(n) = [+1, x_1(n), x_2(n), \ldots, x_m(n)]^T \tag{2.17}$$

$$\mathbf{w}(n) = [b(n), w_1(n), w_2(n), \ldots, w_m(n)]^T \tag{2.18}$$

A partir da equação 2.15, o campo local induzido pode ser definido como:

$$v(n) = \sum_{i=0}^{m} w_i(n) x_i(n) = \mathbf{w}^T(n)\mathbf{x}(n) \tag{2.19}$$

Sendo n fixo, a equação $\mathbf{w}^T\mathbf{x} = 0$ define o hiperplano de decisão entre as duas classes (C_1 e C_2), para diferentes entradas. Assim é possível estabelecer que, para duas classes *linearmente separáveis*, tem-se:

- $\mathbf{w}^T\mathbf{x} > 0$ para toda entrada $\mathbf{x}$ pertencente a classe C_1
- $\mathbf{w}^T\mathbf{x} \leq 0$ para toda entrada $\mathbf{x}$ pertencente a classe C_2

2.5.2 Redes neurais profundas

Uma vez que as classes não sejam linearmente separáveis, são necessárias mais unidades neuronais para resolver o problema. Nesse caso, a rede neural é implementada como sucessivas camadas, compostas por um conjunto de neurônios, conforme o modelo apresentado na figura 2.9. A exceção ocorre à *camada de entrada*, cujos nós são os componentes de um vetor padrão de entrada $\mathbf{x}$ [Gonzalez & Woods, 2009, p. 945]. Por essa razão, essas redes são denominadas como *perceptron de múltiplas camadas* (do inglês *multilayer perceptron*, MLP) pois representam uma generalização do perceptron de camada única, sendo consideradas aproximadores universais [Haykin, 2001, p. 183]. A figura 2.11 apresenta um modelo de rede neural baseado em MLPs.

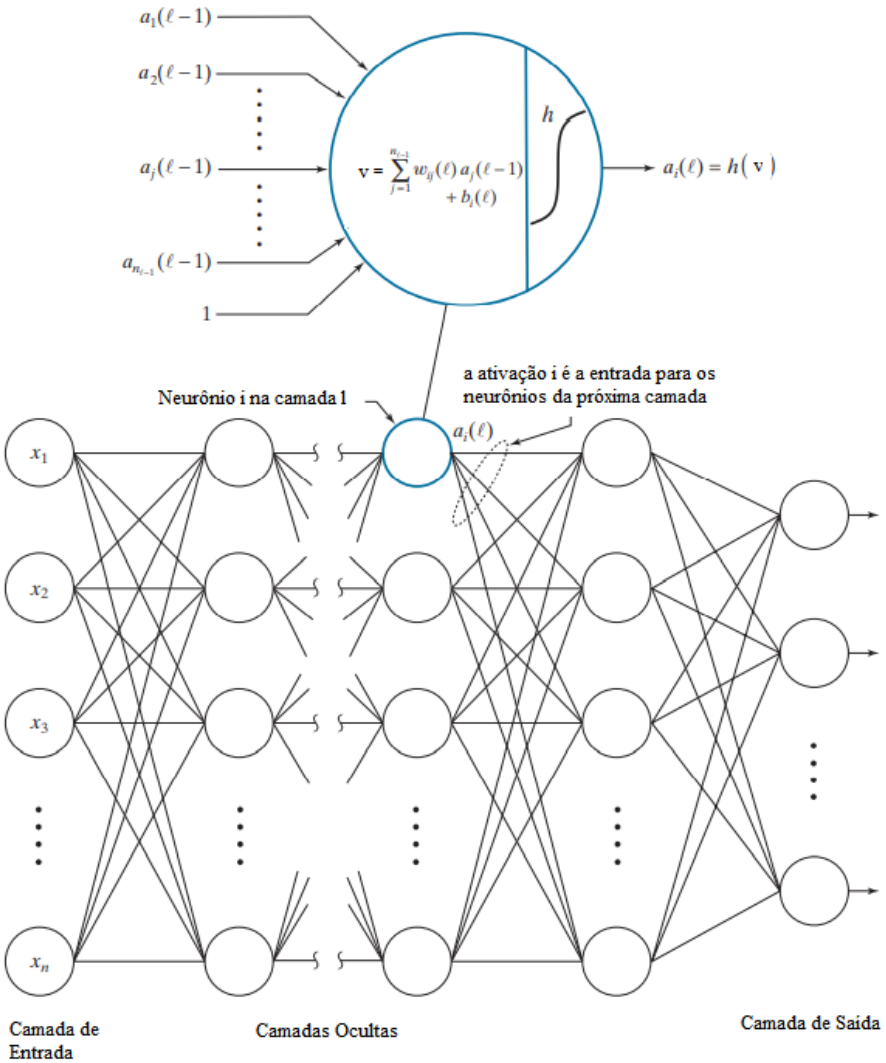


Figura 2.11. Representação de uma rede neural totalmente conectada. O neurônio é o mesmo do modelo apresentado na figura 2.9. É possível observar que a saída de cada neurônio é conectada a entrada de todos os neurônios da camada seguinte, característica dessa arquitetura. Fonte: Gonzalez & Woods [2009]

Uma característica dessa configuração é que a saída de um nó é conectada a entrada de cada nó da próxima camada, constituindo uma *rede totalmente conectada* (também conhecida como *fully connected*). Assim, os valores de ativação dos nós internos tornam-se entradas aos neurônios da próxima camada, constituindo as *camadas ocultas* [Gonzalez & Woods, 2009, p. 945]. Os neurônios ocultos possibilitam a rede extrair progressivamente as características mais significativas dos padrões de entrada [Haykin, 2001, p. 184], em virtude da *profundidade das camadas*. Por esse motivo, essas redes neurais também são denominadas *redes neurais profundas* (do inglês, *deep neural networks*, DNN). Essas redes neurais têm sido amplamente empregados, devido ao treinamento supervisionado aliado a um algoritmo de aprendizagem por *correção de erro*, conhecido como *algoritmo de retropropagação de erro* (do inglês *error back-propagation*) [Haykin, 2001, p. 183–184].

Basicamente o algoritmo de retropropagação consiste em duas etapas: a propagação (ou frente, do inglês *forward*) e a retropropagação (ou trás, do inglês *backforward*). Na propagação, o vetor $\mathbf{x}$ é aplicado a camada de entrada e, cada camada produz uma ativação no sentido entrada-saída da rede (*sinal funcional*). Ao final, obtém-se uma *resposta real* na camada de saída, sendo que nessa etapa os pesos sinápticos são fixos. Durante a retropropagação, os pesos sinápticos são *ajustados* em função de uma regra de correção de erro (normalmente a resposta real é subtraída de uma resposta desejada). O erro obtido é propagado através da rede em sentido oposto ao anterior ou seja, saída-entrada (*sinal de erro*), promovendo a atualização dos pesos sinápticos por camada. Esse processo repete-se ao longo das *épocas* de treinamento[7] da rede neural, de forma que o erro de saída decresce tendendo a zero ou seja, quando a resposta real aproxima-se da resposta desejada [Haykin, 2001, p. 183–184].

2.5.3 Redes convolucionais

Dentre os algoritmos de aprendizagem profunda existentes encontram-se as *Redes Convolucionais* [LeCun et al., 1989], uma classe de rede neural profunda, do tipo *feed-forward*, especializada no processamento de dados que possuem uma topologia de *grade*. Uma imagem, por exemplo, pode ser organizada como uma grade bidimensional de pixels. As redes convolucionais são inspiradas na organização do córtex visual humano, destacando-se como exemplo de como princípios neurocientíficos que influenciam o aprendizado profundo [Goodfellow et al., 2016, p. 321]. Em alusão ao seu nome, ConvNets empregam uma operação linear denominada *convolução* no aprendizado de características, ao invés

[7]Quantidade de vezes em que os dados de treinamento são apresentados para a rede neural de modo que, a atualização dos pesos da rede ocorra na direção da convergência do algoritmo de aprendizado [Theodoridis & Koutroumbas, 2009, p. 169].

da multiplicação matricial, comumente em redes neurais profundas (vide modelo apresentado na figura 2.11) [Goodfellow et al., 2016, p. 322].

A convolução é um operador linear que realiza o somatório do produto entre duas funções, ao longo da região na qual elas se sobrepõem, em razão do deslocamento existente entre elas. Sejam f e g duas funções contínuas no tempo t, e u o deslocamento, a convolução s pode ser descrita matematicamente como:

$$s(t) = \int_{-\infty}^{\infty} f(u)g(t-u)du. \tag{2.20}$$

A equação 2.20 pode ser adaptada de forma a calcular a convolução de sinais cujo domínio de tempo é discreto (adequado ao processamento em sistemas computacionais) [Goodfellow et al., 2016, p. 323]. Dessa forma, assumindo x e w como funções definidas no domínio de tempo discreto t, a convolução s pode ser escrita como:

$$s(t) = \sum_{a=-\infty}^{\infty} x(a)w(t-a). \tag{2.21}$$

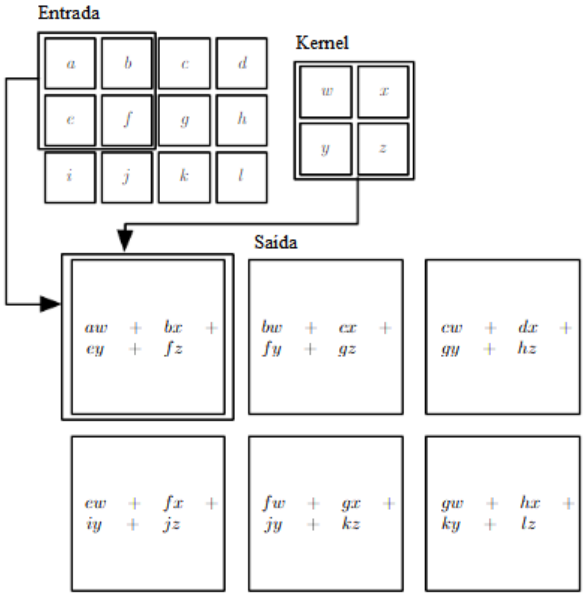


Figura 2.12. Exemplo de uma convolução 2D com um kernel 3×3. O tensor de saída é formado pela aplicação do kernel à região superior esquerda correspondente do tensor de entrada. Na figura, a saída corresponde apenas as posições onde o kernel está inteiramente dentro da imagem, conhecida como *convolução válida*. Fonte: Goodfellow et al. [2016]

Empregando essa notação ao contexto das redes convolucionais, o primeiro argumento (ou seja, a função x) corresponde a entrada, e o segundo argumento (função w) é denominado como *kernel* (figura 2.12). A saída convolvida é referida como *mapa de ativação*. Em aplicações de processamento digital de imagens, a entrada é comumente

representada por uma matriz multidimensional de dados (uma imagem, por exemplo), e o kernel como uma matriz multidimensional de parâmetros que são adaptados pelo algoritmo de aprendizado. Essas estruturas de dados são referidas como *tensores* [Goodfellow et al., 2016, p. 322–323]. Usando uma notação semelhante a equação 2.21, seja $x_{i,j}$ uma imagem bidimensional e w um kernel de dimensões l e k, o valor da convolução em qualquer ponto (i, j) na imagem pode ser descrito como

$$w \star x_{i,j} = \sum_{l} \sum_{k} w_{l,k} x_{i-l,j-k}. \tag{2.22}$$

2.5.4 Arquitetura básica de uma ConvNet

A arquitetura de uma rede convolucional consiste em camadas de extração de características (*camada convolucional* e *pooling*) e classificação (*flatten* e *fully connected*) [LeCun et al., 1989], podendo ser representada pelo diagrama da figura 2.13.

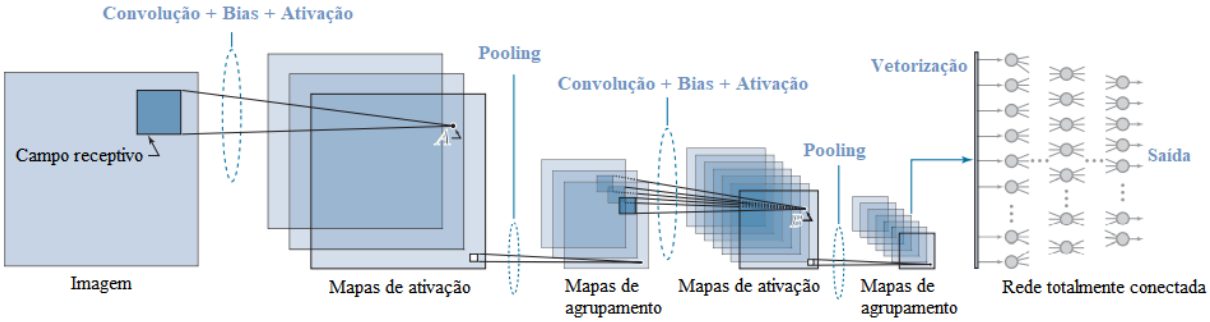


Figura 2.13. Representação contendo os elementos básicos da arquitetura de uma rede convolucional. Em seu trabalho, LeCun et al. [1989] aplicam o modelo com sucesso para o reconhecimento de dígitos de código postal manuscritos, fornecidos pelo Serviço Postal dos Estados Unidos. [LeCun et al., 1989]. Fonte: Gonzalez & Woods [2009]

Pela equação 2.22, infere-se que a soma dos produtos entre os valores dos pixels da imagem e o kernel (conjunto de pesos) resulta em uma saída convolvida (ativação linear). Nesse caso, o *campo receptivo* (mesma dimensionalidade do kernel) desloca-se pela imagem com base no *stride* (ou seja, um parâmetro do kernel que define a quantidade de pixels para movimento na imagem), selecionando a região com os pixels a serem convolvidos. Essa *convolução* é a entrada de uma função de ativação não-linear e, corresponde a saída de um neurônio oculto em uma MLP. Ao final, as saídas convolvidas resultam em um conjunto de valores de ativação (equação 2.24), que são armazenamos na próxima camada como uma matriz 2D, denominada de *mapa de ativação*.

$$z = w \star x_{i,j} + b. \tag{2.23}$$

$$a = \varphi(z). \tag{2.24}$$

Os mapas de ativação são entrada à *camada de pooling* (ou de *agrupamento*). Uma função de *pooling* calcula uma estatística, sobre uma matriz de valores pertencentes a determinada região do mapa de ativação. A camada de pooling produz um conjunto de mapas com resolução espacial reduzida, o que implica na redução do volume de dados processados pela rede neural. Dentre as principais funções de pooling estão: agrupamento médio (*average pooling*), a qual resulta na média dos elementos da matriz; valor máximo (*max-pooling*), que substitui a matriz pelo valor máximo de seus elementos; e L^2 *pooling*, onde o valor resultante é a raiz quadrada da soma dos elementos da matriz ao quadrado. Os agrupamentos médio e por valor máximo são amplamente utilizados na maioria das arquiteturas de ConvNets [Mishkin et al., 2017]. Para cada mapa de ativação, é gerado um mapa de características que são referidos coletivamente como *camada de pooling*.

O último estágio constitui em um classificador. Nesse caso, emprega-se uma rede totalmente conectada. Contudo, a saída das camadas convolucionais e de pooling são matrizes (imagens de dimensões reduzidas), impossibilitando a aplicação direta na MLP. Assim, é necessário vetorizar os mapas de características na última camada usando *indexação linear*. Cada mapa é convertido em um vetor, e os vetores resultantes são concatenados formando um único vetor unidimensional. Esse vetor constitui uma camada, que é entrada à rede totalmente conectada, conhecida como *flatten*.

Redes convolucionais têm alcançado desempenhos satisfatórios devido a sua capacidade de analisar informações espaciais [Min et al., 2017], o que possui relação com propriedades específicas de sua arquitetura. Uma dessas propriedades é a *conectividade esparsa*, ou seja, o uso de um *kernel* de tamanho menor que a entrada (figura 2.14). Assim, há uma concentração na identificação de padrões locais significativos (como padrões de pequena variação em interações intermoleculares [Min et al., 2017]), enquanto a convolução executa em tempo $O(k \times n)$ (onde k é o tamanho do kernel e n o número de saídas) reduzindo os requisitos de memória [Goodfellow et al., 2016, p. 324–326].

Outra propriedade é o *compartilhamento de parâmetros* que refere-se ao uso de um mesmo parâmetro por mais de uma função no modelo (figura 2.15). Em uma rede totalmente conectada, cada elemento da matriz de pesos é usado uma única vez à obtenção da saída. Em redes convolucionais, cada elemento do kernel é convolvido com as unidades da entrada (ou mapa de agrupamento)[Goodfellow et al., 2016, p. 326–328].

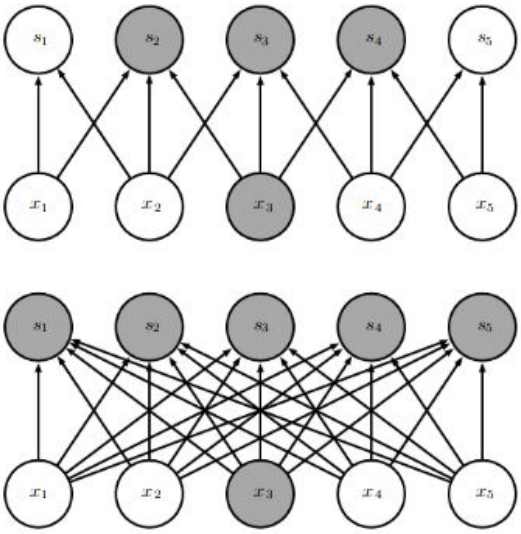


Figura 2.14. Em destaque, a unidade de entrada, x_3, e as unidades de saída s que dependem dessa unidade. (Acima) A convolução da unidade de entrada com um kernel de dimensões 3×3 resulta em três saídas, s_2, s_3, s_4. (Abaixo) Na multiplicação matricial, a conectividade não é esparsa logo, todas as saídas são afetadas por x_3. Fonte: Goodfellow et al. [2016]

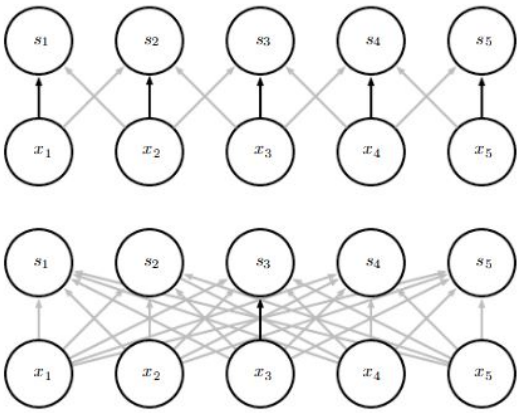


Figura 2.15. As conexões indicam que o parâmetro é usado em modelos diferentes. (Acima) As conexões realçadas indicam o peso central de um kernel de 3×3, e deslocamento de 1 passo, convolvido com as unidades de entrada. (Abaixo) A conexão única, corresponde ao elemento central da matriz de pesos em um rede totalmente conectada. Como não há compartilhamento de parâmetros, o peso é usado apenas uma vez. Fonte: Goodfellow et al. [2016]

Por fim, a *equivariância a translação* implica na correspondência entre a imagem e os mapas de ativação, independentemente da transformação (translação, deslocamento, etc) aplicada ao dado de entrada. Ou seja, mudanças nas imagens são reproduzidas da mesma forma nas camadas posteriores [Goodfellow et al., 2016, p. 329–330]. Uma função $f(x)$ é equivariante a g, se

$$f(g(x)) = g(f(x)). \tag{2.25}$$

2.5.5 Desenvolvimento de modelos baseados em redes convolucionais

A implementação dos modelos baseados em redes neurais convolucionais profundas, normalmente, baseia-se em duas aborgens: tranferência de aprendizado (do inglês, *transfer learning*), ou a utilização de uma arquitetura estabelecida na literatura. Na transferência de aprendizado, é utilizado um modelo previamente treinado em uma base de dados maior, e seu conhecimento é "transferido"para um conjunto de dados menor (por exemplo, os dados do problema)[Goodfellow et al., 2016, p. 526–527]. Essa abordagem é comum em aplicações relacionadas ao reconhecimento de objetos, ou regiões, em uma imagem [Gifani et al., 2020; Ali et al., 2021; Gour & Khanna, 2021].

Por outro lado, para aplicações mais específicas, é possível encontrar arquiteturas recentes que tornaram-se o *estado da arte* na classificação de imagens e reconhecimento de objetos. A LeNet-5 (figura 2.13), por exemplo, foi uma das primeiras estruturas de ConvNet propostas [LeCun et al., 1989], servindo como referência ao desenvolvimento de arquiteturas, cuja quantidade de camadas convolucionais era significativamente maior em relação à ela. A título de exemplo, arquiteturas como a AlexNet [Krizhevsky et al., 2017], GoogLeNet (também conhecida como InceptionNet) [Szegedy et al., 2015], VGGNet [Simonyan & Zisserman, 2014], ResNet [He et al., 2016], DenseNet [Huang et al., 2016a] e a FractalNet [Larsson et al., 2016], impulsionaram o uso de redes convolucionais em diversas áreas nos últimos anos.

Dentre essas áreas, observa-se que a *bioinformática* foi uma das que mais beneficiou-se, na última década, com a expansão dessas arquiteturas, conforme abordado por Min et al. [2017], em uma extensa revisão sobre os principais algoritmos de aprendizado profundo aplicados à pesquisa em bioinformática. No mesmo trabalho, os autores destacam o potencial das ConvNets no campo das ômicas e processamento de imagens biomédicas, o que pôde ser observado alguns anos mais tarde, com o trRosetta [Yang et al., 2020] e a primeira versão do AlphaFold [Alquraishi, 2019], que obteve resultados surpreendentes na 13ª Avaliação Crítica de Técnicas para Predição de Estrutura de Proteínas (do inglês, *13th Critical Assessment of protein Structure Prediction*, CASP13)[8]. Ainda assim, é possível encontrar aplicações de redes convolucionais onde o foco não é a predição de uma estrutura, mas a modelagem de novas estruturas desconhecidas [Gao et al., 2020; Defresne

[8]Em 2020, o AlphaFold 2 [Jumper et al., 2021], uma nova versão desenvolvida pela DeepMind do Google, alcançou uma precisão de 92,4 no Teste de Distância Global (do inglês, *Global Distance Test*, GDT), *score* utilizado pela Avaliação Crítica de Técnicas para Predição de Estrutura de Proteínas (do inglês, *Critical Assessment of protein Structure Prediction*, CASP). O resultado significa que, em mais da metade das predições feitas pelo programa de IA, a corretude dos átomos na estrutura predita encontra-se acima de 92,4%. Esse nível de precisão é comparável a técnicas experimentais, como a cristalografia de raios-X.

et al., 2021]. Essa técnica tem sido referida como *De Novo Protein Design* [Huang et al., 2016b], tendo recebido frequentes desenvolvimentos, como abordagens recentes baseadas em *estruturas de proteínas alucinadas* [Anishchenko et al., 2021].

Tabela 2.2. Relação das arquiteturas para as quais a entrada do modelo é uma estrutura representada por mapas de distâncias.

Classe	Aplicação	Autores
Redes Neurais Convolucionais	*De novo protein design*	Anishchenko et al. [2021], Baek et al. [2021];
	Predição de Estrutura	Alquraishi [2019], Chen et al. [2019b], Xu [2019], Senior et al. [2020], Yang et al. [2020], Jumper et al. [2021], Chen & Cheng [2022];
	Análise Conformacional	Zheng et al. [2020], Chen et al. [2022];
Redes Adversariais Generativas Convolucionais Profundas	*De novo protein design*	Anand & Huang [2018], Ding & Gong [2020];
Redes Neurais Generativas	*De novo protein design*	Li et al. [2017], Karimi et al. [2020];
Redes Neurais Recorrentes	Predição de Estrutura	Walsh et al. [2009];
Transformadores	*De novo protein design*	Jumper et al. [2021];

Uma vez que dados biológicos são tipicamente complexos e de alta dimensionalidade [Min et al., 2017], torna-se necessário o estudo das possíveis representações desses dados, de forma que seja possível a extração de características *significativas* pelos modelos de aprendizado profundo. Em relação as ConvNets, desde a década de 60, o foco das aplicações têm sido extrair características de anotações 2D mais informativas e complexas, como mapas de contatos [Wang et al., 2017; Adhikari et al., 2018; Jones & Kandathil, 2018] e mapas de distâncias (vide tabela 2.2) [Torrisi et al., 2020; Ding et al., 2022]. Nesse caso, é comum o dessas representações como entrada aos modelos.

Dentre essas arquiteturas, a VGGNet (cujo nome faz referência ao grupo de visão computacional desenvolvedor da arquitetura, *Visual Geometry Group*, da Universidade de Oxford, liderado por Andrew Zisserman and Andrea Vedaldi) tem destacado-se frequentemente por alcançar performances similares a ResNet, ou a GoogLeNet (tabela 2.3). A VGGNet caracteriza-se por um modelo *sequencial* ou seja, a rede possui exatamente uma entrada e uma saída, e uniforme, constituída por uma pilha linear de camadas convolucionais cujo número de filtros dobra a medida que a profundidade da rede aumenta [Chollet, 2017, p. 234]. O *kernel* com um campo receptivo de dimensões 3×3, desloca-se pelas imagens (ou mapas de agrupamento) com um *stride* fixado em 1 pixel. Na camada de *pooling*, uma função *max-pooling* é realizada com uma janela de 2×2 pixels, deslocando-se com um *stride* igual a 2. A configuração da camada totalmente conectada é similar as demais arquiteturas[9] [Simonyan & Zisserman, 2014].

Tabela 2.3. Taxa de erro às arquiteturas de ConvNet na ILSVRC2015

Modelo	Autores	Erro (%)
AlexNet	Krizhevsky et al. [2017]	16.42
GoogLeNet	Szegedy et al. [2015]	**6.66**
Inception-v3	Szegedy et al. [2015]	**3.58**
MSRA	He et al. [2014]	**8.06**
ResNet-50	He et al. [2016]	**3.57**
VGG-16	Simonyan & Zisserman [2014]	**6.8**

Taxas de erro (%) para as arquiteturas na *ImageNet Large Scale Visual Recognition Challenge* (ILSVRC2015). Em destaque, os modelos que encontram-se no *top*-5 das menores taxas de erro para o problema de classificação de imagens [He et al., 2016; Russakovsky et al., 2015]

[9]Com base nas configurações apresentadas por Simonyan & Zisserman [2014], a última camada é seguida pela função de ativação softmax. A softmax transforma a saída da rede neural em uma distribuição de probabilidade, que encontra-se em um intervalo de $[0 - 1]$ e, cuja soma seja igual a 1 [Theodoridis & Koutroumbas, 2009, p. 174].

A escolha pela VGGNet motivou-se por sua estrutura sequencial, cuja implementação é similar a ConvNet. Assim, uma vez que não se utiliza *transfer learning* (i.e., uso de modelos de VGG disponibilizados pelo Keras como o VGG16), a implementação do código à VGG ocorreu de forma análoga a uma implementação para a ConvNet (i.e., um *baseline default*, exceto pela quantidade de camadas convolucionais, que é relativamente maior). No trabalho de Simonyan & Zisserman [2014], observa-se algumas configurações da VGG usadas na classificação dos dados da ImageNet (tabela 2.16). Com base nisso, optou-se pela configuração B, também conhecida como VGG13.

Configuração da Rede Convolucional					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Figura 2.16. A profundidade das configurações aumenta da esquerda (A) para a direita (E), à medida que mais camadas convolucionais são adicionadas. Os parâmetros da camada convolucional são denotados como "conv<tamanho do campo receptivo>-<numero de filtros>". Fonte: Simonyan & Zisserman [2014]

Capítulo 3

Abordagem Metodológica

3.1 Representação das Trajetórias de Dinâmica Molecular

3.1.1 Base de dados

As trajetórias das simulações de DM baseiam-se na região do RBD da proteína S (PDB ID 6M0J [Lan et al., 2020]), pois mutações na região podem ocasionar ganho de afinidade com o receptor hACE2, além de mudanças na mobilidade da estrutura [Harvey et al., 2021; Sanches et al., 2021]. Os sistemas possuem triplicatas de 25.000 frames, com simulações de 100 ns, totalizando 75.000 frames (arquivos *.pdb*). Destes, 5.000 frames compõem o *espaço amostral*, amostrados a partir de um *skip*[1] de 15 frames.

Uma vez que foram produzidos um total de 10 sistemas (*selvagem* e mais 9 (nove) mutações), cada análise relacionada às dinâmicas envolveria a trajetória completa. Algumas análises em DM podem tornar-se inviáveis em termos de recursos computacionais. Além disso, o treinamento dos modelos de I.A. seria extenso, pois a rede neural necessitaria processar 120.000 frames (considerando um problema de classificação binária). Assim, o modelo poderia apresentar problemas no teste, como o sobreajuste (do inglês, *overfitting*) [Duda et al., 2001, p.134–135], ou ainda o problema de *Vanishing Gradient*, comum em redes profundas [Gonzalez & Woods, 2009, p. 988–989].

Muito embora o espaço amostral corresponda 6,66% do tamanho do sistema, o conjunto de frames amostrados deriva da trajetória completa ao longo do tempo. O alinhamento das trajetórias foi realizado com relação ao primeiro frame, usando o programa CCPTRAJ.

[1]Intervalo para amostragem dos frames ao longo do tempo na trajetória de dinâmica molecular. Normalmente, é usado para reduzir o tamanho da trajetória completa.

3.1.2 Representação dos frames por mapas de distâncias

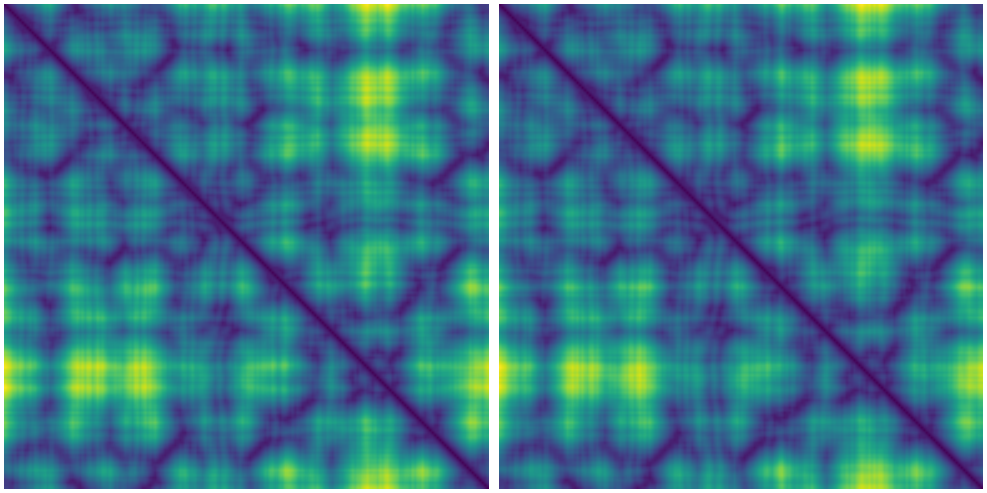
A partir dos *frames* são gerados mapas de distâncias que serão entradas para o modelo de IA [Roe & Cheatham, 2013]. Assim, o conjunto de mapas categorizados com a classe *selvagem*, correspondem aos *frames* de uma simulação do RBD da proteína S, sem qualquer mutação. Por outro lado, o conjunto de mapas categorizados com a classe *mutada*, correspondem aos *frames* de uma simulação do RBD da proteína S com uma mutação (em um, ou mais resíduos). Nesse caso escolheu-se a mutação N501Y (B.1.1.7, ou Alfa), por ser compartilhada por múltiplas variantes do SARS-CoV-2 (vide subseção 2.3.3).

Abordagens matemáticas e computacionais aplicadas ao estudo de propriedades conformacionais de proteínas são bastante úteis em bioinformática estrutural. Uma representação comumente utilizada para descrever a estrutura e dinâmica de uma proteína é a matriz de distâncias [Kloczkowski et al., 2009; Leach, 2001, p. 467–474]. Uma matriz de distâncias, $\mathbf{d} = (d_{ij})$, é obtida a partir do cálculo da distância entre o i-ésimo e o j-ésimo resíduo. Usualmente, a distância é medida entre os átomos de C_α (carbono-α) dos resíduos [Kloczkowski et al., 2009; Wang et al., 2017].

Soluções recentes à classificação e predição da estrutura de proteínas têm utilizado representações 2D (imagens) dessas matrizes, conhecidas por *mapas de distâncias* [Anishchenko et al., 2021; Jumper et al., 2021]. Um mapa de distâncias (MD) é uma matriz das *distâncias inter-resíduos*, de todos os pares de resíduos em uma proteína (normalmente utilizando-se os resíduos C_α [Wang et al., 2018; Defresne et al., 2021], ou C_β [Gao et al., 2020]) e oferece uma representação alternativa de estruturas de proteínas. Uma estrutura que sofre uma mudança conformacional para dois estados, por exemplo, pode ser descrita por dois mapas de distâncias (um para cada conformação) [Iyer et al., 2022]. A figura 3.1, apresenta o mapa de distâncias de cada classe.

Diversos estudos mostraram que proteínas, ou estruturas, homólogas compartilham padrões similares de flexibilidade estrutural, relacionados a mudanças conformacionais que podem ser comparadas com base na representação por mapas de distâncias [Chen et al., 2014; Iyer et al., 2022]. Algumas conformações em larga escala, são observadas usando limiares acima de 3Å RMSD. Contudo, quando o limiar é definido para valores menores, um número maior de conformações locais (em menor escala), podem ser identificadas para uma determinada proteína/estrutura [Iyer et al., 2022].

Recentemente, a biologia estrutural tem-se concentrado em detalhes relacionados à função das proteínas, como as diferenças entre as conformações de estruturas semelhantes [Iyer et al., 2020]. Assim, os mapas de distâncias têm ressurgido em aplicações de predição da estrutura de proteínas [Gao et al., 2020; Torrisi et al., 2020], com o propósito de fornecer uma melhor comparação das estruturas e, uma análise mais detalhada das diferenças entre



(a) Selvagem **(b)** B.1.1.7 (Alfa)

Figura 3.1. Mapas de distâncias para a classe *selvagem* (figura 3.1a) e *mutada* (figura 3.1b), referentes ao frame 4601 da trajetória. É possível observar a similaridade entre os mapas (a olho nu), ilustrando a dificuldade do problema. Fonte: Elaborado pelos autores.

os estados funcionais das proteínas Iyer et al. [2020]. Esses mapas possuem como vantagem uma baixa dimensionalidade, sendo invariantes a rotação (ou translação) das proteínas, tornando o cálculo de parâmetros e aprendizado eficiente [Defresne et al., 2021], algo desejável em aplicações de inteligência artificial.

3.2 Desenvolvimento do Modelo Baseado em Redes Convolucionais

3.2.1 Problema de identificação da variante

A identificação de mudanças conformacionais relacionadas a mutações no RBD da proteína S do SARS-CoV-2, que são responsáveis pelo aumento de afinidade com o receptor humano, pode ser modelada como um problema de *classificação binária*. O objetivo é determinar se padrões dinâmicos no RBD da proteína S, relacionados às distâncias interatômicas, caracterizam a cepa originária de Wuham (também denominada como *2019-nCoV*, ou *selvagem*)[Wu et al., 2020], ou a mutação. Nesse sentido, o problema pode ser formulado como um teste de duas hipóteses mutuamente exclusivas:

- H_0, conhecida como hipótese nula, corresponde a selvagem
- H_1, hipótese alternativa, não-selvagem (i.e., qualquer uma das variantes)

Como são somente duas hipóteses, o teste da *razão da verossimilhança* (do inglês *likelihood ratio*, LR) pode ser utilizado para tomar uma decisão [Fukunaga, 1990, p.51–52]. O classificador ótimo pode ser obtido pela razão da verossimilhança (Λ) entre duas hipóteses e um limiar, representado por

$$\Lambda(y) = \frac{p(y \mid H_1)}{p(y \mid H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \lambda, \tag{3.1}$$

onde $p(y|H_i)$ é a função de verossimilhança para a hipótese H_i, onde $i = \{0, 1\}$, avaliada para a amostra y e, λ é o limiar de decisão (também conhecido como *ponto de operação*). Também conhecida como *regra de decisão de Bayes*, a relação pode ser interpretada da seguinte forma: para uma dada amostra y a hipótese H_0 será rejeitada, caso a razão de verossimilhança seja maior que um limiar λ, que é independente de y [Duda et al., 2001, p. 25].

Existem dois tipos de erros que podem ocorrer na classificação. Erros do tipo I ocorrem quando a hipótese nula H_0 é rejeitada, quando é verdadeira. Os erros deste tipo são chamados de perdas (do inglês *misses*), isto é, o modelo decide que a amostra corresponde a mutação, quando o correto seria a selvagem. Erros do tipo II ocorrem quando a hipótese nula não é rejeitada quando for falsa. Os erros deste tipo são chamados de falso alarme (do inglês, *false alarm*), ou seja, o modelo decide que a amostra corresponde a selvagem, quando a mesma é uma mutação.

3.2.2 Abordagem comum em aprendizado profundo

A implementação do modelo de aprendizado profundo tem como referência a abordagem conhecida como *Representation Learning* [Bengio et al., 2013], sendo apresentada na figura 3.2). Os blocos sombreados indicam componentes na arquitetura da rede neural, que são responsáveis pela extração de conhecimento a partir dos dados dos de entrada [Goodfellow et al., 2016, p. 10]. Basicamente o objetivo no treinamento de modelos baseados em arquiteturas de aprendizado profundo é a otimização dos parâmetros de peso a cada camada, a partir da combinação de características simples e características mais complexas, obtendo representações hierárquicas mais adequadas a partir dos dados de entrada [Min et al., 2017].

É possível substituir os blocos em destaque pelo diagrama da figura 3.3. O neurônio k possui como entrada o *vetor de sinal* $\mathbf{x}(n)$ advindo das camadas ocultas da rede profunda. O argumento n representa o passo no processo iterativo de atualização dos pesos sinápticos do neurônio k. A saída do neurônio k é um sinal representado por $y_k(n)$ o qual é comparado com uma *resposta desejada*, representada por $d_k(n)$. O resultado dessa comparação produz

um *sinal de erro* $e_k(n)$. Esse sinal serve como referência aos ajustes corretivos dos pesos sinápticos do neurônio de saída k[2]. O objetivo é aproximar a cada iteração o sinal de saída $y_k(n)$ da resposta desejada $d_k(n)$, de forma a *minimizar o erro* até que o sistema alcance o *estado estável* (ou seja, os pesos sinápticos estão estabilizados). Esse processo de aprendizagem é conhecido como *aprendizagem por correção de erro* [Haykin, 2001, p. 76–77].

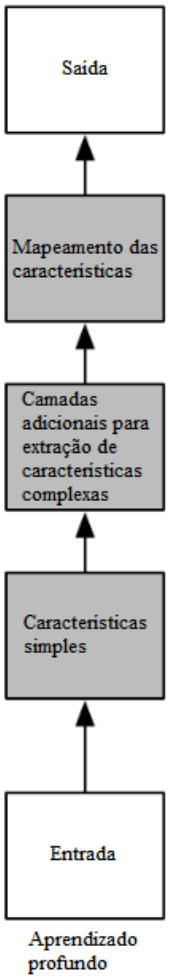


Figura 3.2. Fluxo dos elementos que compõem a maioria das arquiteturas baseadas em aprendizado profundo. Os blocos sombreados indicam componentes na arquitetura da rede neural, que são responsáveis pela extração de conhecimento a partir dos dados de entrada. Fonte: Goodfellow et al. [2016]

Uma vez que o erro $e_k(n)$ resulta da comparação entre o sinal de saída $y_k(n)$ e a resposta desejada $d_k(n)$, infere-se que, no treinamento, o modelo conhece o rótulo real correspondente ao dado de entrada (ou de treinamento). Esse tipo de aprendizado é conhecido como *supervisionado*, pois consiste em apresentar um padrão de entrada, e a

[2]A atualização dos pesos sinápticos normalmente é realizada com base no algoritmo de retropropagação de erro, descrito ao final da seção 2.5.2.

resposta desejada (comumente referida como *resposta de destino*) ao sinal da camada de saída. Assim, o algoritmo de aprendizagem deverá explorar esta informação *a priori* e ajustar os parâmetros da rede (pesos sinápticos) de forma que o sinal de saída aproxime-se da resposta de destino [Duda et al., 2001, p. 289].

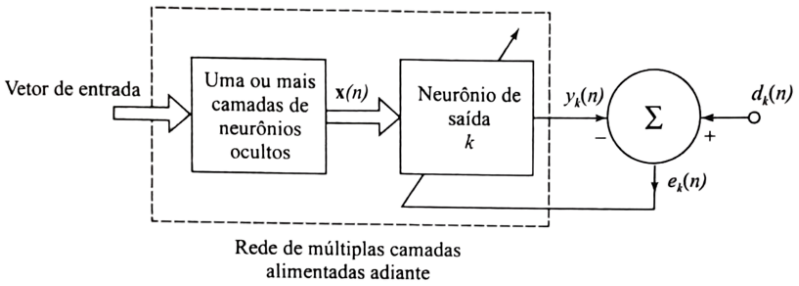


Figura 3.3. Representação simplificada da aprendizagem por correção do erro. A saída do neurônio k é um sinal representado por $y_k(n)$, que é comparado a *resposta desejada*, representada por $d_k(n)$. O resultado da comparação produz um *sinal de erro* $e_k(n)$, que serve de referência ao ajuste dos pesos sinápticos do neurônio de saída k. Fonte: Haykin [2001]

O modelo proposto é uma rede convolucional baseada na arquitetura VGGNet, que aprende a partir de um algoritmo de *apredizagem supervisionada por correção de erro*. Uma vez que mapas de distâncias são uma representação visual 2D de matrizes de distâncias, torna-se possível o uso de redes convolucionais ao problema (vide tabela 2.2). Assim, os mapas de distâncias foram estruturados como *tensores*, definidos em função das dimensões da imagem, número de canais (RGB), e do Tamanho de Lote (do inglês *Batch Size*, BS). O BS é um hiperparâmetro que define o número de amostras utilizadas à atualização dos pesos da rede neural [Chollet, 2017, p. 36], sendo, nesse caso, utilizado um BS igual a 64[3]. As entradas (mapas de distâncias) são pré-processadas de forma que os valores dos pixels são *normalizados* em um intervalo de $[0-1]$, próprio ao processamento por redes neurais [Goodfellow et al., 2016, p. 448]. Com relação a VGG, empregou-se a configuração B (tabela 2.16), também conhecida como VGG-13. Os algoritmos para obtenção dos mapas de distâncias e implementação do modelo foram desenvolvidos na linguagem de programação Python (versão 3.7.9), além de bibliotecas de aprendizado de máquina e redes neurais consolidadas como o TensorFlow [Abadi et al., 2016] e o Keras [Chollet et al., 2015]. O diagrama referente ao processo de aquisição dos dados e treinamento do modelo, é apresentado na figura 3.4.

A base de dados foi particionada em subconjuntos de *treinamento* e *teste*, em proporções de 80% e 20%, respectivamente. Os dados de teste foram previamente extraídos, prevenindo enviesamento ao processo de treinamento. Os dados restantes foram utilizados

[3]Tamanhos de lote na faixa de 64 a 256 amostras, têm produzido melhores resultados em diversas aplicações [Mishkin et al., 2017; Jumper et al., 2021].

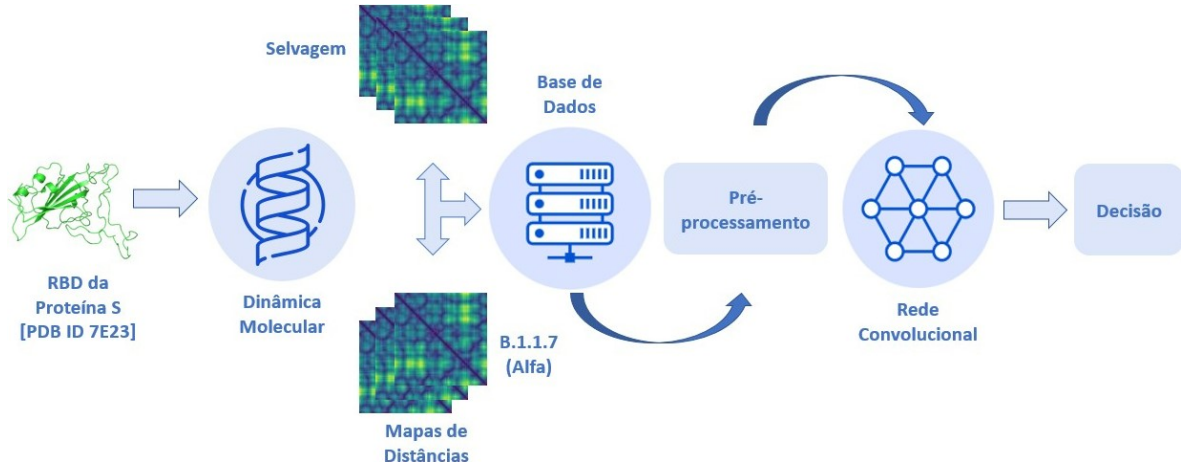


Figura 3.4. Diagrama referente ao processo de aquisição dos dados e treinamento do modelo. Os mapas de distâncias constituem a base de dados do problema, e servem como entrada à rede convolucional. Nesse estágio, o objetivo é a obtenção de um modelo que possua elevada capacidade discriminativa às classes (Selvagem e Alfa). Fonte: Elaborado pelos autores.

para treinamento do modelo, sendo que 80% são usados para ajuste dos hiper-parâmetros da rede, enquanto 20% são usados para a *validação* do modelo (5-*fold cross validation*, descrito na subseção 3.2.3) [Kearns, 1997; Duda et al., 2001, p. 483]. A figura 3.5 representa, de forma simplificada, o processo descrito[4].

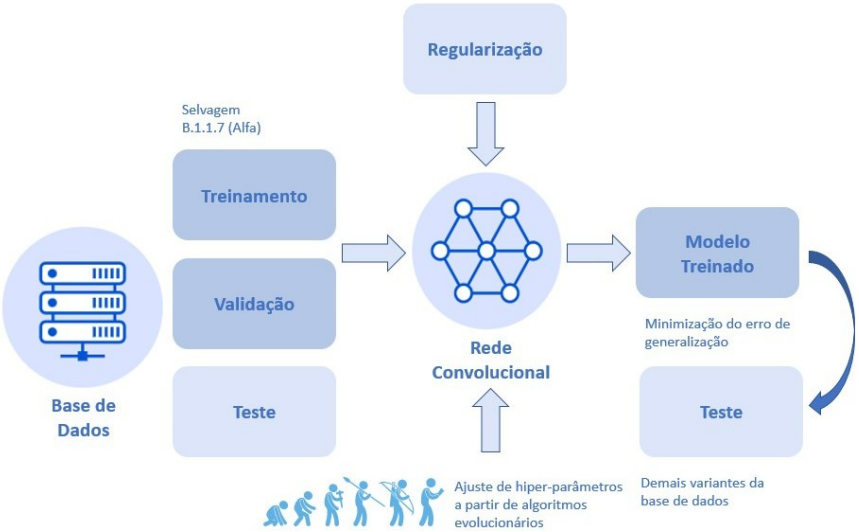


Figura 3.5. Diagrama referente ao processo de validação do modelo. A base de dados é particionada em conjuntos de treinamento e teste. Essa abordagem é necessária ao ajuste de hiperparâmetros e, para a escolha da *técnica de regularização* que resulte em uma configuração final da rede neural, capaz de minimizar a influência do *overfitting*. O objetivo é a obtenção de um modelo que minimize o erro de generalização. Fonte: Elaborado pelos autores.

[4]Algumas imagens nas figuras 3.4 e 3.5 foram adaptadas de <`https://www.deepmind.com/blog/alphafold-using-ai-for-scientific-discovery-2020`>.

3.2.3 Validação do modelo

A validação do modelo desenvolvido baseiou-se na técnica conhecida como *validação cruzada* (do inglês, *cross validation*). A validação cruzada caracteriza-se por uma heurística utilizada na *minimização do erro de generalização* de um modelo. Geralmente, uma partição menor dos dados de treinamento é utilizada para validar o modelo, uma vez que o conjunto de validação serve apenas para definir o momento em que não é mais necessário o ajuste dos parâmetros. Nesse caso, uma vez que o modelo proposto baseia-se em uma arquitetura de rede neural profunda, o critério para interrupção do ajuste de parâmetros é o número de épocas de treinamento. Comumente, é utilizado um percentual $\gamma < 0,5$ dos dados de treinamento para validar o modelo [Duda et al., 2001, p. 483–484].

Contudo, empregou-se uma versão alternativa a essa abordagem, conhecida como *k-fold cross-validation*[Mosteller & Tukey, 1968]. Basicamente, a técnica consiste em particionar randomicamente o conjunto de treinamento em k subconjuntos mutuamente exclusivos e de mesmo tamanho (n/k), onde n é o total de amostras de treinamento. Um subconjunto é utilizado para validação e os $k-1$ restantes são utilizados para estimativa dos parâmetros. Esse processo é realizado k vezes alternando de forma circular o subconjunto de validação. O desempenho, ao final, é estimado com base na média das k taxas de erro correspondentes a cada uma das partições [Duda et al., 2001, p. 484]. Nesse caso, empregou-se um $k = 5$ pois, dessa forma, é possível garantir que $\gamma \geq 0,1$, frequentemente recomendado e eficaz na maioria das aplicações [Duda et al., 2001, p. 484].

A seleção do modelo, de acordo com a validação cruzada, baseou-se em uma abordagem cuja finalidade é a *minimização do erro de generalização*[5] [Haykin, 2001, p. 240–242]. Esse conceito pode ser compreendido da seguinte forma: considere a k-ésima classe de funções $\mathscr{F}_k$ como o conjunto dos modelos de arquitetura similar (por exemplo, perceptrons de múltiplas camadas) e, com vetores de peso $\mathbf{w}$ pertencentes a um espaço de pesos multidimensional $\mathscr{W}_k$. Uma hipótese $F(\mathbf{x}, \mathbf{w})$ pertence a $\mathscr{F}_k$ e, mapeia um vetor de entrada $\mathbf{x}$ em $\{0, 1\}$ (onde $\mathbf{x} \in \mathscr{X}$, espaço de entrada, e $d = \{0, 1\}$). Assim, $\mathscr{F}_k$ pode ser descrito matematicamente como:

$$\mathscr{F}_k = \{F(\mathbf{x}, \mathbf{w}); \mathbf{w} \in \mathscr{W}_k\}\,. \tag{3.2}$$

O erro de generalização, ϵ_g, pode ser definido por

$$\epsilon_g(F) = P(F(\mathbf{x}) \neq d), \tag{3.3}$$

[5]Essa abordagem possui uma filosofia similar a *Minimização Estrutural do Risco*, proposta por Vapnik [1991].

onde $\mathbf{x}$ é uma entrada extraída de $\mathscr{X}$ e, com probabilidade P desconhecida (amostra de teste). O objetivo é selecionar a função, ou hipótese, $F(\mathbf{x}, \mathbf{w})$ que minimiza ϵ_g.

Seja $\mathscr{T}$ um conjunto de treinamento com N amostras, descrito como

$$\mathscr{T} = \{(\mathbf{x}_i, d_i)\}_{i=1}^{N} . \tag{3.4}$$

Um parâmetro r, no intervalo de 0 a 1, determina a partição do conjunto de treinamento $\mathscr{T}$ entre o subconjunto de estimação (de tamanho $(1-r)N$ amostras) e validação (de tamanho rN amostras). O subconjunto de estimação $\mathscr{T}'$ é utilizado para treinar uma sequência de modelos, resultando nas hipóteses $\mathscr{F}_1, \mathscr{F}_2, ..., \mathscr{F}_n$. Portanto, a validação cruzada resulta na escolha

$$\mathscr{F}_{cv} = min_{k=1,2,...,v} \left\{ e_v''(\mathscr{F}_k) \right\}, \tag{3.5}$$

onde $e_v''(\mathscr{F}_k)$ corresponde ao *erro de classificação* produzido pela hipótese $\mathscr{F}_k$ quando testado sobre o subconjunto de validação $\mathscr{T}''$ [Haykin, 2001, p. 240–241].

A validação cruzada caracteriza-se por uma abordagem empírica que valida um modelo experimentalmente. Ao treinar o modelo usando validação cruzada, o erro fornece uma *estimativa* do desempenho do modelo para um conjunto de teste desconhecido [Duda et al., 2001, p. 484]. Se a taxa de erro verdadeira, mas desconhecida, do modelo for p e, se m das n' amostras de teste selecionadas aleatoriamente forem classificadas erroneamente, então m tem uma distribuição binomial

$$P(m) = \binom{n'}{m} p^m (1-p)^{n'-m} \tag{3.6}$$

Assim, a fração de amostras de teste classificadas incorretamente é exatamente a estimativa de máxima verossimilhança para p

$$\hat{p} = \frac{m}{n'} \tag{3.7}$$

A figura 3.6 mostra os intervalos de confiança de 95% em função de $\hat{p}$ e n'. Para um determinado valor de $\hat{p}$, a probabilidade é 0,95 de que o valor verdadeiro de p esteja no intervalo entre as curvas inferior e superior, marcadas pelo número n' de amostras de teste. As curvas mostram que, a menos que n' seja razoavelmente grande, a estimativa da probabilidade deve ser interpretada com cautela [Duda et al., 2001, p. 484].

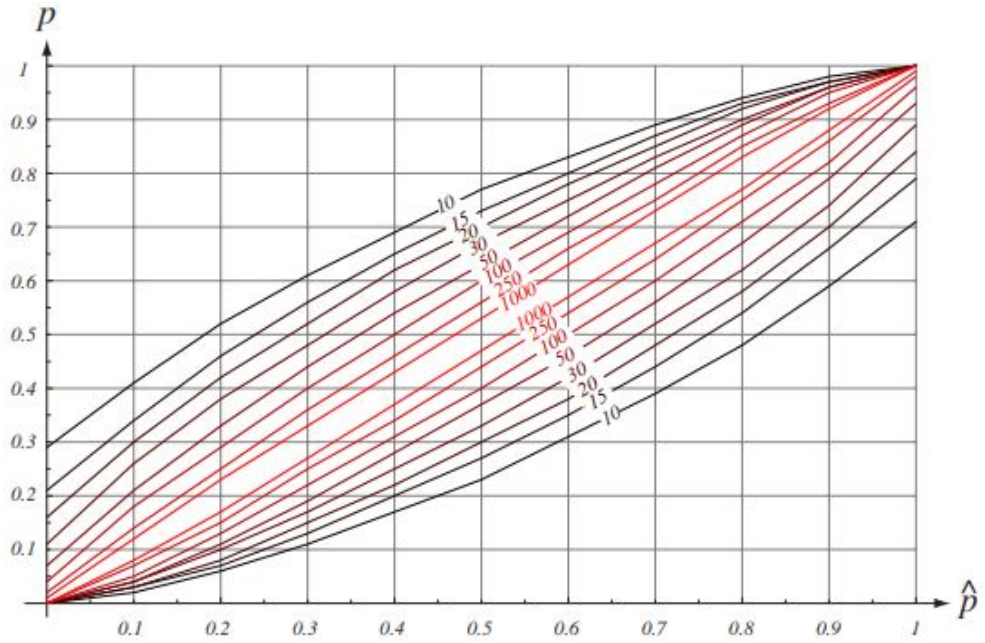


Figura 3.6. Intervalos de confiança de 95% para uma dada probabilidade de erro estimada $\hat{p}$ que podem ser derivados da distribuição binomial da equação 3.6. Por exemplo, se nenhum erro for cometido em 30 amostras de teste, com probabilidade de 0,95, a taxa de erro real está entre 10 a 15%. Por outro lado, caso o modelo não cometa nenhum erro em 1000 amostras, a taxa de erro real está abaixo de 1%. Fonte: [Duda et al., 2001, p. 485]

3.2.4 Otimização de hiperparâmetros

O processo de otimização dos hiperparâmetros pode ser realizado manualmente ou automaticamente [Goodfellow et al., 2016, p. 416]. A otimização manual é altamente dependente da *expertise* do projetista (normalmente, meses ou anos de experiência na exploração dos valores de hiperparâmetros mais adequados para redes neurais aplicadas em tarefas similares). Contudo, na maioria das aplicações, essa abordagem é inviável, sendo que comum o uso de *algoritmos automatizados* podem encontrar valores promissores de hiperparâmetros. O objetivo desses algoritmos é ajustar a *capacidade* do modelo à complexidade da tarefa bem como, encontrar um conjunto de hiperparâmetros que otimize uma função de objetivo, como o erro na validação, tempo de treinamento, memória, etc [Goodfellow et al., 2016, p. 420]. Uma das técnicas que tem obtido sucesso na otimização da topologia, bem como dos parâmetros de aprendizado, de redes neurais artificiais são os *algoritmos genéticos* [Mitchell, 1997, p. 249]

Algoritmos genéticos fornecem uma abordagem de aprendizado motivada por analogia com a evolução biológica. O objetivo é encontrar, em um *espaço de hipóteses*, a melhor hipótese para a solução de um determinado problema. Cada hipótese caracteriza um *indivíduo* pertencente ao conjunto das hipóteses candidatas, denominado *população*.

O algoritmo atualiza iterativamente a população, substituindo uma fração da população atual por hipóteses descententes mais adequadas (indivíduos mais aptos da população atual). As hipóteses *sucessoras* são produzidas pela aplicação dos operadores genéticos de *cruzamento* e *mutação*, em partes das hipóteses mais aptas. A aptidão corresponde a função de *fitness* que define o critério para classificação das hipóteses potenciais, e para seleção probabilistica à inclusão do indivíduo na população da próxima geração. Em um algoritmo genético, a melhor hipótese é aquela que otimiza a *fitness* [Mitchell, 1997, p. 249–252].

Desse modo, a seleção dos hiperparâmetros baseou-se em um espaço de busca contendo as possíveis configurações do modelo, como: *dropout* (descrito na seção 4.1), número de épocas, tamanho de lote e taxa de aprendizado[6] (tabela 3.1). O indivíduo (ou *cromossomo*) é representado por um vetor de valores reais, cujo tamanho é igual ao número de hiperparâmetros analisados. Cada posição do vetor (também denominada como *gene*) contém um valor possível ao hiperparâmetro (considerando o espaço de busca). A seleção do conjunto dos valores mais adequados para o treinamento do modelo, baseou-se na *taxa de erro* (*fitness* do problema) obtida na validação cruzada. Nesse caso, definiu-se como um potencial candidato, o conjunto dos valores de hiperparâmetros que resultassem em uma taxa de erro abaixo de 15%. Por fim, uma vez selecionados os indivíduos mais adequados, foram aplicadas operações de *cruzamento uniforme* e *mutação*, com probabilidades de 0,5 e 0,25 [Hassanat et al., 2019], respectivamente. O número de gerações para atualização da população foi igual a 10.

Tabela 3.1. Espaço de buscas dos hiperparâmetros

Hiperparâmetro	Intervalo de valores
Dropout	$\{0.25, 0.3, 0.5\}$
Número de Épocas	$\{50, 100, 150, 200\}$
Tamanho de lote	$\{32, 64\}$
Taxa de aprendizado	$\{0.001, RLROP, 0.1\}$

A figura 3.7 apresenta a evolução dos indivíduos no decorrer das gerações. Observa-se que a aplicação dos operadores genéticos contribui à seleção do melhor indivíduo ou

[6]Constante positiva usada para moderar o grau em que os pesos da rede neural são atualizados [Mitchell, 1997, p. 88]. Normalmente é definido um valor pequeno (0.1, por exemplo) e, por vezes, pode ser configurada para reduzir quando uma métrica monitorada não progride (conhecido como *Reduce Learn Rate On Plateau*, RLROP) [Chollet et al., 2015].

seja, pelo conjunto dos hiperparâmtros do modelo que resultem na menor taxa de erro. Nesse caso, a combinação que resultou em um erro de 1%, 10ª geração, possui uma taxa de dropout de 0,5, sendo o modelo treinado por 150 épocas, com um BS igual a 64 e utilizando uma taxa de aprendizado igual a 0,001.

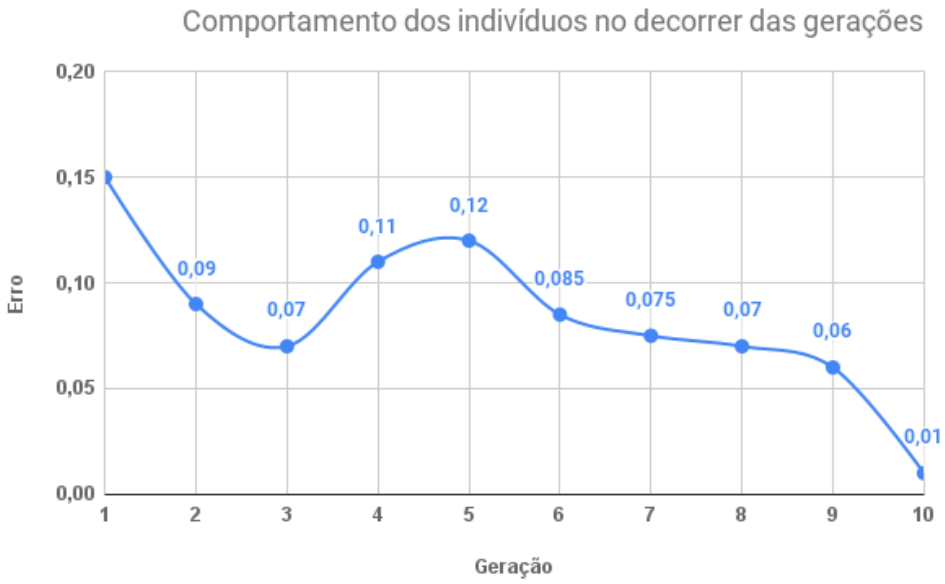


Figura 3.7. Comportamento dos indivíduos no decorrer das gerações. Fonte: Elaborada pelos autores.

3.2.5 Avaliação do modelo quanto à performance

Para avaliar o desempenho de um classificador é importante considerar três aspectos principais: o primeiro diz respeito a *discriminabilidade* do modelo ou seja, o quão bem ele classifica dados não vistos, sendo frequentemente utilizada a *taxa de erro*. o segundo refere-se a *confiabilidade* a qual está relacionada a capacidade do modelo estimar as probabilidades *a posteriori* de pertencimento à classe. Por fim, considera-se a curva característica de operação do receptor (do inglês *Receiver Operating Characteristic Curve*, ROC) como indicador de performance [Webb & Copsey, 2011, p. 405].

Todavia, a probabilidade de erro, por vezes, não é suficiente para avaliar o desempenho do modelo [Theodoridis & Koutroumbas, 2009, p. 573]. Ao utilizar a taxa de erro assume-se que todas as classificações incorretas possuem o mesmo custo, ocultando informações relevantes sobre o comportamento do classificador [Provost et al., 1998]. Assim, a avaliação de um modelo pode ser realizada a partir da comparação de performance com múltiplos classificadores treinados com o mesmo conjunto de dados. Para isso, são necessárias métricas complementares que auxiliem na escolha do "classificador ótimo", sob

a perspectiva do desempenho. Estas métricas podem ser extraídas a partir da matriz de confusão [Webb & Copsey, 2011, p. 405].

Uma matriz de confusão $A = [A(i, j)]$ pode ser definida de modo que, seu elemento A(i,j) seja o número de amostras, cujo rótulo da classe verdadeira foi j e foram preditos como classe i [Theodoridis & Koutroumbas, 2009, p. 573]. A tabela representa uma matriz de confusão para um problema de classificação binária. Os verdadeiros positivos (VP) consistem na quantidadede de amostras da classe positiva, que foram corretamente preditas pelo classificador. Os falsos positivos (FP), por sua vez, consistem na quantidade de amostras da classe negativa, que foram incorretamente preditas pelo classificador [Webb & Copsey, 2011, p. 405]. A soma do total de verdadeiros positivos e falsos positivos resulta na quantidade total de amostras positivas (P), conforme abaixo:

$$P = VP + FP \tag{3.8}$$

Os verdadeiros negativos (VN) consistem na quantidadede de amostras da classe negativa, que foram corretamente preditas pelo classificador. De forma complementar, os falsos negativos (FN), consistem na quantidade de amostras da classe positiva, que foram incorretamente preditas pelo classificador [Webb & Copsey, 2011, p. 405]. A soma do total de verdadeiros negativos e falsos negativos resulta na quantidade total de amostras negativas (N), conforme abaixo:

$$N = VN + FN \tag{3.9}$$

No contexto do problema de identificação da variante, as amostras positivas referem-se a classe *selvagem* e, as amostras negativas referem-se a classe *mutada*. Quando a hipótese H_0 (corresponde a mutação) for verdadeira e o modelo produzir a classe mutada, isso quer dizer que o classificador está realizando a inferência correta de não rejeitar a H_0. Na teoria de teste de hipóteses, o detector tomou uma decisão negativa (por não rejeitar H_0) e verdadeira (classe correta), representando a medida verdadeiro positivo.

A performance do modelo baseia-se na obtenção das métricas extraídas a partir da matriz de confusão. Como se trata de um problema de classificação binária, as relações na tabela 3.2 podem ser utilizadas, sem prejuízo à interpretação do modelo quanto à desempenho [Webb & Copsey, 2011, p. 405], conforme descrito a seguir:

1. Acurácia (Acc): percentual de amostras da classe j que foram corretamente classificadas. Dado um problema de m classes e um conjunto de teste com n amostras, a acurácia pode ser calculada, a partir da matriz de confusão M, de dimensões $m \times m$, por [Theodoridis & Koutroumbas, 2009, p. 573]:

$$Acc = \frac{1}{n}\sum_{j=1}^{m} M(j,j). \tag{3.10}$$

Para o caso de um problema de 2 classes, a acurácia pode ser calculada com base na relação da tabela 3.2.

2. Erro: probabilidade de classificar erroneamente uma amostra selecionada aleatoriamente. É a taxa de erro em um conjunto de teste extraído da mesma distribuição dos dados de treinamento.

3. Revocação: percentual de amostras pertencentes a classe positiva que foram corretamente classificadas, em relação ao total de amostras dessa classe.

4. Especificidade: percentual de amostras pertencentes a classe negativa que foram corretamente classificadas, em relação ao total de amostras dessa classe.

5. Precisão: percentual de amostras pertencentes a classe positiva que foram corretamente classificadas, em relação ao total de amostras positivas preditas.

Tabela 3.2. Métricas de desempenho derivadas de uma matriz de confusão para um problema de classificação binária

Acurácia (Acc)	$\frac{VP+VN}{P+N}$
Taxa de erro (e)	$1 - Acc$
Taxa de falsos positivos	$\frac{FP}{FP+VN}$
Precisão ($Prec$)	$\frac{VP}{VP+FP}$
Revocação (Rec)	$\frac{VP}{VP+FN}$
Especificidade (Esp)	$\frac{VN}{VN+FP}$
F1-*score*	$\frac{2}{\frac{1}{Prec}+\frac{1}{Rec}}$

Relação de métricas de desempenho empregadas na avaliação de modelos de classficação binária. Cada métrica pode ser extraída a partir da matriz de confusão. Adaptado de [Webb & Copsey, 2011, p. 405].

Conforme mencionado anteriormente, existem algumas limitações quanto ao uso da taxa de erro para mensurar o desempenho de redes neurais [Webb & Copsey, 2011, p. 413].

Contudo, a partir dos rótulos preditos, ainda é possível obter a Taxa de Verdadeiros Positivos (do inglês, *True Positive Rate* - TPR), ou Revocação, e a Taxa de Falsos Positivos (do inglês, *False Positive Rate*, FPR).

Com essa informação, utilizou-se uma representação gráfica complementar denominada Curva Característica de Operação do Receptor (do inglês, *Receiver Operating Characteristic Curve*, Curva ROC), que ilustra a performance da regra de discriminação binária. A curva ROC fornece um método confiável à *seleção do limiar de operação* do modelo [Webb & Copsey, 2011, p. 415].

Nesse caso, a medida que a TPR aumenta (e a FPR diminui), a taxa de erro do modelo diminui [Webb & Copsey, 2011, p. 416]. Além disso, conhecendo a FPR, é possível obter a *Especificidade*, calculando a taxa de verdadeiros negativos (*True Negative Rate*, TNR) a partir da relação

$$TNR = 1 - FPR. \tag{3.11}$$

Modelos com uma capacidade discriminativa desejável devem apresentar, para um determinado limiar de operação, uma alta TPR e TNR.

Capítulo 4

Discussões e Resultados

4.1 Análise Preliminar das Arquiteturas

Uma vez que o problema de identificação da variante encontra-se na categoria de *IA-completo*[1] [Shapiro, 1992, p. 54—57], uma abordagem comum é utilizar um *baseline default* de aprendizado profundo [Goodfellow et al., 2016, p. 413]. Considerando que a estrutura do dado de entrada (mapas de distâncias) é topológica, tornou-se possível o emprego de uma ConvNet [LeCun et al., 1989]. Nesse sentido, a arquitetura da rede convolucional foi desenvolvida como um arranjo de camadas em uma estrutura de cadeia, onde cada camada torna-se uma função da camada que a procede [Goodfellow et al., 2016, p. 191].

Também é necessária a definição de uma função de ativação apropriada assim como um algoritmo de otimização para treinamento, sendo empregados a função ReLU (detalhada na subseção 2.5.1) e o otimizador *Adam* (do inglês, *Adaptive Moment Estimation*, Adam) [Kingma & Ba, 2014], respectivamente [Goodfellow et al., 2016, p. 413].

Com base nessas definições iniciais, foram implentadas duas arquiteturas: uma ConvNet e a VGGNet. A ConvNet contém 4 (quatro) camadas convolucionais, de forma que a primeira camada possui um total de 32 filtros, e as demais têm o dobro do número de filtros da camada anterior. A classificação é realizada por uma rede totalmente conectada de duas camadas, cujo número de unidades neuronais por camada é 1024 e 256. A ativação do neurônio de saída é realizada pela função sigmóide (subseção 2.5.1). Para a ConvNet, foram implementadas duas variações, onde uma contém *normalização por lote* (do inglês, *batch normalization*, BN), e outra utiliza *Dropout*. No que tange a VGG, a

[1] No campo da inteligência artificial, problemas difíceis são conhecidos como IA-completos. Isso significa, que a dificuldade desses problemas (assumindo que a inteligência é computacional) é equivalente à resolver um problema central de inteligência artificial – tornando computadores tão inteligentes quanto as pessoas, ou IA forte [Shapiro, 1992, p. 54—57].

implementação baseou-se na arquitetura VGG-13 (figura 2.16).

Os modelos foram implementados no *Colab*, ambiente virtual na nuvem da Google que possibilita acesso a um *Jupyter notebook*. Em relação às características do hardware usado, o ambiente disponibiliza um processador *dual core*, com 13.6 GB de memória RAM e cache L3 de 40-50 MB. Contudo, foi utilizada uma GPU NVIDIA Tesla P100, com 16GB de memória, como acelerador. A tabela 4.1 apresenta a performance dos modelos a partir das seguintes métricas calculadas: Acurácia, Erro, e F1-Score[2].

Tabela 4.1. Desempenho dos modelos empregando 5-fold CV, com destaque à partição na qual obteve-se o melhor desempenho em cada ensaio e à média considerando as K partições

Partição	Acurácia	Erro	F1-Score
1	0.86	0.33	0.89
2	**0.93**	**0.17**	**0.93**
3	0.91	0.20	0.91
4	0.90	0.27	0.89
5	0.85	0.46	0.87
	0.89	**0.29**	**0.90**

(a) ConvNet usando regularização via *Dropout*

Partição	Acurácia	Erro	F1-Score
1	**0.90**	**0.28**	**0.88**
2	0.82	0.39	0.83
3	0.83	0.41	0.84
4	0.79	0.25	0.80
5	0.85	0.21	0.89
	0.84	**0.31**	**0.85**

(b) VGG empregando normalização por lote

O Dropout [Srivastava et al., 2014] consiste em uma técnica de regularização de fácil implementação e compatível com diversos modelos e algoritmos de treinamento, a qual consiste em treinar um conjunto de todas as sub-redes que podem ser formadas (figura 4.1), removendo as unidades neuronais (exceto a de saída), a partir de uma de uma rede base subjacente. A maioria das implementações de Dropout removem uma unidade neuronal multiplicando o valor de saída dessa unidade por zero [Goodfellow et al., 2016, p. 251]. Essa remoção é baseada em um hiperparâmetro denominado *probabilidade de retenção da unidade neuronal*, p. de acordo com Srivastava et al. [2014], valores de p na faixa de $[0,3 - 0,6]$ tendem a produzir uma redução significativa na taxa de erro no treinamento. Também é recomendável o Dropout para problemas cuja bases de dados sejam de tamanho superior a $5K$. Em bases de dados de menor tamanho, o Dropout não proporciona melhorias efetivas [Srivastava et al., 2014].

Por sua vez, a normalização de lote foi inicialmente proposta por Ioffe & Szegedy [2015] para minimizar o problema de *deslocamento interno da covariável*. Em virtude da atualização dos pesos da rede durante o treinamento, podem ocorrer mudanças na distribuição das ativações, o que impacta na convergência da rede. Assim, de modo a promover a estabilização da rede neural, sem ajustes a taxa de aprendizado, a técnica

[2]F1-Score: é uma métrica empregada na avaliação de sistemas de classificação binária, sendo definida como a média harmônica da precisão e do *recall* [Webb & Copsey, 2011, p. 405].

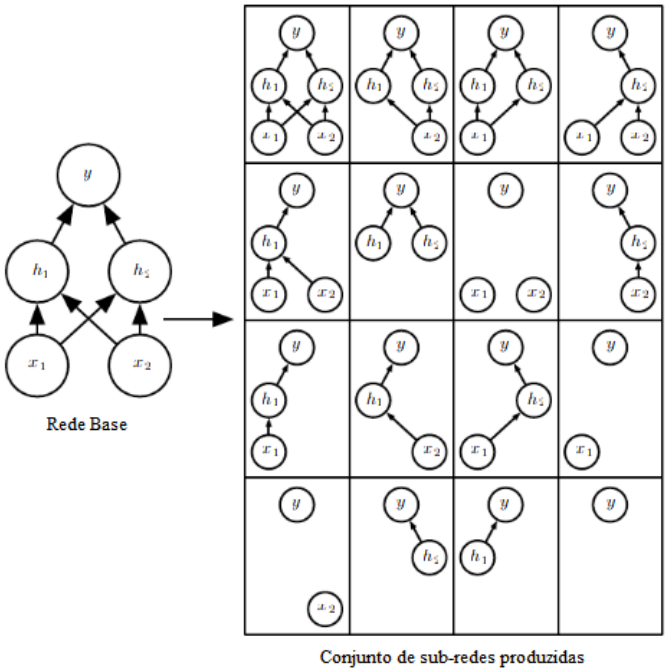


Figura 4.1. Dropout treina um conjunto de todas as sub-redes que podem ser formadas, removendo as unidades neuronais (exceto a de saída), a partir de uma de uma rede base subjacente. Na figura, uma rede básica com duas unidades de entrada e duas unidades ocultas. Assim, existem 16 (dezesseis) sub-redes possíveis, que podem ser formadas removendo diferentes unidades da rede original. Fonte: Goodfellow et al. [2016]

aplica a normalização das entradas, subtraindo pela sua média e devio padrão, com base no mini-lote [Goodfellow et al., 2016, 309–310]. Como consequência, há uma aceleração do treinamento [Schilling, 2016], sendo posivel o uso de taxas de aprendizado maiores. A normalização por lote é comum em implementações de redes totalmente conectadas e convolucionais, diferindo para cada uma, e possui um efeito regularizador, permitindo a omissão do Dropout[3] [Ioffe & Szegedy, 2015; Goodfellow et al., 2016, p. 413]. Em termos de implementação, Ioffe & Szegedy [2015] sugerem introduzir a normalização por lote antes da ativação.

Como forma de ilustrar o comportamento das arquiteturas implementadas, a figura 4.2 apresenta o desempenho de aprendizado dos modelos no treinamento e na validação, a partir de *curvas de aprendizado*. Em relação a tabela 4.1, foi adicionada a implementação da ConvNet empregando apenas a normalização por lote. Nesse caso, é perceptível que a taxa de erro permanece elevada, sugerindo a ocorrência de *overfitting* [Duda et al., 2001, p. 134–135]. Também foi adicionada uma camada regularizadora conhecida como *Spatial*

[3]Em um estudo recente, Li et al. [2019] constataram que o uso da abordagem clássica de Dropout combinada à normalização por lote em ConvNets, pode impactar o treinamento e teste do modelo, acarretando em um efeito indesejado conhecido por *Deslocamento de Variância*

Dropout 2D [Tompson et al., 2014] nas camadas iniciais da ConvNet. Dessa forma, o desempenho do modelo alcançou uma melhora significativa em relação a variação com o Dropout.

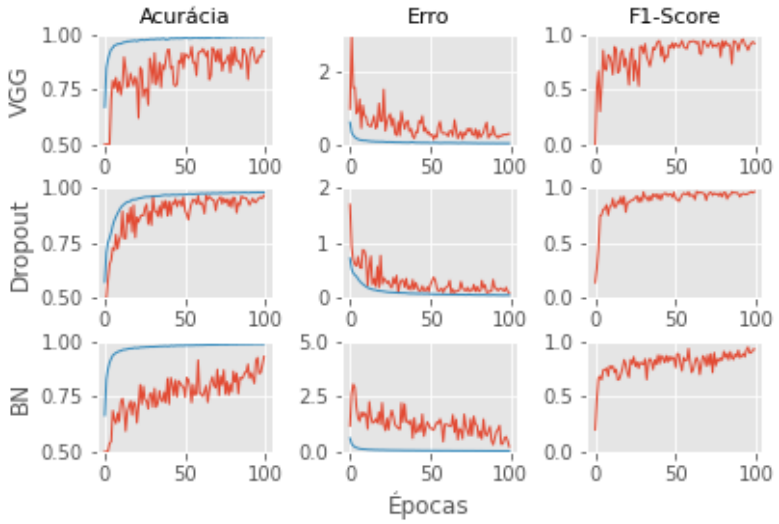
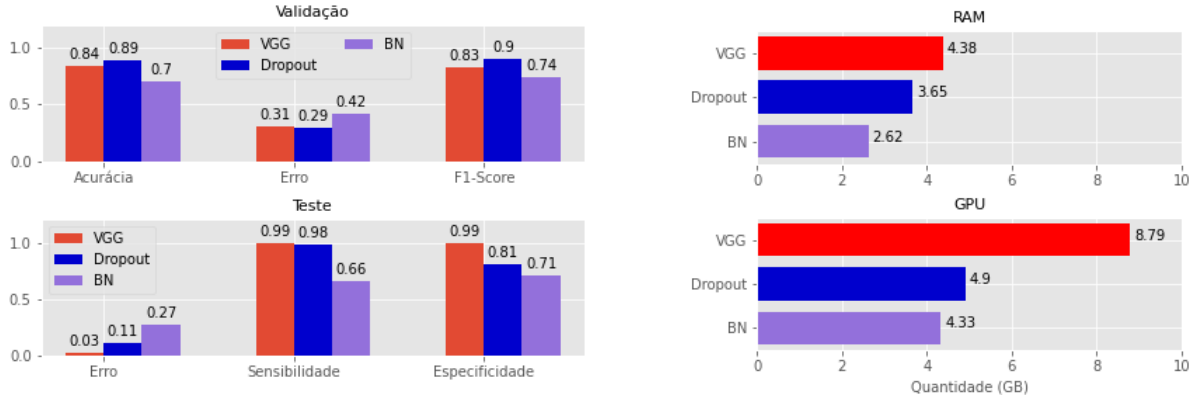


Figura 4.2. Desempenho de aprendizado dos modelos no Treinamento (*azul*) e na Validação (*laranja*), com base nas *curvas de aprendizado*. Fonte: Elaborada pelos autores.

Como os hiper-parâmetros da rede são ajustados na validação cruzada, o modelo é selecionado com relação a performance sobre os dados de teste. Nesse caso, segue-se uma abordagem na qual seleciona-se o modelo que minimize o erro de generalização [Haykin, 2001, p. 240–242]. A figura 4.3a ilustra um comparativo da performance dos modelos na validação e no teste. É possível verificar que a VGG-13 obteve uma taxa de erro inferior as ConvNets, sobre mapas de distâncias advindos do conjunto de teste.

Contudo, quando analisamos a quantidade de recursos utilizados no treinamento da VGG, observa-se que o modelo demanda quase o dobro da quantidade de memória RAM e GPU, quando comparado às implementações da ConvNet (figura 4.3b). Isso deve-se ao número de camadas convolucionais da VGG, o que a torna mais densa em relação à ConvNet, possibilitando o reconhecimento de padrões complexos em representações de imagem [Simonyan & Zisserman, 2014]. O tempo médio de treinamento, considerando a validação cruzada, foi de 3 horas e 26 minutos.

Conforme mencionado na subseção 3.2.5, existem algumas limitações quanto ao uso da taxa de erro na representatividade das classes de um determinado problema [Webb & Copsey, 2011, p. 413]. Nesse caso, utiliza-se a *Curva ROC*, que fornece um método confiável à seleção de um limiar de decisão adequado [Webb & Copsey, 2011, p. 415]. A figura 4.4 apresenta a Curva ROC à implementação da VGG, onde é possível analisar a relação entre a taxa de verdadeiros e falsos positivos. Nesse caso, uma taxa de verdadeiros



(a) **(b)**

Figura 4.3. Performance das variações do modelo com relação aos dados do conjunto de validação e teste. Em (a) é possível verificar a diferença de desempenho entre as arquiteturas para ambos conjuntos de dados. Em (b), é possível verificar a quantidade de recursos requisitados pelos modelos no treinamento. Fonte: Elaborada pelos autores.

positivos alta (combinada a uma taxa de falsos positivos baixa), corresponde a uma taxa de erro na classificação baixo [Webb & Copsey, 2011, p. 416](figura 4.3b).

A partir da curva ROC é possível obter a especificidade, calculando a taxa de verdadeiros negativos (conforme descrito ao final da subseção 3.2.5). Observa-se que a taxa de verdadeiros positivos é elevada, implicando em uma taxa de falsos negativos baixa (figura 4.4). Assim, pode-se concluir que, a VGG conseguiu distinguir as classes do problema, *selvagem* e *mutada*, com elevadas sensibilidade e especificidade, respectivamente, podendo ser visualizado na figura 4.3a.

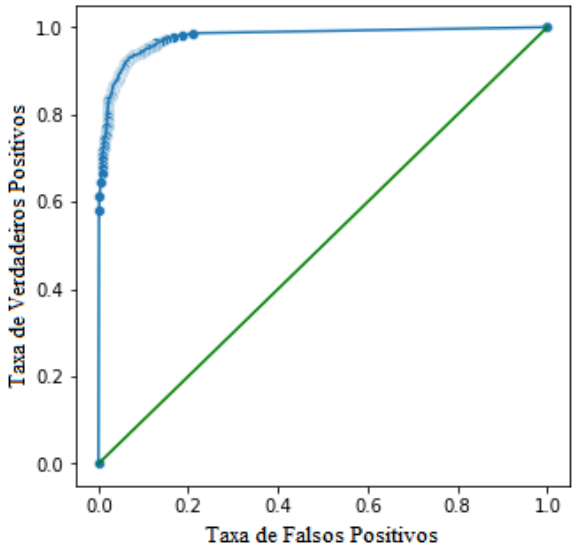


Figura 4.4. Curva ROC aos modelos, onde é possível analisar a relação entre a taxa de verdadeiros e falsos positivos. Fonte: Elaborada pelos autores.

4.2 Performance do Modelo às Mutações da Base de Dados

Com base nos resultados obtidos através das arquiteturas, selecionou-se a VGG para avaliar o desempenho das redes convolucionais com as demais variantes da base de dados. Na seção 4.1, constatou-se que a VGG conseguiu distinguir as classes *selvagem* e *mutada*. Contudo, as classes do problema são conhecidas (ou seja, a classe *mutada* corresponde a mutação S:N501Y, característica a variante B.1.1.7, ou Alfa).

Assim, em um primeiro ensaio, o modelo foi avaliado com base em um conjunto de teste formado pelos mapas de distâncias correspondentes às mutações conhecidamente similares, no que tange a infectividade, à variante B.1.1.7 (subseção 2.3.3). Nesse caso, esses mapas não *são apresentados ao modelo durante o treinamento*, embora seja conhecida a capacidade infectante das mutações. O objetivo foi verificar se o modelo é capaz de predizer o potencial de infectividade dessas mutações, com base nas variações dos movimentos intrínsecos da estrutura (ou nos padrões de distâncias interatômicas) nos mapas de distâncias.

De forma análoga, o modelo também foi avaliado para um conjunto de teste contendo mapas que correspondem a dinâmica do RBD com mutações de efeito neutro, em relação a afinidade com o receptor humano (ou a resistência ao sistema imune). A figura 4.5 apresenta a *precisão* do modelo para cada sistema (alguns contendo mutações responsáveis por variantes de maior infectividade) pertencente a base de dados.

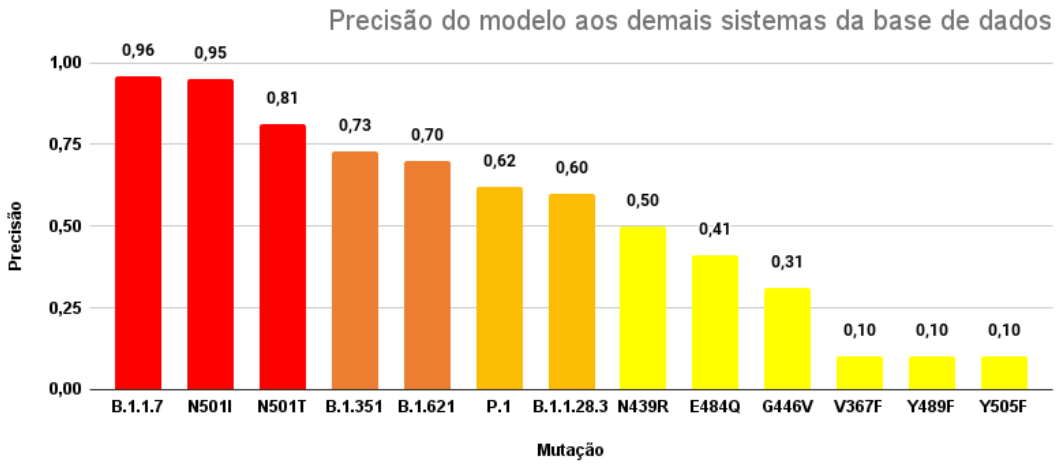


Figura 4.5. A precisão do modelo é representada em um formato percentual, considerando duas casas decimais. Variantes às quais a precisão do modelo é elevada, estão relacionadas a uma maior infectividade na literatura. No entanto, mutações ditas neutras, não estão relacionadas à maior afinidade ao receptor humano, ou a mudanças na energia livre de ligação. Fonte: Elaborada pelos autores.

Os resultados preliminares sugerem que a rede neural conseguiu discriminar mutações de maior infectividade como a B.1.351 (Beta), B.1.621 (Mu), P.1 (Gama), e a B.1.1.28.3 (Teta) com uma precisão na faixa de 60,0% a 73,0%. Nesse caso, o modelo rejeita a classe *selvagem*, indicando que o comportamento da dinâmica é mais similar a classe *mutada* (em função de uma maior precisão). Por outro lado, para mutações de caráter neutro, a precisão encontra-se próxima, ou abaixo, de 50,0% indicando uma dubiedade na decisão da classe. De modo análogo, o modelo rejeita a classe *mutada* predizendo que os mapas de distância correspondem a classe *selvagem*. Esse resultado está de acordo com a literatura da área, até então, uma vez que não foram observadas mudanças significativas na energia livre de ligação, para mutações neutras [Teng et al., 2020; Chen et al., 2021b].

Em um segundo ensaio, avaliou-se a performance do modelo para variantes emergentes (*emerging variants*, EV). Assim, foram necessárias simulações adicionais, referentes às mutações N501I e N501T. Estudos recentes, sugerem que essas mutações são responsáveis por proporcionar um aumento de afinidade da proteína S com o receptor humano e, mudanças conformacionais na estrutura [Verma & Subbarao, 2021; Khan et al., 2022]. Nesse caso, o modelo alcançou uma precisão de 95,0% e 81%, respectivamente (figura 4.5).

Por fim, é possível empregar a mesma abordagem para avaliar um modelo treinado, a partir de dinâmicas subamostradas com um *skip* igual a 10 frames (figura 4.6). Com base nos resultados obtidos, é possível observar que não houve alteração significativa no desempenho do modelo com a alteração do tamanho do *skip*.

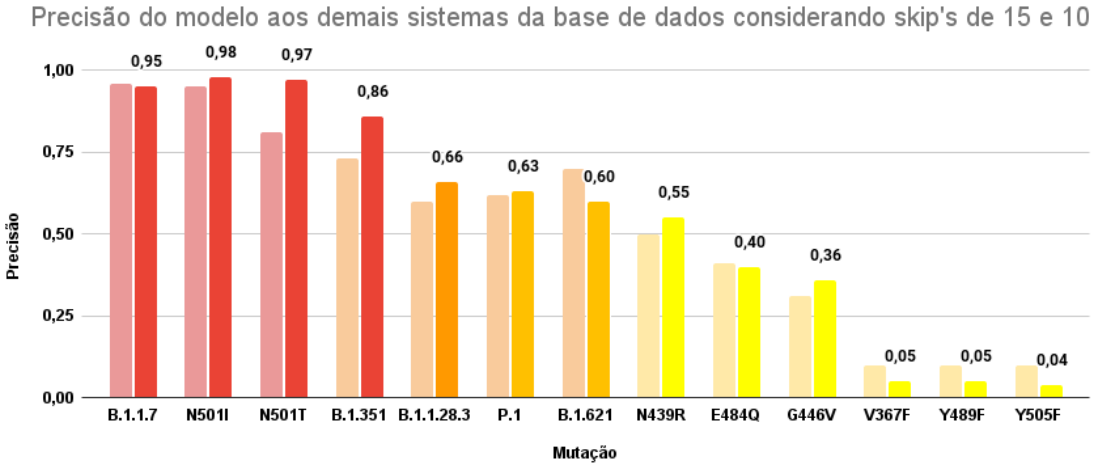


Figura 4.6. A precisão do modelo em formato percentual, considerando um *skip* de 10 frames, é destacada por barras com intensidade de cor maior. Verifica-se a similaridade dos resultados para um *skip* de 15 frames, em barras com intensidade de cor menor (figura 4.5). Fonte: Elaborada pelos autores.

4.3 Relação entre a Precisão do Modelo e o $\Delta\Delta G$

Conforme descrito na subseção 2.3.3, mudanças na energia livre de ligação, $\Delta\Delta G$, induzidas por mutações no RBD da proteína S, têm sido frequentemente relacionadas à infectividade do SARS-CoV-2 [Chen et al., 2021b; Hoffmann et al., 2020; Teng et al., 2020; Walls et al., 2020]. Assim, tendo em vista essas constatações, procurou-se estabelecer uma relação entre a *precisão do modelo*, às mutações da base de dados, e o $\Delta\Delta G$. O objetivo foi buscar identificar correlações possíveis, entre a capacidade preditiva do modelo e mudanças na energia livre de ligação. A tabela 4.2 apresenta a relação de mutações no RBD e o $\Delta\Delta G$ (kcal/mol) correspondente.

Tabela 4.2. Relação de mutações no RBD referentes à variantes de preocupação (VOC) e de interesse (VOI) do SARS-CoV-2, nomeadas pela Organização Mundial de Saúde [OMS, 2022], bem como variantes emergentes (EV). O $\Delta\Delta G$ (kcal/mol) foi obtido com base em resultados obtidos por Chen et al. [2021a] e Verma & Subbarao [2021]. Para mutações pontuais, os valores foram extraídos da aplicação *Mutation Analyzer*, disponível em `https://weilab.math.msu.edu/MutationAnalyzer/` [Chen et al., 2021a]. Foi realizada a normalização dos valores de $\Delta\Delta G$, escalados para uma faixa de $[0-1]$, conforme consta na coluna $\Delta\Delta G_{norm}$.

Variante	Mutação no RBD	$\Delta\Delta G$ (kcal/mol)	$\Delta\Delta G_{norm}$
B.1.1.7	[N501Y]	0,55	0,74
B.1.351	[K417N,E484K,N501Y]	0,81	1,0
P.1	[K417T,E484K,N501Y]	0,66	0,85
B.1.621	[R346K,E484K,N501Y]	0,77	0,95
B.1.1.28.3	[E484K,N501Y]	0,65	0,84
EV	[N501I]	0,80	0,99
EV	[N501T]	0,45	0,64
-	[N439R]	-0,18	0,00
-	[E484Q]	0,01	0,19
-	[G446V]	0,15	0,33
-	[V367F]	0,17	0,35
-	[Y489F]	-0,14	0,04
-	[Y505F]	0,17	0,35

Com base nos valores de *precisão do modelo* da figura 4.5 e $\Delta\Delta G$ da tabela 4.2, foi possível desenvolver o gráfico de dispersão da 4.7. Inicialmente, o $\Delta\Delta G$ foi normalizado para uma faixa de $[0-1]$ (ou seja, a mesma em que se encontram os valores de precisão do modelo), $\Delta\Delta G_{norm}$, utilizando a técnica *Min-Max*. Cada coordenada no gráfico apresentado na figura 4.7, representa uma enupla contendo a precisão do modelo e, o valor de $\Delta\Delta G_{norm}$ correspondente a uma determinada mutação da base de dados. Nesse caso, é possível observar a formação de duas regiões características: uma contendo os pontos relativos às mutações neutras (onde, $\Delta\Delta G_S \approx \Delta\Delta G_M$) e, outra contendo mutações frequentemente associadas a uma maior infectividade viral. Ao final, estimou-se a *correlação produto-momento* (ρ) [Pearson, 1896] entre o conjunto de valores relativos à precisão e ao $\Delta\Delta G_{norm}$, resultando em um valor de ρ igual a 0,77.

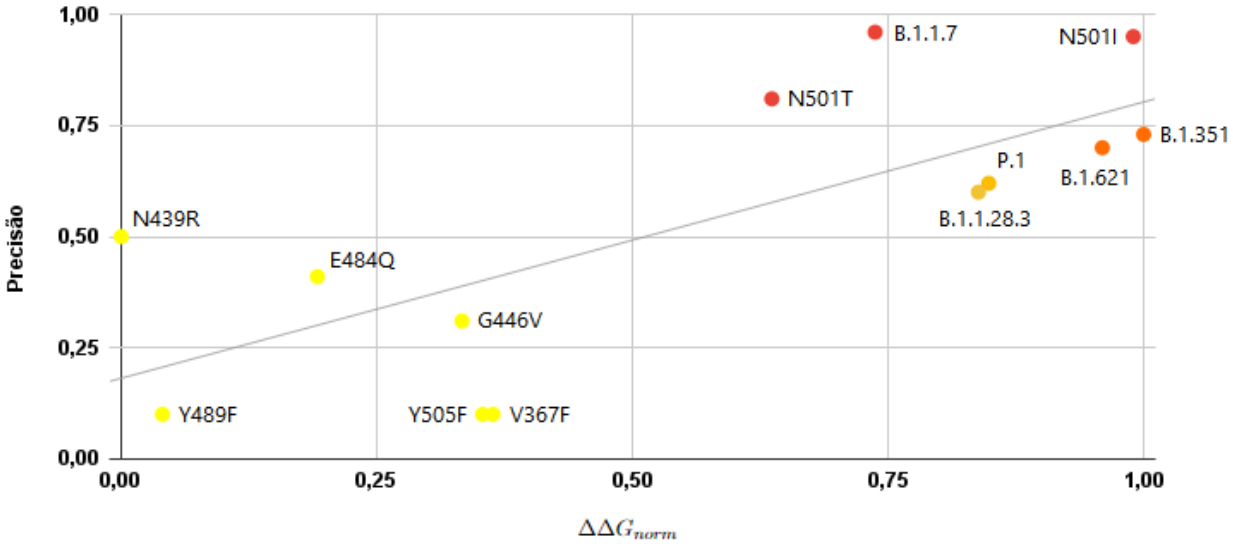


Figura 4.7. Relação entre a precisão obtida a partir do modelo e as mudanças na energia livre de ligação ($\Delta\Delta G_{norm}$), com base nas mutações da base de dados. Também é possível observar a linha de tendência, sendo estimada uma *correlação produto-momento* $\rho = 0,77$. Fonte: Elaborada pelos autores.

A partir da figura 4.7 é possível observar que, dinâmicas para as quais obtém-se uma precisão elevada estão associadas a mutações em regiões de interação da proteína S com o receptor hACE2, onde substituições de aminoácidos tendem a ocasionar mudanças na mobilidade natural da estrutura. Essas mutações possuem valores de $\Delta\Delta G$ relativamente maiores que as mutações neutras, conforme apresentado na tabela 4.2. Assim, embora não seja possível estabelecer uma relação direta entre a *precisão do modelo* e o $\Delta\Delta G$, a correlação entre essas variáveis indica que a *performance do modelo* (para o conjunto de mutações da base de dados) tende a convergir com resultados encontrados na literatura [Chen et al., 2021b; Hoffmann et al., 2020; Teng et al., 2020; Walls et al., 2020].

4.4 Visualizando Recursos

Conforme descrito na subseção 2.5.4, as redes convolucionais conseguem aprender padrões abstratos a partir de imagens brutas. Uma matriz de pixels, por exemplo, não é uma entrada *adequada* para uma SVM, sendo necessária a definição de um *vetor de características*. Nesse caso, cada imagem é representada por um conjunto de atributos como cor, textura, domínio da frequência, etc (processo conhecido como *engenharia de recuros*). Em ConvNets, esse processo é abstraído pelas camadas convolucionais, que extraem recursos de alto nível e complexos, de acordo com o *representation learning* (subseção 3.2.2). A abordagem na qual pretende-se explicitar esses recursos aprendidos é referida como *visualização de recursos* [Molnar, 2022].

Uma visualização 2D comumente utilizada em aplicações com ConvNets é o *mapa de recursos* (do inglês, *feature maps* ou, também conhecido como *channels*), que resulta das saídas de camadas convolucionais. Essa representação também é referida na literatura como *unidade* ou, mais especificamente, *convolution channel*[4] [Molnar, 2022; Goodfellow et al., 2016, p.322–323]. Por padrão, as primeiras camadas focam no aprendizado de características como bordas e texturas. As camadas ocultas extraem padrões de maior complexidade, enquanto as últimas camadas na identificação de partes específicas da imagem, como segmentos de curva ou objetos [Molnar, 2022].

Nesse sentido, considerando que o modelo do presente trabalho é treinado a partir do conjunto dos mapas de distâncias, referentes à trajetória da estrutura ao longo do tempo, é possível obter um mapa de recursos correspondente, que exiba as *regiões* do mapa de distâncias mais significativas à decisão do modelo. A figura 4.9 apresenta o complexo RBD-hACE2 do SARS-CoV-2 (PDB ID 6M0J) nas figuras 4.9b e 4.9d, que tem como destaque o RBD da proteína S em uma representação tridimensional do mapa de recursos 2D à esquerda (figuras 4.9a e 4.9c). O mapa de recursos resulta da primeira camada convolucional, a partir de um mapa de distâncias referente ao *frame* 4500 das trajetórias das trajetórias do tipo *selvagem*, ou *mutada*. A princípio, objetiva-se identificar quais *regiões* do mapa de recursos são mais significativas à decisão do modelo e, analisar a relação delas com as mutações.

É possível observar no complexo que, os resíduos correspondentes ao RBD da proteína S (especificamente do RBM) apresentam diferentes *intensidades de cor*. A definição dos valores segue uma abordagem que, calcula o maior valor das componentes RGB, para cada resíduo (333 ao 526), considerando o mapa de recursos correspondente. Assim, o

[4] As camadas, como uma unidade, são empregadas pelo DeepDream do Google, que adiciona repetidamente os mapas de recursos resultantes das camadas convolucionais a imagem original, produzindo em uma versão onírica da entrada [Molnar, 2022].

RBD fornece um retrato do mapa de recursos, com destaque às regiões onde ocorre a maximização da ativação. Nesse caso, as regiões compreendem os resíduos 477 a 487 e, 498 a 502. Conforme descrito na subseção 2.3.3, algumas variantes de preocupação como a Alfa (B.1.1.7), Beta (B.1.351), Gama (P.1), Omicron (BA.1, BA.2, BA.4-5 e BA.2.12.1) e de interesse como a Mu (B.1.621) e a Teta (B.1.1.28.3), compartilham mutações em comum como a S:N501Y e a S:E484K. Essas variações entre as intensidades de cor nas figuras 4.9b e 4.9d, podem estar relacionadas com mudanças nos padrões de distâncias, nas regiões onde encontram-se essas mutações.

Conforme descrito na subseção 2.2.2, o cálculo do RMSF é uma abordagem comum em dinâmica molecular à identificação de regiões de mobilidade em uma estrutura. A figura 4.8 ilustra as curvas RMSF para o RBD da proteína S, sendo uma correspondente a estrutura com a mutação S:N501Y enquanto a outra, a estrutura selvagem. É possível observar que a região que compreende os resíduos 498 a 502, da curva correspondente à mutação, apresenta valores de RMSF próximos a $1,5$Å ou seja, relativamente maiores que os valores da curva referente à selvagem.

Esse comportamento tem sido constatado em trabalhos recentes, como em Verma & Subbarao [2021], que observaram uma maior magnitude de movimento no RBD de um complexo RBD-hACE2 com as mutações S:N501Y e S:N501I. Nesse caso, o RBM possuía movimentos de elevada amplitude em direção ao receptor humano, indicando um aumento da afinidade de ligação Verma & Subbarao [2021]. Alaofi & Shahid [2021] também apontam que, a região do RBM possui regiões de loop associadas a conformações favoráveis à ligação. Em seu estudo, o RBD com a mutação S:N501Y apresentou confórmeros no RBM que variaram significativamente na região de loop 498-502 (ligeiramente flexível) durante simulações de dinâmica molecular de 100 ns.

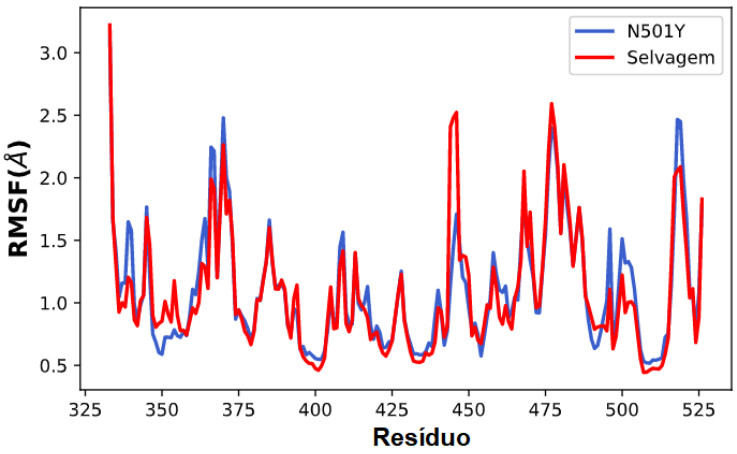
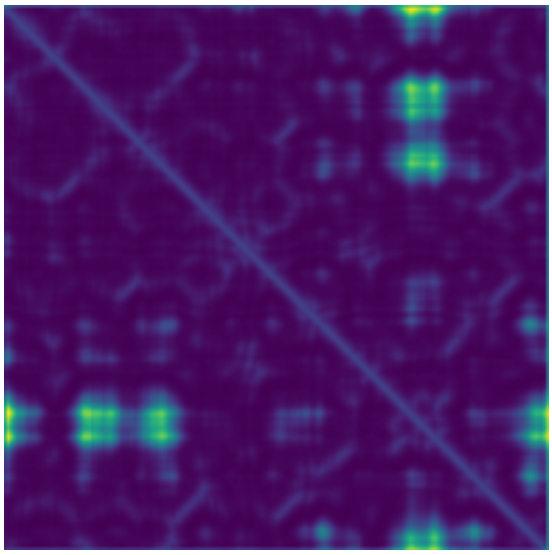
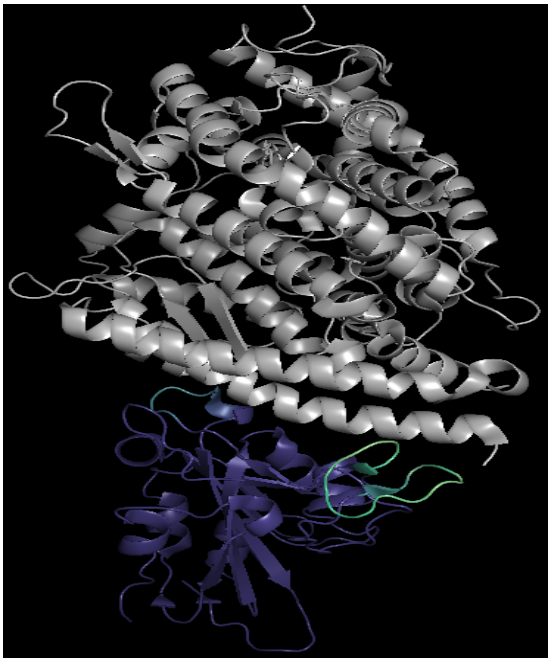


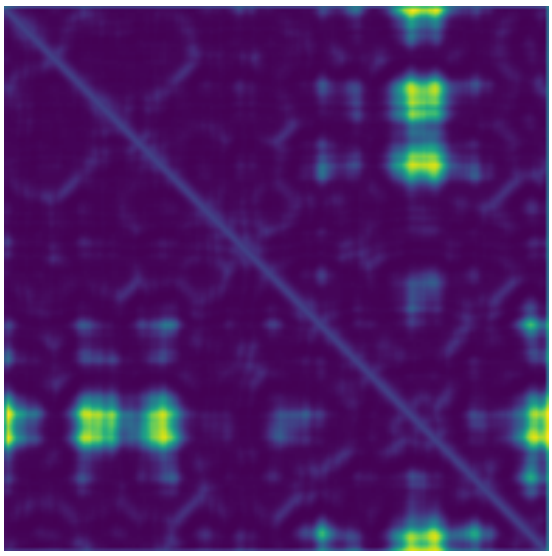
Figura 4.8. Curvas RMSF para o RBD da proteína S selvagem (em *vermelho*) e, com a mutação S:N501Y (em *azul*). Fonte: Obtida a partir do colaborador MSc. José Gutembergue Mendonça.



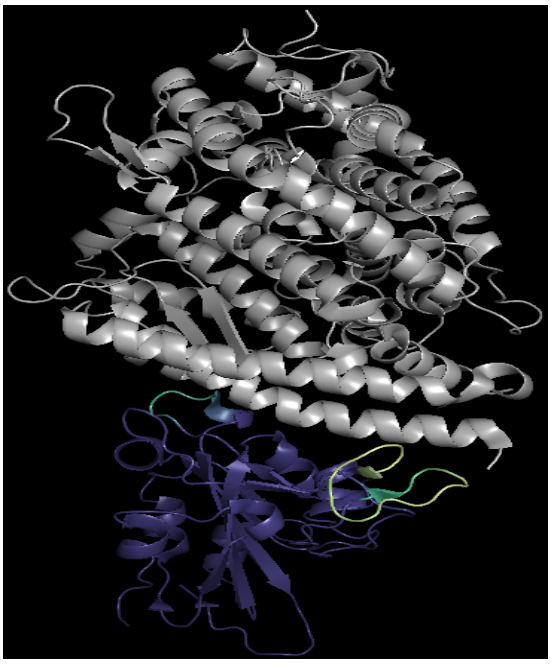
(a) Mapa de recursos referente ao *frame* 4500 da trajetória selvagem.



(b) Estrutura do Complexo RBD-hACE2 do SARS-CoV-2 (PDB ID 6M0J).



(c) Mapa de recursos referente ao *frame* 4500 da trajetória mutada.



(d) Estrutura do Complexo RBD-hACE2 do SARS-CoV-2 (PDB ID 6M0J).

Figura 4.9. O complexo RBD-hACE2 (à direita) tem como destaque os resíduos correspondentes a regiões no mapa de recursos 2D (à esquerda), onde ocorre a maximização da ativação, a partir da primeira camada convolucional. Na parte superior, tem-se o mapa de recursos para um *frame* referente à trajetória da *selvagem*, enquanto abaixo o mapa de recursos para um *frame* referente à trajetória da *mutada*. É possível observar para as estruturas, variações nas intensidades de cor em regiões onde encontram-se as mutações, como entre os resíduos 477 a 487 (parte frontal) e/ou, 498 a 502 (parte posterior). Algumas dessas variações podem estar associadas a mudanças conformacionais da estrutura. Fonte: Elaborada pelos autores.

Capítulo 5

Conclusões e perspectivas

O presente trabalho inicia na análise preliminar das redes convolucionais aplicadas a um problema simples ou seja, a identificação de mudanças conformacionais em larga escala na estrutura de proteínas, representadas por mapas de distâncias. Nesse caso, a proteína alvo foi a glicoproteína espicular S do SARS-CoV-2, sendo que a proteína poderia se encontrar em dois estados: trímeros afastados, ou pré-fusão, (aberta) e com os trímeros próximos (fechada). Uma vez que os resultados dessa análise foram bem sucedidos, o trabalho foi apresentado no Brazilian Symposium on Bioinformatics (BSB 2022), promovido pela Sociedade Brasileira de Computação (SBC), em Búzios-RJ. Contudo, ainda restava verificar se essas redes neurais também eram capazes de identificar conformações de menor escala em proteínas, a partir dos mapas de distâncias.

Assim, na seção 1.5, a questão de pesquisa instiga sobre a possibilidade do uso de simulações de dinâmica molecular, representadas como mapas de distâncias, à obtenção de modelos baseados em aprendizado profundo, capazes de identificar mudanças sutis nos padrões de movimento da estrutura. Essa abordagem possibilitaria, por exemplo, prever o impacto de mutações no RBD da proteína S do SARS-CoV-2, como o ganho de afinidade com o receptor humano. Assim, observa-se a relevância do tema, bem como a necessidade dos resultados serem congruêntes a trabalhos recentemente publicados.

Com base nos resultados do capítulo 4, é possível verificar as seguintes contribuições:

1. Predição de mutações com potencial de provocar ganho de afinidade. Isso poderia ser obtido a partir de dinâmicas curtas, por exemplo.

2. Correlação entre a precisão do modelo e o $\Delta\Delta G$ (considerando os sistemas da base de dados), sugerindo uma coerência nas predições realizadas pelo modelo;

3. Identificação das regiões mais significativas ao aprendizado e à decisão do modelo;

Em relação ao item I, é possível observar que a precisão do modelo está diretamente relacionada à *afinidade* de ligação com o receptor. Assim, os resultados obtidos para as mutações N501I e N501T (além das VOCs e VOIs conhecidas) acompanham estudos recentemente publicados [Harvey et al., 2021; Liu et al., 2021; Sanches et al., 2021; Verma & Subbarao, 2021; Khan et al., 2022; McCallum et al., 2022], indicando uma corretude das predições do modelo quanto à infectividade. O mesmo ocorre ao sistema composto pelas mutações L452R, T478K e N501Y, cuja precisão foi de 72,0%. Nesse caso, é possível observar duas mutações encontradas na variante B.1.617.2 (Delta, δ), além da mutação S:N501Y. Por fim, aos sistemas contendo mutações ditas neutras [Teng et al., 2020; Chen et al., 2021b], como a S:Q498K, ou ainda a S:Y508H, o modelo alcança uma precisão de apenas 39,0% e 10%, respectivamente.

A correlação calculada ($\rho = 0,77$) entre a *precisão* do modelo e o $\Delta\Delta G$ converge positivamente, indicando que o ganho de afinidade está relacionado com as mudanças de energia livre de ligação [Chen et al., 2021b; Hoffmann et al., 2020; Teng et al., 2020; Walls et al., 2020]. Muito embora os valores de $\Delta\Delta G$ tenham sido estimados às diferentes mutações no RBD da proteína S [Teng et al., 2020; Chen et al., 2020b; Verma & Subbarao, 2021; Chen et al., 2021a; Wang et al., 2022], esses resultados foram validados com base em dados experimentais [Chen et al., 2021a; Linsky et al., 2020]. Desse modo, também é possível correlacionar a precisão do modelo com resultados de abordagens independentes.

Quanto à visualização dos recursos aprendidos pelo modelo, observa-se que os mapas extraídos apresentam como destaque regiões abrangidas pelo RBM (resíduos 438-510). Trabalhos recentes apontam a ocorrência de picos de flutuação, relacionados a loops, na região que abrange os resíduos 475 ao 487 do RBM. Embora esses picos sejam comuns no RBD da selvagem, como às mutações associadas ao ganho de afinidade, as maiores flutuações podem ser verificadas às variantes [Alaofi & Shahid, 2021; Verma & Subbarao, 2021]. Além disso, o RBD com a mutação S:N501Y apresentou confórmeros no RBM que variaram significativamente na região de loop 498-502 (ligeiramente flexível), revelando movimentos de elevada amplitude em direção ao receptor humano [Alaofi & Shahid, 2021; Verma & Subbarao, 2021]. Essas constatações são refletidas nos filtros que apresentam valores distintos às duas classes do problema.

Nesse sentido, é possível concluir que as redes convolucionais mostram-se promissoras à identificação de mudanças conformacionais sutis em trajetórias de dinâmica molecular, representadas como mapas de distâncias. Os resultados alcançados estimulam o uso da abordagem à análise conformacional em diferentes estruturas, possibilitando a compreensão de propriedades conformacionais relevantes em sistemas moleculares complexos. Um exemplo, seria avaliar a capacidade desses modelos baseados em ConvNets, identificarem mutações que também proporcionem ao vírus a capacidade de evadir-se do

sistema imune.

Por fim, pretende-se avaliar essa abordagem para outras proteínas-alvo que interajam com outros ligantes, permitindo a exploração de funcionalidades diversas. Também pretende-se extrair *insights* a partir de dinâmicas curtas, sendo possível prever informações, antes obtidas apenas com dinâmicas longas. Isso possibilitaria o estudo de potenciais mutações, com ganho de afinidade, em um intervalo de tempo reduzido.

Referências Bibliográficas

Abadi, M. et al. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems.

Adhikari, B.; Hou, J. & Cheng, J. (2018). Dncon2: improved protein contact prediction using two-level deep convolutional neural networks. *Bioinformatics*, 34(9):1466--1472.

Akkız, H. (2022). The biological functions and clinical significance of SARS-CoV-2 variants of corcern. *Frontiers in Medicine*, 9. doi:10.3389/fmed.2022.849217.

Alaofi, A. L. & Shahid, M. (2021). Mutations of sars-cov-2 rbd may alter its molecular structure to improve its infection efficiency. *Biomolecules*, 11(9):1273. doi:10.3390/biom11091273.

Alberts, B. et al. (2002). *Molecular Biology of the Cell*. Garland Science, New York, NY, USA, 4ª edição.

Alder, B. J. & Wainwright, T. E. (1957). Phase transition for a hard sphere system. *The Journal of Chemical Physics*, 27(5):1208--1209.

Ali, M. S. et al. (2021). An enhanced technique of skin cancer classification using deep convolutional neural network with transfer learning models. *Machine Learning with Applications*, 5:100036. doi:10.1016/j.mlwa.2021.100036.

Alquraishi, M. (2019). Alphafold at casp13. *Bioinformatics*, 35:4862–4865. doi:10.1093/bioinformatics/btz422.

Anand, N. & Huang, P. (2018). Generative modeling for protein structures. Em Bengio, S. et al., editores, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Anishchenko, I. et al. (2021). De novo protein design by deep network hallucination. *Nature*, 600:547–552. doi:10.1038/s41586-021-04184-w.

Baek, M. et al. (2021). Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373:871–876. doi:10.1126/science.abj8754.

Bengio, Y.; Courville, A. & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828. doi:10.1109/TPAMI.2013.50.

Chen, C. et al. (2022). A CNN model for predicting binding affinity changes between SARS-CoV-2 spike RBD variants and ACE2 homologues. doi:10.1101/2022.03.22.485413.

Chen, J. et al. (2019a). Effect of mutations on binding of ligands to guanine riboswitch probed by free energy perturbation and molecular dynamics simulations. *Nucleic Acids Research*, 47(13):6618--6631. doi:10.1093/nar/gkz499.

Chen, J. et al. (2020a). Effects of disulfide bonds on binding of inhibitors to β-amyloid cleaving enzyme 1 decoded by multiple replica accelerated molecular dynamics simulations. *ACS Chemical Neuroscience*, 11(12):1811--1826. doi:10.1021/acschemneuro.0c00234.

Chen, J. et al. (2020b). Mutations strengthened SARS-CoV-2 infectivity. *Journal of Molecular Biology*, 432(19):5212--5226. 10.1016/j.jmb.2020.07.009.

Chen, J. et al. (2021a). Revealing the threat of emerging SARS-CoV-2 mutations to antibody therapies. *Journal of Molecular Biology*, 433(18):167155. doi:10.1016/j.jmb.2021.167155.

Chen, J.; Wang, R.; Gilby, N. B. & Wei, G.-W. (2021b). Omicron (b.1.1.529): Infectivity, vaccine breakthrough, and antibody resistance. https://arxiv.org/abs/2112.01318.

Chen, J. E.; Huang, C. C. & Ferrin, T. E. (2014). Rrdistmaps: a ucsf chimera tool for viewing and comparing protein distance maps. *Bioinformatics*, 31(9):1484–1486. doi:10.1093/bioinformatics/btu841.

Chen, S. et al. (2019b). To improve protein sequence profile prediction through image captioning on pairwise residue distance map. *Journal of Chemical Information and Modeling*, 60(1):391--399. doi:10.1021/acs.jcim.9b00438.

Chen, X. & Cheng, J. (2022). DISTEMA: distance map-based estimation of single protein model accuracy with attentive 2d convolutional neural network. *BMC Bioinformatics*, 23(S3). doi:10.1186/s12859-022-04683-1.

Chollet, F. (2017). *Deep Learning with Python*. Manning, 4ª edição.

Chollet, F. et al. (2015). Keras.

Dauber-Osguthorpe, P. & Osguthorpe, D. J. (1993). Partitioning the motion in molecular dynamics simulations into characteristic modes of motion. *Journal of Computational Chemistry*, 14(11):1259--1271. doi:10.1002/jcc.540141102.

Defresne, M.; Barbe, S. & Schiex, T. (2021). Protein design with deep learning. *International Journal of Molecular Sciences*, 22. doi:10.3390/ijms222111741.

Ding, W. & Gong, H. (2020). Predicting the real-valued inter-residue distances for proteins. *Advanced Science*, 7(19):2001314. doi:10.1002/advs.202001314.

Ding, W.; Nakai, K. & Gong, H. (2022). Protein design via deep learning. *Briefings in Bioinformatics*, 23(3). doi:10.1093/bib/bbac102.

Duda, R. O.; Hart, P. E. & Stork, D. G. (2001). *Pattern Classification*. Wiley, New York, NY, USA, 2ª edição.

Durrant, J. D. & McCammon, J. A. (2011). Molecular dynamics simulations and drug discovery. *BMC Biology*, 9(1). doi:10.1186/1741-7007-9-71.

Faria, N. R. et al. (2021). Genomics and epidemiology of the p.1 SARS-CoV-2 lineage in manaus, brazil. *Science*, 372(6544):815--821. doi:10.1126/science.abh2644.

Fukunaga, K. (1990). *Introduction to Statistical Pattern Recognition*. Academic Press Professional, Inc., San Diego, CA, USA, 2ª edição.

Gao, W.; Mahajan, S. P.; Sulam, J. & Gray, J. J. (2020). Deep learning in protein structural modeling and design. *Patterns*, 1. doi:10.1016/j.patter.2020.100142.

Gifani, P.; Shalbaf, A. & Vafaeezadeh, M. (2020). Automated detection of COVID-19 using ensemble of transfer learning with deep convolutional neural network based on CT scans. *International Journal of Computer Assisted Radiology and Surgery*, 16(1):115--123. doi:10.1007/s11548-020-02286-w.

Gobeil, S. M. et al. (2021). Effect of natural mutations of sars-cov-2 on spike structure, conformation, and antigenicity. *Science*, 373. doi:10.1126/science.abi6226.

Gonzalez, R. & Woods, R. (2009). *Processamento Digital De Imagens*. Pearson, Londres, Reino Unido.

Goodfellow, I.; Bengio, Y. & Courville, A. (2016). *Deep Learning*. MIT Press, Massachusetts, USA.

Gour, N. & Khanna, P. (2021). Multi-class multi-label ophthalmological disease detection using transfer learning based convolutional neural network. *Biomedical Signal Processing and Control*, 66:102329. doi:10.1016/j.bspc.2020.102329.

Guruprasad, L. (2021). Human SARS CoV-2 spike protein mutations. *Proteins: Structure, Function, and Bioinformatics*, 89(5):569--576.

Halliday, D.; Resnick, R. & Walker, J. (2013). *Fundamentals of Physics*. Wiley, Nova Jersey, EUA, 10ª edição.

Harvey, W. T. et al. (2021). Sars-cov-2 variants, spike mutations and immune escape. *Nature Reviews Microbiology*, 19:409–424. doi:10.1038/s41579-021-00573-0.

Hassanat, A. et al. (2019). Choosing mutation and crossover ratios for genetic algorithms—a review with a new dynamic approach. *Information*, 10(12):390. doi:10.3390/info10120390.

Haykin, S. (2001). *Redes Neurais: Princípios e Prática*. Bookman, 2ª edição.

He, K. et al. (2014). Spatial pyramid pooling in deep convolutional networks for visual recognition. Em *Computer Vision ECCV 2014*, pp. 346--361. Springer International Publishing. doi:10.1007/978-3-319-10578-9-23.

He, K. et al. (2016). Deep residual learning for image recognition. Em *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. doi:10.1109/cvpr.2016.90.

Hodcroft, E. B. (2021). CoVariants: SARS-CoV-2 Mutations and Variants of Interest. `https://covariants.org/`, último acesso em 22/07/2022.

Hoffmann, M. et al. (2020). Sars-cov-2 cell entry depends on ace2 and tmprss2 and is blocked by a clinically proven protease inhibitor. *Cell*, 181(2):271–280.e8. doi:10.1016/j.cell.2020.02.052.

Hollingsworth, S. A. & Dror, R. O. (2018). Molecular dynamics simulation for all. *Neuron*, 99(6):1129--1143. doi:10.1016/j.neuron.2018.08.011.

Huang, G. et al. (2016a). Densely connected convolutional networks. https://arxiv.org/abs/1608.06993.

Huang, P.-S.; Boyken, S. E. & Baker, D. (2016b). The coming of age of de novo protein design. *Nature*, 537(7620):320--327.

Huang, Y. et al. (2020). Structural and functional properties of SARS-CoV-2 spike protein: potential antivirus drug development for COVID-19. *Acta Pharmacologica Sinica*, 41(9):1141--1149.

Ioffe, S. & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. 10.48550/ARXIV.1502.03167.

Iyer, M. et al. (2020). Difference contact maps: From what to why in the analysis of the conformational flexibility of proteins. *PLOS ONE*, 15(3):e0226702. doi:10.1371/journal.pone.0226702.

Iyer, M. et al. (2022). What the protein data bank tells us about the evolutionary conservation of protein conformational diversity. *Protein Science*, 31(7). doi:10.1002/pro.4325.

Johns Hopkins University (JHU) (2022). Johns Hopkins Coronavirus Resource Center (CRC). `https://coronavirus.jhu.edu/`, último acesso em 20/02/2022.

Jones, D. T. & Kandathil, S. M. (2018). High precision in protein contact prediction using fully convolutional neural networks and minimal sequence features. *Bioinformatics*, 34(19):3308--3315. doi:10.1093/bioinformatics/bty341.

Jumper, J. et al. (2021). Highly accurate protein structure prediction with alphafold. *Nature*, 596:583–589. doi:10.1038/s41586-021-03819-2.

Karimi, M. et al. (2020). De novo protein design for novel folds using guided conditional wasserstein generative adversarial networks. *Journal of Chemical Information and Modeling*, 60(12):5667--5681. doi:10.1021/acs.jcim.0c00593.

Kearns, M. (1997). A Bound on the Error of Cross Validation Using the Approximation and Estimation Rates, with Consequences for the Training-Test Split. *Neural Computation*, 9(5):1143–1161. doi:10.1162/neco.1997.9.5.1143.

Kessel, A. & Ben-Tal, N. (2018). *Introduction to Proteins: Structure, Function, and Motion*. Chapman and Hall/CRC, Boca Raton, Florida, USA, 2ª edição.

Khan, A. et al. (2022). Computational modelling of potentially emerging sars-cov-2 spike protein rbds mutations with higher binding affinity towards ace2: A structural modelling study. *Computers in Biology and Medicine*, 141:105163. doi:10.1016/j.compbiomed.2021.105163.

Kingma, D. P. & Ba, J. (2014). Adam: A method for stochastic optimization. 10.48550/ARXIV.1412.6980.

Kloczkowski, A.; Jernigan, R. L.; Wu, Z.; Song, G.; Yang, L.; Kolinski, A. & Pokarowski, P. (2009). Distance matrix-based approach to protein structure prediction. *Journal of Structural and Functional Genomics*, 10:67–81. doi:10.1007/s10969-009-9062-2.

Krizhevsky, A.; Sutskever, I. & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84--90. doi:10.1145/3065386.

Kurzweil, R. (1990). *The Age of Intelligent Machines*. Kurzweil Foundation.

Laffeber, C.; de Koning, K.; Kanaar, R. & Lebbink, J. H. (2021). Experimental evidence for enhanced receptor binding by rapidly spreading SARS-CoV-2 variants. *Journal of Molecular Biology*, 433(15):167058. doi:10.1016/j.jmb.2021.167058.

Laiton-Donato, K. et al. (2021). Characterization of the emerging b.1.621 variant of interest of SARS-CoV-2. *Infection, Genetics and Evolution*, 95:105038. doi:10.1016/j.meegid.2021.105038.

Lan, J. et al. (2020). Structure of the sars-cov-2 spike receptor-binding domain bound to the ace2 receptor. *Nature*, 581:215–220. doi:10.1038/s41586-020-2180-5.

Larsson, G.; Maire, M. & Shakhnarovich, G. (2016). Fractalnet: Ultra-deep neural networks without residuals. https://arxiv.org/abs/1605.07648.

Leach, A. (2001). *Molecular modelling : principles and applications*. Prentice Hall, USA, New Jersey, 2ª edição.

LeCun, Y. et al. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541--551.

Li, X. et al. (2019). Understanding the disharmony between dropout and batch normalization by variance shift. Em *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. doi:10.1109/cvpr.2019.00279.

Li, Z.; Nguyen, S. P.; Xu, D. & Shang, Y. (2017). Protein loop modeling using deep generative adversarial network. Em *2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE. doi:10.1109/ictai.2017.00166.

Lima, L. H. F. D. et al. (2021). Conformational flexibility correlates with glucose tolerance for point mutations in beta-glucosidases – a computational study. *Journal of Biomolecular Structure and Dynamics*, 39(5):1621–1634. doi:10.1080/07391102.2020.1734484.

Linsky, T. W. et al. (2020). De novo design of potent and resilient hACE2 decoys to neutralize SARS-CoV-2. *Science*, 370(6521):1208--1214. doi:10.1126/science.abe0075.

Liu, H. et al. (2021). The basis of a more contagious 501y.v1 variant of SARS-CoV-2. *Cell Research*, 31(6):720--722. doi:10.1038/s41422-021-00496-8.

Luan, B.; Wang, H. & Huynh, T. (2021). Enhanced binding of the n501y-mutated SARS-CoV-2 spike protein to the human ACE2 receptor: insights from molecular dynamics simulations. *FEBS Letters*, 595(10):1454--1461. doi:10.1002/1873-3468.14076.

Lupala, C. S. et al. (2022). Mutations on RBD of SARS-CoV-2 omicron variant result in stronger binding to human ACE2 receptor. *Biochemical and Biophysical Research Communications*, 590:34--41. doi:10.1016/j.bbrc.2021.12.079.

McCallum, M. et al. (2022). Structural basis of sars-cov-2 omicron immune evasion and receptor engagement. *Science*, 375:894–898. doi:10.1126/science.abn8652.

McCulloch, W. S. & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4):115--133. doi:10.1007/bf02478259.

Min, S.; Lee, B. & Yoon, S. (2017). Deep learning in bioinformatics. *Briefings in bioinformatics*, 18:851–869. doi:10.1093/bib/bbw068.

Ministério da Saúde (2022). Coronavírus Brasil. `https://covid.saude.gov.br/`, último acesso em 17/06/2022.

Mishkin, D.; Sergievskiy, N. & Matas, J. (2017). Systematic evaluation of convolution neural network advances on the imagenet. *Computer Vision and Image Understanding*, 161:11--19.

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill, New York, USA, 2ª edição.

Molnar, C. (2022). Interpretable machine learning - a guide for making black box models explainable. https://christophm.github.io/interpretable-ml-book.

Mosteller, F. & Tukey, J. W. (1968). Data analysis, including statistics. *Handbook of social psychology*, 2:80--203.

Muik, A. et al. (2021). Neutralization of SARS-CoV-2 lineage b.1.1.7 pseudovirus by BNT162b2 vaccine–elicited human sera. *Science*, 371(6534):1152--1153. doi:10.1126/science.abg6105.

Nelson, D. L. & Cox, M. M. (2004). *Lehninger Principles of Biochemistry*. W. H. Freeman, New York, USA, 4ª edição.

OMS (2022). Tracking sars-cov-2 variants. `https://www.who.int/en/activities/tracking-SARS-CoV-2-variants/`.

Ou, J. et al. (2021). V367f mutation in SARS-CoV-2 spike RBD emerging during the early transmission phase enhances viral infectivity through increased human ACE2 receptor binding affinity. *Journal of Virology*, 95(16). doi:10.1128/jvi.00617-21.

Pearson, K. (1896). VII. mathematical contributions to the theory of evolution.—III. regression, heredity, and panmixia. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 187:253--318. doi:10.1098/rsta.1896.0007.

Provost, F. J.; Fawcett, T.; Kohavi, R. et al. (1998). The case against accuracy estimation for comparing induction algorithms. Em *ICML*, volume 98, pp. 445--453.

Roe, D. R. & Cheatham, T. E. (2013). PTRAJ and CPPTRAJ: Software for processing and analysis of molecular dynamics trajectory data. *Journal of Chemical Theory and Computation*, 9(7):3084--3095. doi:10.1021/ct400341p.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386--408. doi:10.1037/h0042519.

Roy, R. et al. (2020). Investigating the mechanism of recognition and structural dynamics of nucleoprotein-RNA complex from peste des petits ruminants virus via gaussian accelerated molecular dynamics simulations. *Journal of Biomolecular Structure and Dynamics*, 40(5):2302--2315. doi:10.1080/07391102.2020.1838327.

Russakovsky, O. et al. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252. doi:10.1007/s11263-015-0816-y.

Russell, S. & Norvig, P. (2010). *Artificial Intelligence: A Modern Approach*. Prentice Hall, 3 edição.

S, U. K. et al. (2020). Mutational landscape of k-ras substitutions at 12th position-a systematic molecular dynamics approach. *Journal of Biomolecular Structure and Dynamics*, 40(4):1571--1585. doi:10.1080/07391102.2020.1830177.

Sanches, P. R. et al. (2021). Recent advances in sars-cov-2 spike protein and rbd mutations comparison between new variants alpha (b.1.1.7, united kingdom), beta (b.1.351, south africa), gamma (p.1, brazil) and delta (b.1.617.2, india). *Journal of Virus Eradication*, 7. doi:10.1016/j.jve.2021.100054.

Saxena, S. K. et al. (2022). Characterization of the novel SARS-CoV-2 omicron (b.1.1.529) variant of concern and its global perspective. *Journal of Medical Virology*, 94(4):1738--1744. doi:10.1002/jmv.27524.

Schilling, F. (2016). The effect of batch normalization on deep convolutional neural networks.

Sehnal, D. et al. (2021). Mol* viewer: modern web app for 3d visualization and analysis of large biomolecular structures. *Nucleic Acids Research*, 49(W1):431--437. doi:10.1093/nar/gkab314.

Senior, A. W. et al. (2020). Improved protein structure prediction using potentials from deep learning. *Nature*, 577:706–710. doi:10.1038/s41586-019-1923-7.

Shapiro, S. C. (1992). *Encyclopedia of Artificial Intelligence*. Number v. 1 in A Wiley-Interscience publication. Wiley.

Simonyan, K. & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. 10.48550/ARXIV.1409.1556.

Srivastava, N. et al. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929--1958.

Strodel, B. (2021). Energy landscapes of protein aggregation and conformation switching in intrinsically disordered proteins. *Journal of Molecular Biology*, 433(20):167182. doi:10.1016/j.jmb.2021.167182.

Stryer, L.; Berg, J. & Tymoczko, J. (2002). *Biochemistry*. W.H. Freeman, New York, NY, USA, 5ª edição.

Szegedy, C. et al. (2015). Going deeper with convolutions. Em *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. doi:10.1109/cvpr.2015.7298594.

Tao, K. et al. (2021). The biological and clinical significance of emerging SARS-CoV-2 variants. *Nature Reviews Genetics*, 22(12):757--773. doi:10.1038/s41576-021-00408-x.

Tegally, H. et al. (2021). Detection of a SARS-CoV-2 variant of concern in south africa. *Nature*, 592(7854):438--443. doi:10.1038/s41586-021-03402-9.

Teng, S. et al. (2020). Systemic effects of missense mutations on sars-cov-2 spike glycoprotein stability and receptor-binding affinity. *Briefings in Bioinformatics*, 22(2):1239–1253. doi:10.1093/bib/bbaa233.

Teruel, N.; Mailhot, O. & Najmanovich, R. J. (2021). Modelling conformational state dynamics and its role on infection for SARS-CoV-2 spike protein variants. *PLOS Computational Biology*, 17(8):e1009286. doi:10.1371/journal.pcbi.1009286.

Theodoridis, S. & Koutroumbas, K. (2009). *Pattern Recognition*. Academic Press, Burlington, MA, USA.

Tian, F. et al. (2021). N501y mutation of spike protein in SARS-CoV-2 strengthens its binding to receptor ACE2. *eLife*, 10. doi:10.7554/elife.69091.

Tompson, J. et al. (2014). Efficient object localization using convolutional networks. doi:10.48550/ARXIV.1411.4280.

Torrisi, M.; Pollastri, G. & Le, Q. (2020). Deep learning methods in protein structure prediction. *Computational and Structural Biotechnology Journal*, 18:1301–1310. doi:10.1016/j.csbj.2019.12.011.

Vapnik, V. (1991). Principles of risk minimization for learning theory. Em Moody, J.; Hanson, S. & Lippmann, R., editores, *Advances in Neural Information Processing Systems*, volume 4, pp. 831–838. Morgan-Kaufmann.

Verma, J. & Subbarao, N. (2021). Insilico study on the effect of sars-cov-2 rbd hotspot mutants' interaction with ace2 to understand the binding affinity and stability. *Virology*, 561:107–116. doi:10.1016/j.virol.2021.06.009.

Vijayakumar, S. & Das, P. (2019). Structural, molecular motions, and free-energy landscape of leishmania sterol-14-alpha-demethylase wild type and drug resistant mutant: a comparative molecular dynamics study. *Journal of Biomolecular Structure and Dynamics*, 37(6):1477--1493. doi:10.1080/07391102.2018.1461135.

Vora, J. et al. (2019). Pharmacophore modeling, molecular docking and molecular dynamics simulation for screening and identifying anti-dengue phytocompounds. *Journal of Biomolecular Structure and Dynamics*, pp. 1--15. doi:10.1080/07391102.2019.1615002.

Walls, A. C. et al. (2020). Structure, function, and antigenicity of the SARS-CoV-2 spike glycoprotein. *Cell*, 181(2):281--292.e6. doi:10.1016/j.cell.2020.02.058.

Walsh, I. et al. (2009). Ab initio and template-based prediction of multi-class distance maps by two-dimensional recursive neural networks. *BMC Structural Biology*, 9(1). doi:10.1186/1472-6807-9-5.

Wang, J. et al. (2018). Computational protein design with deep learning neural networks. *Scientific Reports*, 8(1). doi:10.1038/s41598-018-24760-x.

Wang, P. et al. (2021). Antibody resistance of SARS-CoV-2 variants b.1.351 and b.1.1.7. *Nature*, 593(7857):130--135. doi:10.1038/s41586-021-03398-2.

Wang, R. et al. (2022). Emerging vaccine-breakthrough SARS-CoV-2 variants. *ACS Infectious Diseases*, 8(3):546--556. doi:10.1021/acsinfecdis.1c00557.

Wang, S. et al. (2017). Accurate de novo prediction of protein contact map by ultra-deep learning model. *PLoS Computational Biology*, 13. doi:10.1371/journal.pcbi.1005324.

Webb, A. & Copsey, K. (2011). *Statistical Pattern Recognition*. Wiley, New York, USA.

Wibmer, C. K. et al. (2021). SARS-CoV-2 501y.v2 escapes neutralization by south african COVID-19 donor plasma. *Nature Medicine*, 27(4):622--625. doi:10.1038/s41591-021-01285-x.

Wingler, L. M. et al. (2019). Angiotensin analogs with divergent bias stabilize distinct receptor conformations. *Cell*, 176(3):468--478.e11. doi:10.1016/j.cell.2018.12.005.

Wu, F. et al. (2020). A new coronavirus associated with human respiratory disease in china. *Nature*, 579(7798):265--269.

Xu, J. (2019). Distance-based protein folding powered by deep learning. *Proceedings of the National Academy of Sciences*, 116(34):16856--16865. doi:10.1073/pnas.1821309116.

Yang, J. et al. (2020). Improved protein structure prediction using predicted interresidue orientations. *Proceedings of the National Academy of Sciences*, 117(3):1496--1503. doi:10.1073/pnas.1914677117.

Zhang, J. et al. (2021). Structure of sars-cov-2 spike protein. *Current Opinion in Virology*, 50:173–182. doi:10.1016/j.coviro.2021.08.010.

Zheng, S. et al. (2020). Predicting drug–protein interaction using quasi-visual question answering system. *Nature Machine Intelligence*, 2(2):134--140. doi:10.1038/s42256-020-0152-y.