

1 INTRODUÇÃO

1.1 Contextualização

O surgimento do vírus SARS-CoV-2 (Coronavirus 2 da síndrome respiratória aguda grave), no final de 2019 (Zhu *et al.*, 2020) causador da epidemia de COVID-19 (Coronavirus Disease 2019), mergulhou o mundo em um estado de incerteza sem precedentes. Esse novo coronavírus, rapidamente se disseminou por todo o planeta, expondo a carência de conhecimento sobre sua origem, mecanismos de ação e epidemiologia (Organization, 2020).

A comunidade científica se viu diante de um patógeno inédito, com características e comportamentos ainda a serem desvendados (Petrosillo *et al.*, 2020). A escassez de informações precisas dificultava o desenvolvimento de estratégias eficazes de controle da pandemia, a criação de testes diagnósticos confiáveis e o direcionamento de esforços para o desenvolvimento de vacinas e terapias.

Diante da urgência em compreender o SARS-CoV-2 e seus mecanismos de ação, a análise genômica se tornou uma ferramenta crucial para desvendar os segredos do vírus. Através da sequenciamento do genoma viral, pesquisadores puderam identificar mutações, rastrear a origem e a disseminação do vírus, e entender os fatores que contribuem para sua virulência e transmissibilidade (Walls *et al.*, 2020); (Kucharski *et al.*, 2020). Com isso gerou-se um grande volume de dados, somente em três anos de existência, segundo o GISAID(Global Initiative on Sharing All Influenza Data) a comunidade científica gerou quase 17 milhões de genomas de SARS-CoV-2. Para poder realmente entender o volume de dados gerado pelo sequenciamento genômico do SARS-CoV-2 durante a epidemia de COVID-19, é necessário consultar quantos genomas tem sido gerados de outros patógenos. Segundo o GENBANK (banco de dados de sequências de nucleotídeos publicamente disponíveis), que pertence ao National Institute of Health (NIH) dos Estados Unidos, do Zika vírus, que é sequenciado desde 1947, tem pouco mais de 2500 genomas, enquanto do vírus da Dengue, que é sequenciado desde os anos 60, aproximadamente 17 mil genomas.

Neste cenário, marcado pelo crescente volume dos dados genômicos gerados com modernas e eficientes técnicas de sequenciamento (NGS - *Next – Generation Sequencing*) (Metzker; L., 2010), a bioinformática se apresenta como uma ferramenta crucial para a análise e interpretação desses dados, impulsionando avanços em diversas áreas da ciência, especialmente no combate a pandemias (Rana, 2023). Esse método científico, que combina métodos computacionais e técnicas estatísticas, permite extrair informações valiosas de sequências genômicas e proteicas, abrindo caminho para uma compreensão mais profunda da evolução dos vírus e o desenvolvimento de estratégias eficazes contra doenças como a COVID-19 (causada pelo vírus SARS-CoV-2), a Febre da Dengue (causado pelo arbovírus

DENV), a AIDS (Síndrome da Imunodeficiência Humana causada pelo vírus HIV), a Doença do Ebola (Doença de altíssima taxa de mortalidade causada pelo vírus ZEV) e a Febre Amarela (causada pelo arbovírus da Febre Amarela YFV), entre outras (Wilkinson *et al.*, 2023; Lespinet; Médigue; Lazarević, 2021).

No contexto da pandemia de COVID-19, a bioinformática foi crucial na análise das sequências do vírus SARS-CoV-2, permitindo o rastreamento da origem e evolução da doença, a identificação de novas variantes e o desenvolvimento de vacinas e medicamentos mais eficazes (Padmanabhan; Heinen; Chockalingam, 2022; Olson; Schaffer; Shores, 2023). Através da análise de grandes conjuntos de dados genômicos, foi possível mapear a disseminação do vírus em diferentes regiões do mundo, identificar mutações que afetam sua transmissibilidade e virulência, e desenvolver modelos preditivos para o futuro da pandemia.

Fruto da mobilização da comunidade internacional de bioinformáticos, diversas ferramentas foram desenvolvidas em muito curto tempo para analisar as novas cepas que eram sequenciadas cada dia. Dentre dessas ferramentas se destaca o Genome Detective, desenvolvido por um grupo internacional de pesquisadores liderado na área de programação, por um pesquisador brasileiro, egresso e docente atualmente do nosso Colegiado de Sistemas de Informação da UNEB, o Prof. Dr. Vagner Fonseca.

O impacto causado pelo uso dessas ferramentas bioinformáticas no combate ao COVID mostrou a necessidade/conveniência de utilizar mais intensivamente o sequenciamento dos patógenos virais conhecidos, combinado à análise bioinformática das sequências genômicas virais armazenadas, utilizando as novas técnicas e recursos (bases de dados, principalmente) desenvolvidas durante a epidemia de COVID-19, assim como a necessidade da melhoria continuada das técnicas bioinformáticas para o estudo da evolução viral.

1.2 Justificativa do Trabalho

O crescente volume de dados genômicos demanda o aprimoramento de ferramentas bioinformáticas mais eficientes e escaláveis (Meena; Kumar; Tomar, 2021). Contudo, a escalabilidade para considerar um grande número de sequências, na grande maioria das ferramentas atuais estudadas de uso público¹, estão limitadas por dois fatores principais:

1. Não utilizam Inteligência Artificial (IA) (OECD, 2024) para a coleta de padrões que possam ser armazenados e utilizados como referência para análise de novas cepas virais,
2. Derivado da limitação anterior, para analisar uma nova cepa, é preciso refazer o processo de clusterização, que é computacionalmente complexo (demorado e consumidor de memória), com todas as sequências de referência acrescidas da nova sequência. Isto faz

¹ Esta pesquisa não consultou ferramenta que não são de uso público

com que seja necessário estabelecer um balanceamento entre o número de sequências de referência consideradas e o tempo de análise de uma sequência nova. Como não é possível, salvo raras exceções (Buchfink *et al.*, 2023), considerar todas as sequências distintas disponíveis como referência, o conjunto de sequências de referência, não representa "toda" a diversidade genética já sequenciada, o que de fato impossibilita a identificação totalmente confiável de novas cepas, ou seja, poder dizer que uma cepa recém sequenciada é radicalmente diferente de todas as outras sequenciadas até esse momento. Em outras palavras, que a nova cepa representa uma nova variante do vírus.

Neste cenário se justifica explorar abordagens ao problema de genotipagem viral que utilizem a IA, e em particular o Aprendizado de Máquina (AM). (Goodfellow; Bengio; Courville, 2016). Neste sentido o presente projeto se alinha nos esforços do Grupo de Pesquisas em Bioinformática e Biologia Computacional (G2BC) da UNEB para desenvolver uma ferramenta WEB para genotipagem viral baseada em AM.

1.3 Objetivos do Trabalho

A missão é continuar o aprimoramento dos algoritmos de processamento e o desenvolvimento da ferramenta AGUA (Ad-hoc Genotyping Tool with Unsupervised Algorithm) para a genotipagem rápida de sequências virais, baseada em AM. Alinhados ao problema de pesquisa abordado, foram elencados os seguintes objetivos:

1.3.1 *Objetivo Geral*

Contribuir com o desenvolvimento da ferramenta AGUA para a genotipagem viral baseada em Aprendizado de Máquina.

1.3.2 *Objetivos Específicos*

1. Assimilar o funcionamento detalhado do backend existente e dos algoritmos utilizados para o pré-processamento de sequências.
2. Projetar, desenvolver e validar pipeline do backend para a classificação de novas sequências utilizando os modelos pré-treinados.
3. Projetar e desenvolver camadas frontend e intermediária para disponibilizar o uso (com fins de teste) da ferramenta AGUA na WEB (fase piloto).
4. Melhorar o desempenho do algoritmo de otimização dos parâmetros do método de clusterização CLOPE, integrante do backend.

5. Aferir o desempenho da nova versão do algoritmo utilizando sequências de SARS-CoV-2 e do vírus da Dengue, comparando com ferramenta clássica de filogenia do estado da arte.
6. Validação dos agrupamentos gerados com AGUA utilizando conjunto de sequências genotipadas do vírus da Dengue, comparando com os resultados obtidos anteriormente para SARS-CoV-2.
7. Avaliação integral da ferramenta e proposta de melhorias para a próxima fase de desenvolvimento.

1.4 Contribuições

Este trabalho oferece várias contribuições significativas ao campo da bioinformática e à luta contra as pandemias virais, como a COVID-19. A seguir, destacam-se as principais contribuições deste projeto:

1. Desenvolvimento de Ferramenta Inovadora:

A criação da ferramenta AGUA, que utiliza algoritmos de Aprendizado de Máquina (AM) para genotipagem viral, representa um avanço significativo em relação às ferramentas tradicionais. A AGUA facilita a análise rápida e precisa de sequências virais, o que é crucial para o monitoramento e controle de pandemias.

2. Integração de AM:

A aplicação de técnicas de Aprendizado de Máquina no processamento de dados genômicos oferece uma abordagem mais eficiente para a análise de grandes volumes de dados. A IA permite a identificação de padrões e a utilização de sequências de referência de maneira mais inteligente, reduzindo a necessidade de refazer processos complexos de clusterização.

3. Melhoria de Algoritmos de Clusterização:

A otimização do algoritmo de clusterização CLOPE melhora significativamente o desempenho do backend. Esta otimização permite um processamento mais rápido e eficiente das sequências, possibilitando análises em tempo real e com menor consumo de recursos computacionais.

4. Avaliação Comparativa com Ferramentas Clássicas:

A comparação dos resultados gerados pela AGUA com ferramentas clássicas de filogenia e genotipagem estabelece um benchmark de desempenho. Esta avaliação ajuda a validar a eficácia da ferramenta e a identificar áreas de melhoria contínua.

5. Validação com Dados Reais:

A validação da ferramenta AGUA utilizando conjuntos de dados reais, como as sequências de SARS-CoV-2 e do vírus da Dengue, assegura a aplicabilidade e a robustez da ferramenta em contextos práticos. Esta validação demonstra a capacidade da AGUA de lidar com diferentes tipos de vírus e suas variantes.

6. Facilitação do Acesso e Uso da Ferramenta:

O desenvolvimento das camadas frontend e intermediária disponibiliza a AGUA na web, permitindo que pesquisadores e profissionais de saúde possam utilizar a ferramenta de forma acessível e intuitiva. Esta fase piloto é fundamental para testar e refinar a usabilidade da ferramenta antes de sua implementação em larga escala.

7. Proposta de Melhorias Futuras:

A avaliação integral da ferramenta e a proposta de melhorias contínuas garantem que a AGUA evolua de acordo com as necessidades emergentes da pesquisa viral e das técnicas bioinformáticas. Esta abordagem proativa assegura a relevância e a eficácia da ferramenta em longo prazo.

Em resumo, este trabalho não só contribui com uma ferramenta poderosa para a genotipagem viral, mas também estabelece um framework para a integração de IA, em particular de AM, em bioinformática, promovendo avanços significativos na análise de dados genômicos e no combate a pandemias.

1.5 Estrutura da Monografia

JOILSON -> fazer no final

2 FUNDAMENTAÇÃO TEÓRICA

Na escrita desta seção assumimos que o leitor possui o conhecimento básico de Biologia molecular, em particular sobre os seguinte tópicos¹:

- Estrutura e funcionamento celular em eucariotos, em particular, a composição e funcionamento do núcleo e do citoplasma celular.
- Mecanismo de produção de proteínas na célula a partir do DNA.
 - Genes e processo de transcrição dos genes - de sequências de nucleotídeos no DNA para sequências de códons no mRNA.
 - Processo de tradução de genes - de sequências de códons no mRNA para sequências de aminoácidos (proteínas).
- Estrutura e funcionamento dos vírus de RNA, em particular, o papel das proteínas de ligação do vírus à célula hospedeira.

Devido a que a rápida identificação e genotipagem de cepas virais são essenciais para a resposta eficaz a surtos de doenças infecciosas e para o monitoramento contínuo da evolução viral, este projeto visa desenvolver um sistema web baseado em aprendizado de máquina para a genotipagem rápida de cepas virais. Baseado nisto, este capítulo abordará os seguintes assuntos:

- Os vírus em geral, na seção 2.1,
- O SARS-CoV-2, na seção 2.2,
- O vírus da Dengue, na seção 2.3,
- A genotipagem viral em Laboratório, na seção 2.4,
- A genotipagem viral com Bioinformática, na seção ??,
- Técnicas de Aprendizado de Máquina, na seção 2.5,
- O Aprendizado de Máquina Aplicado na Genômica, na seção 2.6,
- O Aprendizado de Máquina Aplicado para Genotipagem, na seção 2.7,
- Estrutura de sistema WEB para Bioinformática, na seção 2.8.

¹ Caso o leitor não estiver familiarizado com esses assuntos, sugerimos que estude os assuntos listados a seguir, antes de retornar à análise deste trabalho

2.1 Os vírus em geral

Os vírus, que são agentes infecciosos submicroscópicos, apresentam uma estrutura fundamental composta por um material genético (DNA ou RNA) encapsulado por uma capa proteica ou envelope lipídico. Contudo, eles somente podem se replicar dentro das células de um hospedeiro, já que não possuem estruturas capazes de transcrever e traduzir essa informação genética em proteínas funcionais. Para isso, eles precisam do sistema de transcrição/tradução da célula hospedeira, ou seja, do mecanismo de síntese de proteínas e enzimas do tipo de célula que é infectado pelo vírus.

A diversidade genética viral é vasta, com milhares de espécies conhecidas infectando uma ampla gama de organismos, incluindo humanos, animais e plantas (Koonin; Dolja; Krupovic, 2019). A compreensão da diversidade genética dos vírus é crucial para a epidemiologia e para o desenvolvimento de tratamentos e vacinas. A vasta gama de genomas virais, resultante de mutações e recombinações, permite que os vírus se adaptem rapidamente a novos hospedeiros e ambientes, tornando o controle de infecções virais um desafio contínuo (Domingo; Perales, 2019). A compreensão detalhada das mutações é essencial para o desenvolvimento de vacinas e terapias eficazes. Através da vigilância genômica contínua, os cientistas podem identificar rapidamente novas variantes que possam escapar da imunidade conferida pelas vacinas atuais e ajustar as estratégias de saúde pública conforme necessário (Grubaugh *et al.*, 2020).

A estrutura de um vírus é essencial para sua função e patogenicidade. O capsídeo proteico que envolve o material genético serve como uma proteção contra degradação e auxilia no reconhecimento e na infecção das células hospedeiras. Alguns vírus possuem um envelope lipídico derivado da membrana celular do hospedeiro, o que pode ajudar na evasão do sistema imunológico do hospedeiro (Howley; Knipe; Enquist, 2021).

Além disso, os vírus possuem mecanismos sofisticados para entrar nas células hospedeiras. Eles se ligam a receptores específicos na superfície da célula, o que desencadeia a entrada do vírus ou do seu material genético na célula. Uma vez dentro, o vírus usa a maquinaria celular para replicar seu genoma e produzir novas partículas virais, que são então liberadas para infectar novas células (Wagner *et al.*, 2021).

Nas próximas seções descrevemos brevemente os dois vírus que já foram usados para testar a ferramenta AGUA: o SARS-CoV-2, estudado no ano passado (Menezes, 2023), e o vírus da Dengue, objeto de estudo principal neste trabalho.

2.2 SARS-CoV-2

O SARS-CoV-2, o vírus causador da COVID-19, surgiu no final de 2019 e rapidamente se espalhou pelo mundo, causando uma pandemia de proporções históricas.

2.2.1 Estatísticas Relevantes

- Casos Confirmados: Mais de 600 milhões de casos confirmados de COVID-19 até o momento, segundo a Universidade Johns Hopkins.
- Mortes: Mais de 6 milhões de mortes por COVID-19 até o momento, segundo a Universidade Johns Hopkins.
- Distribuição Global: Afeta quase todos os países do mundo, com maior impacto em regiões com sistemas de saúde precários e alta densidade populacional.
- Impacto Econômico: Devastador, com perdas trilionárias em diversos setores da economia global.

2.2.2 Consequências da COVID-19

- Doença Grave e Morte: A COVID-19 pode causar doença grave, com sintomas como pneumonia, insuficiência respiratória e morte, principalmente em grupos de risco como idosos e pessoas com comorbidades.
- Sobrecarga nos Sistemas de Saúde: O aumento exponencial de casos sobrecarregou os sistemas de saúde, levando à escassez de leitos, respiradores e outros recursos médicos.
- Impacto Social: Isolamento social, fechamento de escolas e empresas, restrições de viagens e medidas de distanciamento afetaram a vida das pessoas em todo o mundo.
- Desafios na Prevenção e Controle: A alta transmissibilidade do vírus, a mutação constante e a existência de pessoas assintomáticas dificultam o controle da pandemia.

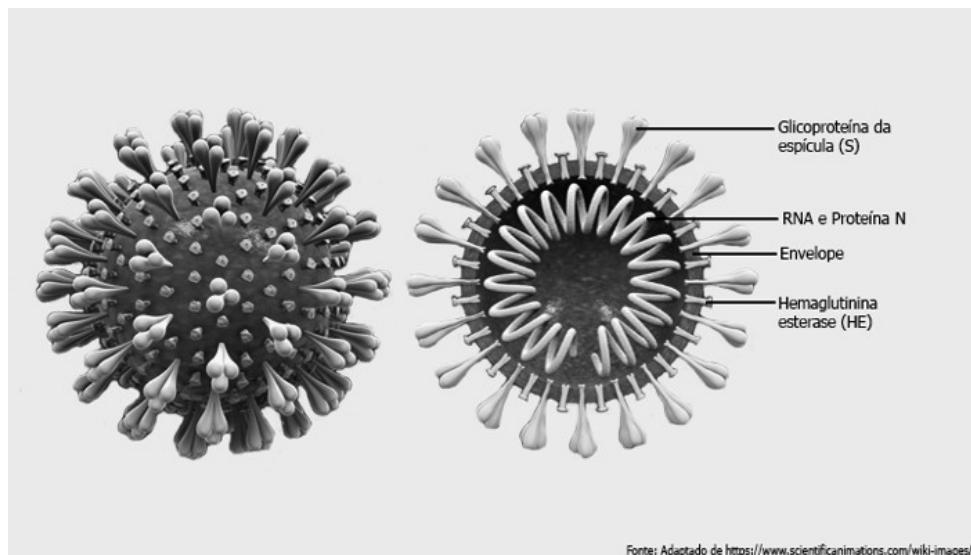
2.2.3 Estrutura Viral

A estrutura do SARS-CoV-2 é complexa e desempenha um papel crucial em sua capacidade de infectar células humanas. O vírus possui um genoma de RNA de fita simples, encapsulado em um nucleocapsídeo proteico. Esse nucleocapsídeo é rodeado por um envelope lipídico que contém três proteínas principais: a proteína Spike (S), a proteína de membrana (M) e a proteína do envelope (E). A proteína Spike é particularmente importante porque facilita a entrada do vírus nas células hospedeiras ao se ligar ao receptor ACE2 (enzima conversora de angiotensina 2) na superfície das células humanas.

Além disso, o SARS-CoV-2 é conhecido por sua capacidade de sofrer mutações, especialmente na região da proteína Spike. Essas mutações podem aumentar a afinidade do vírus pelo receptor ACE2, tornando-o mais transmissível. Por exemplo, a mutação

D614G na proteína Spike foi uma das primeiras a ser identificada como potencialmente aumentadora da transmissibilidade do vírus (Yurkovetskiy *et al.*, 2020).

Figura 1 – Estrutura do SARS-CoV-2



Fonte: (Lima; Rezende, 2020)

Na figura 1 mostramos a estrutura do SARS-CoV-2. O vírus possui um capsídeo esférico que contem o RNA. O capsídeo é formado por uma dupla membrana lipídea na qual se fixa a glicoproteína Spike que forma estruturas em forma de cogumelo. Em particular, como comentado acima, as mutações do gene Spike estão relacionadas com o escape do sistema imunológico, pelo que é o gene mais monitorado no estudo da evolução do vírus SARS-CoV-2 (Naqvi *et al.*, 2020).

2.3 Vírus da Dengue

A Dengue, causada por um vírus flavivírus, é uma doença viral transmitida por mosquitos do gênero Aedes, principalmente Aedes aegypti. O vírus da dengue, representa um sério problema de saúde pública, afetando milhões de pessoas em todo o mundo (Saúde, 2023).

2.3.1 Estatísticas Relevantes:

- Estimativa Anual: Entre 390 e 500 milhões de infecções por dengue a cada ano, segundo a Organização Pan-Americana da Saúde (OPAS).
- Distribuição Geográfica: Mais de 125 países em regiões tropicais e subtropicais estão em risco, com maior prevalência na Ásia, América Latina e Caribe.
- Mortalidade: Aproximadamente 20.000 mortes anuais por dengue grave, principalmente entre crianças e adolescentes.

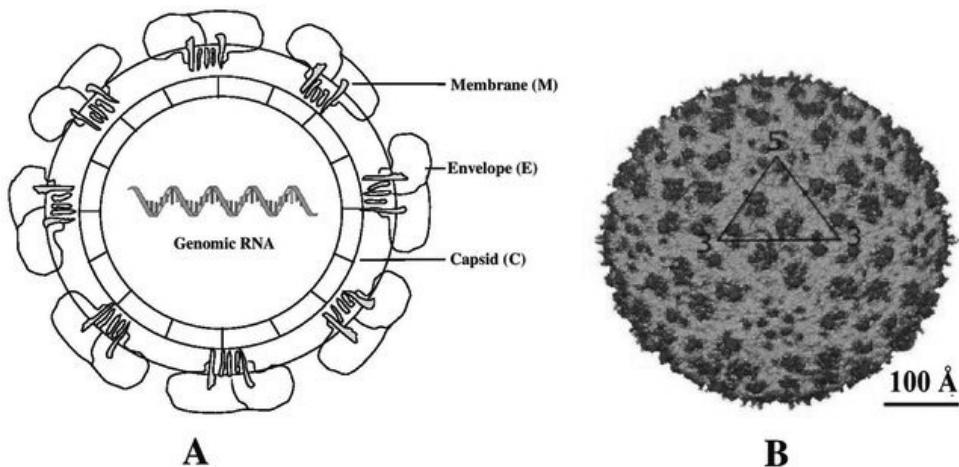
- Impacto Econômico: Significativo, com custos diretos e indiretos relacionados à hospitalização, perda de produtividade e impacto no turismo.

2.3.2 Consequências do Dengue:

- Doença Debilitante: A dengue pode causar sintomas graves, como febre alta, dor muscular e articular, erupção cutânea e, em casos mais graves, hemorragia e choque.
- Sobrecarga nos Sistemas de Saúde: O aumento de casos de dengue sobrecarrega os sistemas de saúde, especialmente em países com recursos limitados.
- Impacto Social: A dengue afeta a qualidade de vida das pessoas, causando absenteísmo escolar e profissional, além de restrições nas atividades cotidianas.
- Desafios na Prevenção e Controle: A complexa ecologia do mosquito Aedes aegypti e a adaptação do vírus a diferentes ambientes dificultam o controle da doença.

2.3.3 Estrutura Viral

Figura 2 – Estrutura do vírus da Dengue



Fonte: (Roy; Bhattacharjee, 2021)

Na figura 2 mostramos a estrutura do vírus da Dengue. Um capsídeo esférico protege o RNA genômico. No capsídeo as proteínas Envelope e M se juntam para formar estruturas que se fixam na membrana celular das células alvo, e são responsáveis pela resposta imune do organismo. Em particular, mutações do gene Envelope estão relacionadas com o escape do sistema imunológico, pelo que é o gene mais monitorado no estudo da evolução do vírus da Dengue (Gibson *et al.*, 2015). Por esta razão a genotipagem do gene Envelope do vírus da Dengue é crucial para monitorar a circulação de diferentes serotipos e para o desenvolvimento de vacinas e medicamentos mais eficazes (Saúde, 2023).

2.4 Genotipagem Viral em Laboratório Molhado (Sem Sequenciamento)

A genotipagem viral é crucial para o estudo e manejo de infecções virais, permitindo a identificação de variantes genéticas, a personalização do tratamento e o monitoramento da evolução viral. Diversas técnicas de genotipagem sem sequenciamento estão disponíveis, utilizando dados clínicos e resultados de exames laboratoriais para caracterizar os vírus. Abordaremos aqui as principais abordagens, destacando suas variantes, características e aplicações.

2.4.1 PCR com Conjuntos de Primers Selecionados

2.4.1.1 Princípio

A PCR com conjuntos de primers selecionados é uma técnica de genotipagem viral que utiliza primers direcionados a polimorfismos de nucleotídeos únicos (SNPs) ou regiões variáveis do genoma viral para amplificar regiões específicas. A análise dos produtos de PCR permite a identificação das variantes genéticas presentes no vírus.

2.4.1.2 Variantes

Existem duas variantes principais da PCR com conjuntos de primers selecionados:

- PCR em Tempo Real: Essa variante utiliza fluoróforos específicos para quantificar a amplificação em tempo real, permitindo a determinação da carga viral e o monitoramento da resposta ao tratamento antiviral.
- PCR Multipla (Multiplex PCR): Essa variante permite a amplificação simultânea de múltiplos alvos genômicos em um único ensaio, aumentando a eficiência da genotipagem.

2.4.1.3 Exemplos de Marcas Comerciais

- TaqMan® SARS-CoV-2 Genotyping Assay (Thermo Fisher Scientific): Genotipagem das variantes E, L452R e N501Y do SARS-CoV-2.
- Abbott RealTime HCV Genotyping Assay: Genotipagem dos genótipos 1-6 do vírus da hepatite C.

2.4.1.4 Características

A PCR com conjuntos de primers selecionados apresenta as seguintes características:

- Alta especificidade e sensibilidade.
- Rápida e relativamente barata.
- Útil para genotipagem de um número limitado de alvos genômicos.

2.4.1.5 Aplicações

A PCR com conjuntos de primers selecionados tem diversas aplicações, incluindo:

- Identificação de variantes virais associadas à resistência a medicamentos.
- Monitoramento da resposta ao tratamento antiviral.
- Estudos epidemiológicos de populações virais.

2.4.2 **RFLP (Polimorfismo de Comprimento de Fragmento de Restrição)**

2.4.2.1 Princípio

O RFLP (Polimorfismo de Comprimento de Fragmento de Restrição) é uma técnica de genotipagem viral que utiliza enzimas de restrição para digerir o DNA viral, gerando fragmentos de tamanhos distintos para diferentes genótipos. A análise dos fragmentos de RFLP em gel de agarose permite a identificação das variantes genéticas presentes no vírus.

2.4.2.2 Variantes

Existem duas variantes principais do RFLP:

- RFLP com Southern Blot: Essa variante envolve a transferência dos fragmentos de DNA para uma membrana e hibridização com sondas radioativas ou fluorescentes específicas para os genótipos.
- RFLP com PCR-RFLP: Essa variante envolve a amplificação por PCR de regiões genômicas virais, seguida da digestão com enzimas de restrição e análise dos fragmentos de RFLP em gel de agarose.

2.4.2.3 Exemplos de Marcas Comerciais

- Inno-Lipa HCV II (Siemens Healthcare): Genotipagem dos genótipos 1-6 do vírus da hepatite C.
- ZEBRA® Zika Genotyping Kit (Zebra Science): Genotipagem dos genótipos asiáticos e americanos do vírus Zika.

2.4.2.4 Características

O RFLP apresenta as seguintes características:

- Alta especificidade e sensibilidade.
- Relativamente barata.
- Útil para genotipagem de um número limitado de alvos genômicos.

2.4.2.5 Aplicações

O RFLP tem diversas aplicações, incluindo:

- Identificação de variantes virais associadas à patogenicidade ou resposta ao tratamento.
- Estudos epidemiológicos de populações virais.

2.4.3 Hibridização de Microrarrays

2.4.3.1 Princípio

A hibridização de microrarrays é uma técnica de genotipagem viral que utiliza microrarrays contendo sondas específicas para diferentes genótipos virais. O DNA viral amplificado é hibridizado com as sondas no microarray, e a análise dos padrões de hibridização permite a identificação das variantes genéticas.

2.4.3.2 Variantes

Existem duas variantes principais da hibridização de microrarrays:

- Microrrays de Oligonucleotídeos (ONA): Essas microrarrays utilizam sondas curtas de DNA com sequências específicas para os genótipos virais. As ONAs hibridizam com regiões complementares do DNA viral amplificado, permitindo a identificação da variante presente.
- Microrrays de cDNA: Essas microrarrays utilizam sondas de cDNA derivadas de diferentes genótipos virais. O DNA viral amplificado hibridiza com a sonda de cDNA complementar, possibilitando a identificação do genótipo.

2.4.3.3 Exemplos de Marcas Comerciais

- VersaChip® HCV Genotype II Assay (Trivitron): Genotipagem dos genótipos 1-6 do vírus da hepatite C.

- Luminex® ARIES® HCV Genotype 2.0 Assay (Luminex Corporation): Genotipagem dos genótipos 1-6 do vírus da hepatite C.

2.4.3.4 Características

A hibridização de microrarrays apresenta as seguintes características:

- Alta especificidade e sensibilidade.
- Permite a genotipagem simultânea de múltiplos alvos genômicos, tornando-a ideal para estudos de alto rendimento.
- Útil para genotipagem de grandes populações virais.

2.4.3.5 Aplicações

A hibridização de microrarrays tem diversas aplicações, incluindo:

- Identificação de variantes virais associadas à resistência a medicamentos.
- Estudos de vigilância viral para rastrear a disseminação de novas variantes.
- Caracterização de surtos virais.

2.4.4 Análise Crítica das Técnicas de Genotipagem Viral sem Sequenciamento: Desafios e Perspectivas

Embora as técnicas de genotipagem viral sem sequenciamento (PCR com primers selecionados, RFLP e hibridização de microrarrays) ofereçam ferramentas valiosas para a caracterização de vírus, é crucial reconhecer suas limitações e desafios, especialmente em um cenário em constante mutação viral.

1. Falsos Negativos e Novas Cepas:

Um dos principais desafios reside na capacidade de detecção de novas cepas virais. Como essas técnicas dependem da identificação de alvos genômicos específicos, o surgimento de novas mutações significativas pode gerar resultados falso-negativos. Isso significa que o vírus pode estar presente, mas não ser detectado pelo método utilizado.

2. Atualização Lenta dos Kits:

Outro desafio reside na atualização dos kits diagnósticos para incorporar novas variantes. O processo de desenvolvimento e validação de novos kits pode ser lento, o que significa que os métodos disponíveis podem não estar sempre atualizados com as cepas virais mais recentes em circulação. Isso pode atrasar o diagnóstico preciso e a tomada de decisões clínicas adequadas.

3. Limitações na Resolução:

As técnicas sem sequenciamento geralmente oferecem resolução limitada na identificação de variantes genéticas. Em alguns casos, mutações sutis em regiões alvo podem não ser detectadas, dificultando a diferenciação entre variantes com características clínicas ou epidemiológicas distintas.

4. Considerações de Custo e Acessibilidade:

O custo dos kits diagnósticos e dos equipamentos necessários para algumas técnicas (por exemplo, microrarrays) pode ser um obstáculo para sua implementação em regiões com recursos limitados. Isso pode gerar disparidades no acesso ao diagnóstico preciso e ao tratamento adequado de infecções virais.

5. Necessidade de Validação Contínua:

É crucial que a performance e a confiabilidade das técnicas de genotipagem viral sem sequenciamento sejam continuamente validadas em diferentes cenários epidemiológicos e com a circulação de novas variantes. Isso garante a qualidade dos resultados e a confiabilidade das informações obtidas para o manejo clínico e a saúde pública.

2.4.4.1 *Perspectivas e Avanços Tecnológicos:*

Apesar dos desafios, a comunidade científica busca constantemente aprimorar as técnicas de genotipagem viral sem sequenciamento e desenvolver novas ferramentas para superar as limitações atuais.

- **Desenvolvimento de Kits Mais Flexíveis:** Kits diagnósticos com alvos genômicos mais abrangentes e adaptáveis podem permitir a detecção de novas variantes com maior eficiência.
- **Aprimoramento da Bioinformática:** Avanços na bioinformática podem auxiliar na análise dos dados de genotipagem, permitindo a identificação de padrões sutis e a diferenciação de variantes com maior precisão. Na seção 2.4.5 descrevemos como o processamento de dados clínicos e laboratoriais com técnicas de AM podem contribuir a melhorar os diagnósticos médicos.
- **Integração com Sequenciamento:** A integração de técnicas de genotipagem sem sequenciamento com métodos de sequenciamento de próxima geração pode oferecer uma solução mais completa e robusta para a caracterização viral.

2.4.4.2 *Conclusão:*

As técnicas de genotipagem viral sem sequenciamento continuam a ser ferramentas valiosas para o diagnóstico e manejo de infecções virais. No entanto, é fundamental

reconhecer suas limitações e investir em pesquisas para o desenvolvimento de métodos mais robustos, flexíveis e acessíveis, capazes de acompanhar a rápida evolução viral e garantir um diagnóstico preciso e oportuno, mesmo frente ao surgimento de novas cepas.

Devido a estas limitações, a genotipagem viral por bioinformática, também conhecida por *in-silico*, é um processo crucial para o estudo e combate a vírus, permitindo a identificação de variantes genéticas, o acompanhamento da evolução viral e o desenvolvimento de medidas eficazes de controle. Como a base da genotipagem bioinformática são as sequências genômicas disponíveis em bases de dados, na seção 2.4.6 fazemos uma rápida descrição do processo padrão (pipeline) para obter as sequências genômicas dos vírus de interesse.

Posteriormente, na seção 2.4.7, descrevemos as técnicas tradicionais de bioinformática para o estudo da evolução de genes virais.

2.4.5 Dados Clínicos como Pedaço Crucial do Quebra-Cabeça: Resolvendo Inconclusões na Genotipagem Viral

Sim, os dados clínicos são ferramentas valiosas para complementar os resultados da genotipagem viral sem sequenciamento, especialmente em situações onde a genotipagem não é conclusiva ou apresenta dúvidas. Ao integrar informações clínicas com os dados de genotipagem, podemos obter uma visão mais completa da infecção viral e tomar decisões mais precisas para o manejo do paciente.

1. Esclarecendo Inconclusões:

- *Histórico de Exposição:* A informação sobre a exposição do paciente a diferentes vírus ou cepas virais pode auxiliar na interpretação de resultados inconclusivos de genotipagem, direcionando a investigação para os genótipos mais prováveis.
- *Manifestações Clínicas:* A correlação entre os sintomas do paciente e os padrões de genotipagem pode fornecer pistas sobre o tipo viral envolvido, mesmo quando a genotipagem não é definitiva.
- *Dados Epidemiológicos:* A prevalência de diferentes genótipos em uma região específica pode auxiliar na interpretação de resultados inconclusivos, considerando a probabilidade de circulação de cada genótipo.

2. Superando Dúvidas na Genotipagem:

- *Confirmando a Presença do Vírus:* Dados clínicos como a presença de marcadores inflamatórios ou alterações laboratoriais sugestivas de infecção viral podem fortalecer a hipótese de infecção, mesmo quando a genotipagem não é definitiva.

- *Avaliando a Severidade da Doença:* A correlação entre os dados clínicos e a gravidade da infecção pode auxiliar na avaliação da patogenicidade do vírus, mesmo quando a genotipagem não identifica um genótipo específico.
- *Monitorando a Resposta ao Tratamento:* A evolução dos sintomas e dos marcadores laboratoriais pode auxiliar na avaliação da resposta ao tratamento antiviral, mesmo quando a genotipagem não fornece informações definitivas sobre a persistência viral.

3. Exemplos Práticos:

- *Paciente com sintomas de hepatite viral, mas genotipagem indefinida para o vírus da hepatite C:* A informação sobre o histórico de exposição a diferentes vírus hepatites, a presença de marcadores laboratoriais específicos e a prevalência de genótipos na região pode auxiliar no diagnóstico e na seleção do tratamento adequado.
- *Paciente com quadro clínico sugestivo de COVID-19, mas genotipagem com resultado negativo para SARS-CoV-2:* A informação sobre a data de início dos sintomas, a presença de comorbidades e o histórico de contato com casos confirmados pode auxiliar na confirmação do diagnóstico, mesmo com genotipagem negativa.

4. Considerações Importantes:

- *Integração cuidadosa:* A análise conjunta de dados clínicos e resultados de genotipagem deve ser realizada por profissionais experientes, considerando todas as informações disponíveis e o contexto individual do paciente.
- *Comunicação eficaz:* A comunicação clara e precisa entre os profissionais de saúde envolvidos no cuidado do paciente é fundamental para garantir a correta interpretação dos dados e a tomada de decisões adequadas.
- *Aprimoramento contínuo:* Pesquisas devem buscar desenvolver ferramentas e algoritmos que facilitem a integração de dados clínicos com os resultados da genotipagem, otimizando a interpretação e o manejo de casos com resultados inconclusivos ou duvidosos.

2.4.5.1 *Conclusão:*

Os dados clínicos são peças essenciais do quebra-cabeça para a interpretação precisa dos resultados da genotipagem viral sem sequenciamento, especialmente em situações onde a genotipagem não é conclusiva ou apresenta dúvidas. Ao integrar informações clínicas com os dados de genotipagem, podemos obter uma visão mais completa da infecção viral, tomar decisões mais precisas para o manejo do paciente e otimizar os resultados do tratamento.

2.4.6 Pipeline de Sequenciamento Genético: Da Amostra ao Depósito da Sequência na Base de Dados

Nesta seção descrevemos o pipeline de sequenciamento, desde a extração do material genético até a anotação das sequências no GenBank (Metzker, 2010), que é fundamental para ter uma visão holística do contexto da pesquisa realizada. Segundo (Chen; Zhou, 2020) o pipeline consta das seguintes etapas:

1. Obtenção da Amostra:

- Tipo de Amostra: A escolha da amostra depende do vírus em estudo, podendo ser sangue, tecido, secreções respiratórias, entre outras.
- Coleta da Amostra: A coleta deve seguir protocolos rigorosos para garantir a integridade do material genético.
- Armazenamento da Amostra: A amostra deve ser armazenada em condições adequadas para preservar o material genético.

2. Extração de Ácido Nucleico:

- Método de Extração: A escolha do método depende do tipo de amostra e do vírus em estudo. Métodos comuns incluem extração por fenol-clorofórmio e kits comerciais.
- Purificação do Ácido Nucleico: O ácido nucleico extraído deve ser purificado para remover contaminantes.

3. Síntese de cDNA (Wacker *et al.*, 2002):

- Transcrição Reversa: Se o material genético for RNA, a transcrição reversa é necessária para convertê-lo em cDNA (DNA complementar).
- Síntese de cDNA: A síntese de cDNA é realizada usando a enzima transcriptase reversa e primers específicos.

4. Amplificação por PCR (Mullis; Faloona, 1987):

- Design de Primers: Primers específicos para o genoma viral são projetados para amplificar a região de interesse.
- Reação de PCR: A reação de PCR amplifica o DNA ou cDNA viral, gerando múltiplas cópias da região de interesse.

5. Purificação do Produto de PCR:

- Métodos de Purificação: O produto de PCR é purificado para remover primers, reagentes do PCR (dNTPs) e outros contaminantes.

- Colunas de Separação: Colunas de centrifugação ou kits de purificação por sílica são comumente utilizados.

6. Sequenciamento de DNA:

- Tecnologia de Sequenciamento: Diversas tecnologias de sequenciamento estão disponíveis, como Sanger (Sanger; Nicklen; Coulson, 1977), NGS (Next-Generation Sequencing)(Metzker; L., 2010) e Illumina (Bentley *et al.*, 2008).
- Geração de Leitura: O sequenciamento gera leituras curtas ou longas da sequência de DNA.

7. Montagem e Análise de Sequências:

- Montagem de Contigs: As leituras curtas de sequenciamento são montadas em contigs (sequências sintéticas maiores formadas pela sobreposição de fragmentos menores sequenciados usando técnicas de bioinformática), reconstruindo a sequência completa do genoma viral (Pop; Kosack; Salzberg, 2004).
- Análise da Qualidade das Sequências: Softwares bio-informáticos são utilizados para analisar a qualidade do sequenciamento, podendo também identificar variantes genéticas, mutações e outras características, nesse processo (Cock *et al.*, 2009).

8. Anotação e Depósito no GenBank:

- Anotação da Sequência: A sequência viral é anotada com informações sobre genes, proteínas e outras características (Boeckmann *et al.*, 2003).
- Depósito no GenBank: A sequência anotada é submetida ao GenBank, um banco de dados público de sequências de DNA.

Vale a pena destacar, que os primeiros 6 passos descritos aqui, descrevem mais detalhadamente o pipeline no laboratório molhado (in-vitro) mostrado na figura ???. Os outros passos ocorrem no laboratório seco (in-silico).

2.4.7 Técnicas Tradicionais de Bioinformática para o Estudo da Evolução de Sequências Virais

Na seção anterior, exploramos o trajeto que as amostras virais percorrem, desde a coleta até o depósito de suas sequências genômicas no GenBank. Agora, abordamos o método tradicional de estudar a evolução viral usando essas sequências, que é baseado na Filogenia Molecular.

A filogenia, consiste no estudo das relações evolutivas entre organismos e é uma ferramenta fundamental na virologia para entender como os vírus evoluem e se diversificam.

Técnicas filogenéticas são usadas para traçar a origem e a propagação de diferentes cepas virais, permitindo a identificação de linhagens emergentes e a previsão de possíveis surtos futuros (Felsenstein, 1985). No contexto do SARS-CoV-2, por exemplo, a análise filogenética tem sido usada para monitorar a emergência de novas variantes e suas implicações epidemiológicas (Rambaut *et al.*, 2020).

Com um vasto acervo de sequências à disposição, podemos realizar análises em larga escala, abrangendo diferentes regiões geográficas e um número expressivo de cepas virais. Essa abordagem poderosa nos permite:

1. **Identificar novos genótipos:** Através da seleção de genes específicos, extração de sequências, alinhamentos precisos, curadoria meticolosa e construção de árvores filogenéticas robustas baseadas na contagem de SNPs (Polimorfismo de Nucleotídeo Simples (Brookes, 1999)) e no relógio molecular(Kumar *et al.*, 2017), podemos identificar novas variantes virais, expandindo nosso conhecimento sobre a diversidade viral.
2. **Traçar redes de transmissão:** As árvores filogenéticas, quando combinadas com dados epidemiológicos e de geolocalização, revelam as rotas de transmissão viral, permitindo o rastreamento de surtos e a implementação de medidas de controle mais eficazes.

O estudo da evolução viral em grande escala exige um pipeline bio-informático robusto e bem estruturado, composto por etapas cruciais descritas a seguir:

1. **Seleção do(s) gene(s):** A escolha do gene alvo depende do objetivo específico do estudo. Genes com alta taxa de mutação, como aqueles que codificam proteínas estruturais, são mais propensos a revelar eventos de diversificação viral. Contudo, genes que produzem proteínas expostas são mais relevantes desde o ponto de vista de infectividade e evasão do sistema imunológico,
2. **Extração de sequências:** As sequências de interesse são extraídas do GenBank ou de outras bases de dados relevantes, considerando critérios rigorosos de qualidade e confiabilidade.
3. **Alinhamento de sequências:** Alinhamento de sequências: Algoritmos de alinhamento precisos, como MAFFT (Katoh; Standley, 2013), Clustal Omega (Sievers *et al.*, 2011) e MINIMAP (Li, 2016), garantem a sobreposição correta das sequências, minimizando erros e maximizando a acurácia das análises subsequentes.
4. **Curadoria de sequências:** A curadoria manual ou automatizada remove sequências de baixa qualidade, duplicadas ou contaminadas, garantindo a confiabilidade do conjunto de dados.

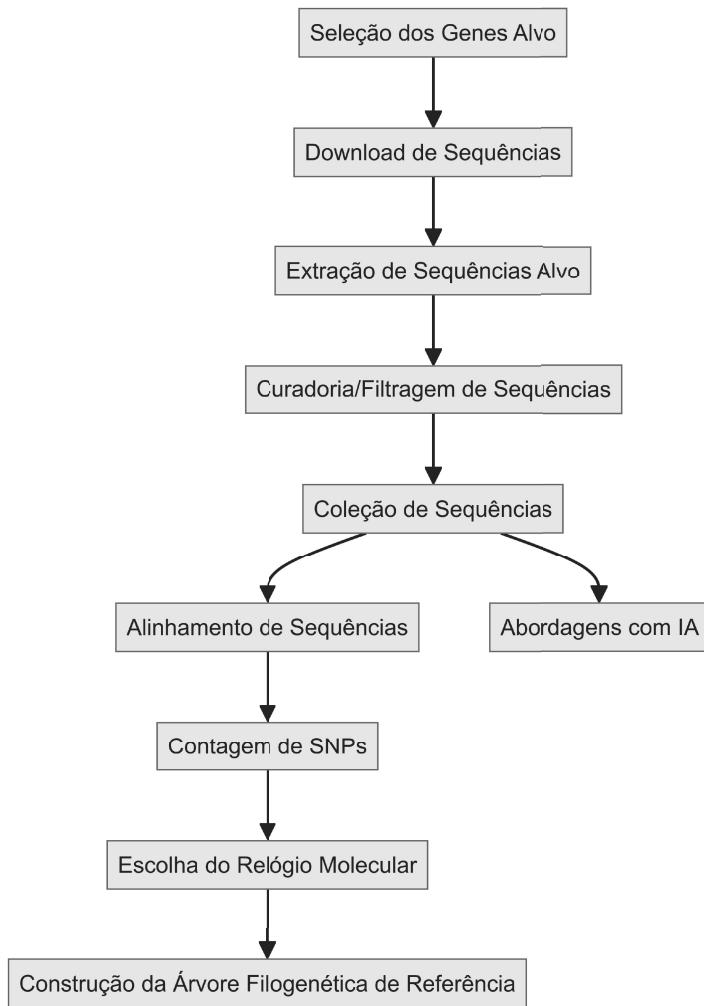


Figura 3 – Pipeline Típico para a Construção da Árvore de Referência Filogenética para Genotipagem Viral

5. **Análise de SNPs e relógio molecular:** A contagem de SNPs e a aplicação do relógio molecular permitem estimar taxas de mutação e divergência viral, inferindo datas de eventos evolutivos e padrões de dispersão geográfica.
6. **Construção de árvores filogenéticas:** Diversos métodos, como Maximum Likelihood (ML) (Saitou; Nei, 2013) ou Neighbor-Joining (NJ) (Saitou; Nei, 1987), são utilizados para construir árvores filogenéticas robustas, revelando as relações evolutivas entre as cepas virais.

A identificação do genótipo de novas sequências requer a comparação com as já conhecidas. Para isto, se constrói a árvore filogenética de referência, como mostrado na figura 3, e descrito na seção 2.4.7.1.

2.4.7.1 Árvores Filogenéticas de Referência

As árvores filogenéticas de referência, construídas a partir de um conjunto abrangente de sequências virais, servem como ferramentas valiosas para genotipar novas cepas sequenciadas. Ao posicionar uma nova sequência na árvore, podemos determinar seu genótipo e inferir sua relação evolutiva com outras cepas conhecidas. Essa abordagem é crucial para a vigilância viral, permitindo a rápida identificação e caracterização de novas variantes que podem representar riscos à saúde pública.

2.4.7.2 Considerações Adicionais

- **Bases de Dados:** Além do GenBank, outras bases de dados como o ViPR e o Nextstrain fornecem acesso a sequências virais e ferramentas bioinformáticas para análise da evolução viral.
- **Ferramentas Bioinformáticas:** Diversos softwares e plataformas online, como BEAST (Suchard *et al.*, 2018), PhyloSuite (Zhang *et al.*, 2020) e Genome Detective (Vilsker *et al.*, 2019), facilitam a construção e análise de árvores filogenéticas e a aplicação de métodos de relógio molecular.

Na figura 3 indicamos um fluxo alternativo depois da **Coleção de Sequências** sob o identificador **Abordagens com IA**. Nas próximas seções nos desbruçaremos na descrição desse ramo, começando na seção 2.5 com as Técnicas de Aprendizado de Máquina, que são os métodos da IA aplicáveis para a solução do problema posto: genotipagem viral.

2.5 Técnicas de Aprendizado de Máquina

2.5.1 Definição

O Aprendizado de Máquina (AM) é um campo da Inteligência Artificial (IA) que se concentra no desenvolvimento de algoritmos que podem aprender com dados e melhorar seu desempenho em tarefas específicas sem serem explicitamente programados (Goodfellow; Bengio; Courville, 2016). Essa capacidade de aprender e se adaptar torna o AM uma ferramenta poderosa para diversos campos, como diagnóstico médico, análise financeira, reconhecimento de imagens e tradução automática, entre muitas outras (Géron; Aurélien, 2019).

2.5.2 Tipos de Aprendizado de Máquina

As técnicas de AM podem ser divididas em duas categorias principais:

- **Aprendizado Supervisionado:** Nesse tipo de aprendizado, o algoritmo recebe um conjunto de exemplos rotulados, onde cada exemplo possui um conjunto de atributos de entrada e um atributo de saída (rótulo). O objetivo do algoritmo é aprender a mapear os atributos de entrada para o rótulo de saída (James *et al.*, 2013). Por exemplo:
 - Redes Neurais Artificiais (RNA) do tipo Multilayer Perceptron (MLP) (Haykin, 2009)
 - Máquinas de Vetores de Suporte (MVS) (Cortes; Vapnik, 1995)
 - Algoritmos Genéticos (AGs) (Holland, 1992)
 - Árvores de Decisão (Breiman *et al.*, 1984)
- **Aprendizado Não Supervisionado:** Nesse tipo de aprendizado, o algoritmo recebe apenas um conjunto de exemplos não rotulados. O objetivo do algoritmo é encontrar padrões ou agrupamentos nos dados sem a necessidade de rótulos pré-definidos (Bishop, 2006). Por exemplo:
 - Redes Neurais Artificiais do tipo Mapa Auto-Organizável(SOM) (Kohonen, 1988)
 - Algoritmo K-Means (MacQueen, 1961)
 - Algoritmos de Agrupamento Hierárquico (Ward, 1963)

2.5.3 Conceitos Importantes

- Exemplo (padrão, instância, amostra): Uma unidade de informação a partir da qual um modelo será aprendido ou utilizado. Na maioria dos casos, exemplos são descritos por vetores de características (Géron; Aurélien, 2019).
- Característica (atributo, variável - *feature*): Uma propriedade que descreve um exemplo. Cada atributo possui um domínio definido por seu tipo, que determina os valores que ele pode assumir. Além disso, os atributos podem ser numéricos (representados por variáveis inteiras ou reais) ou categóricos (representados por caracteres ou cadeias de caracteres - *strings*) (James *et al.*, 2013).
- Vetor de características: Uma lista de características que descreve um exemplo (Géron; Aurélien, 2019).
- Classe: No aprendizado supervisionado, cada exemplo possui um atributo especial chamado rótulo ou classe, que indica a categoria à qual ele pertence (James *et al.*, 2013).

- Conjunto de exemplos (conjunto de dados, conjunto de treinamento): Um conjunto de exemplos com seus respectivos valores de atributos. No aprendizado supervisionado, cada exemplo também possui um rótulo associado (James *et al.*, 2013).
- Conjunto de treinamento desbalanceado: Quando o número de exemplos de algumas classes é muito maior que o de outras classes (James *et al.*, 2013).
- Ruído: Imperfeições nos dados, como erros de medição ou rotulagem incorreta (Géron; Aurélien, 2019).
- Overfitting (superajuste): Ocorre quando o modelo se adapta excessivamente aos dados de treinamento, apresentando baixo desempenho em novos dados (Géron; Aurélien, 2019).
- Classificação Incorreta: Um exemplo de determinada classe A foi classificado como de outra qualquer classe (James *et al.*, 2013).
- Exemplo Não Classificado: Um exemplo de determinada classe A que não foi classificado como nenhuma das classes existentes, nem mesmo da classe à qual aparentemente pertence (James *et al.*, 2013). Geralmente representa um caso de ruído por anotação errada ou por atributos incorretos.
- Métricas de Desempenho para Classificadores Supervisionados Binários (índices de qualidade)
 - Falso positivo: Um exemplo da classe B (negativa) classificado como da classe A (positiva) (James *et al.*, 2013).
 - Falso negativo: Um exemplo da classe A (positiva) classificado como da classe B (negativa) (James *et al.*, 2013).
 - Acurácia: A proporção de previsões corretas feitas pelo modelo em um conjunto de dados (Géron; Aurélien, 2019).
 - Precisão (Precision): A proporção de verdadeiros positivos entre os exemplos classificados como positivos (James *et al.*, 2013).
 - Revocação (Recall): A proporção de verdadeiros positivos entre os exemplos que são realmente positivos (James *et al.*, 2013).
 - F1-Score: A média harmônica entre precisão e revocação, proporcionando um equilíbrio entre os dois (James *et al.*, 2013).
 - Curva ROC e AUC: A curva ROC (Receiver Operating Characteristic) representa a relação entre a taxa de verdadeiros positivos e a taxa de falsos positivos. A área sob a curva (AUC) é uma medida do desempenho do modelo (Géron; Aurélien, 2019).

- Métricas de Desempenho para Aprendizado Não Supervisionado: As métricas de desempenho para aprendizado não supervisionado, como a análise de clusters, são usadas para avaliar a qualidade dos agrupamentos sem rótulos de classe pré-definidos.
 - Índice de Silhueta: Mede o quanto semelhante um exemplo é ao seu próprio cluster (coesão) em comparação com outros clusters (separação). Varia de -1 a 1, onde valores altos indicam que os exemplos estão bem correspondidos ao seu próprio cluster e mal correspondidos a clusters vizinhos (Rousseeuw, 1987).
 - Índice de Davies-Bouldin: Calcula a média das razões das distâncias dentro do cluster para a distância entre os clusters. Valores mais baixos indicam melhores separações entre os clusters (Davies; L; Bouldin, 1979).
 - Coeficiente de Variação Intracluster (WCV): Mede a dispersão dentro dos clusters. Valores menores indicam clusters mais compactos.
 - Índice de Rand Ajustado (ARI): Avalia a similaridade entre a clusterização obtida e uma clusterização de referência, corrigida pelo acaso. Valores próximos de 1 indicam alta similaridade (Hubert; Lawrence; Arabie, 1985).
 - Índice de Dunn: Mede a menor distância entre os pontos de diferentes clusters dividida pela maior distância intracluster. Valores mais altos indicam melhor clusterização (Dunn, 1974).
- Métricas de Desempenho para Aprendizado Supervisionado com Múltiplas Classes: As métricas de desempenho para aprendizado supervisionado com múltiplas classes são usadas quando os dados de treinamento possuem rótulos de múltiplas classes.
 - Acurácia Global: A proporção de previsões corretas feitas pelo modelo em um conjunto de dados com múltiplas classes (Géron; Aurélien, 2019).
 - Precisão por Classe: A proporção de verdadeiros positivos entre os exemplos classificados como positivos para cada classe individualmente (James *et al.*, 2013).
 - Revocação por Classe: A proporção de verdadeiros positivos entre os exemplos que são realmente positivos para cada classe individualmente (James *et al.*, 2013).
 - F1-Score por Classe: A média harmônica entre precisão e revocação para cada classe (James *et al.*, 2013).
 - Matriz de Confusão: Uma tabela que descreve o desempenho do modelo mostrando as previsões corretas e incorretas de cada classe (Géron; Aurélien, 2019).
 - Acurácia Balanceada: A média da acurácia para cada classe, útil para conjuntos de dados desbalanceados (Brodersen *et al.*, 2010).

- Micro/macro/média ponderada: Diferentes formas de calcular a média de precisão, revocação e F1-Score. A média micro agrupa todas as classes como uma, a média macro calcula a média simples entre as classes, e a média ponderada leva em consideração o número de exemplos em cada classe (Géron; Aurélien, 2019).

2.6 O Aprendizado de Máquina Aplicado na Genômica

O aprendizado de máquina, uma área muito importante da Inteligência Artificial (OECD, 2024), tem se mostrado uma ferramenta poderosa na análise de grandes volumes de dados genômicos. A seguir listamos algumas aplicações:

- Algoritmos de aprendizado de máquina podem ser treinados para identificar padrões em sequências de DNA, como por exemplo para identificação de regiões codificantes e intrones, regiões promotoras, estruturas secundárias de proteínas, entre outros. Estes algoritmos são utilizados principalmente na anotação de novos genomas.
- Redes neurais convolucionais e recorrentes (conhecidas como técnicas de Aprendizado Profundo) podem ser utilizadas para analisar sequências genômicas e identificar relações complexas entre diferentes genes e proteínas. Estas ferramentas são essenciais para desvendar as redes de regulação gênica.
- Mediante os modelos treinados para Processamento de Linguagem Natural (PLN) a IA pode processar e analisar grandes volumes de texto científico, como artigos e relatórios, para extrair informações relevantes e identificar novas hipóteses de pesquisa.
- Modelos de aprendizado de máquina também podem ser desenvolvidos para identificar mutações associadas à virulência ou resistência a medicamentos de patógenos vírais. Este tipo de problema é conhecido com a predição do fenótipo a partir do genótipo.

Na seção (2.7) descrevemos as abordagens identificadas que usam de técnicas de AM para genotipagem viral.

2.7 Aprendizado de Máquina para Genotipagem

Na revisão da literatura encontramos que existem duas abordagens bem diferenciadas de AM para genotipagem viral: (1) As baseadas em dados clínicos e exames laboratoriais de amostras contendo o vírus (sem sequenciamento) e (2) As baseadas em sequenciamento do vírus.

A primeira abordagem está fora do escopo deste trabalho, pelo que focaremos na segunda abordagem, onde se utiliza a sequência do material genético do vírus. Nesta abordagem precisamos diferenciar duas abordagens: (1) baseada em atributos numéricos e (2) baseada em atributos categóricos.

Na opção de atributos numéricos, existem diversas propostas para extrair das sequências atributos numéricos, que são descritos na seção 2.7.1.

Já no caso de atributos categóricos, somente conhecemos a abordagem adotada neste trabalho. Na seção 2.7.2 descrevemos: (a) o método utilizado para a extração de atributos categóricos das sequências e (b) método para clusterizar as sequências representadas como listas de atributos categóricos (CLOPE).

Na figura 4 mostramos um esquema com a diversidade de abordagens investigadas.

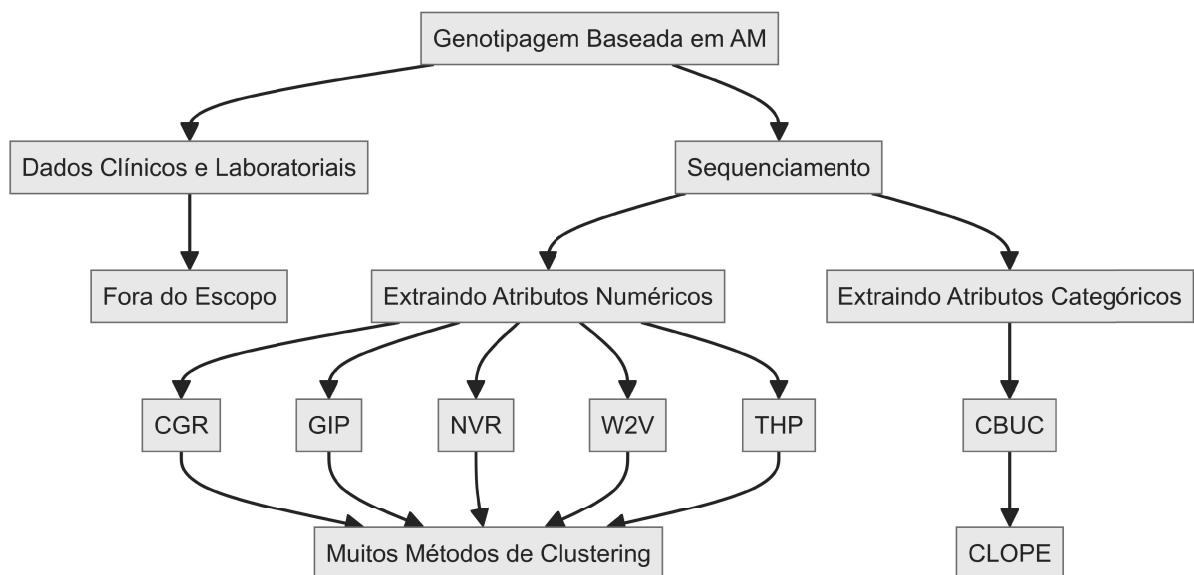


Figura 4 – Caption

2.7.1 *Métodos de Extração de Atributos Numéricos de Sequências de DNA*

A representação de sequências de DNA é um passo fundamental para diversos estudos em bioinformática, permitindo a análise, comparação e interpretação de dados genéticos. Superando a limitação da sequência básica de bases nitrogenadas (A, C, G e T), diversas abordagens alternativas surgem para converter as sequências em formatos mais adequados para técnicas de aprendizado de máquina e análise estatística.

1. **Genome Image Pattern (GIP)** (Delibas; Arslan, 2020): Transformando DNA em Imagens Unidimensionais

- *Descrição:* O GIP propõe uma representação visual das sequências de DNA, convertendo cada base nitrogenada em um pixel com cor específica. Essa conversão gera uma imagem unidimensional que pode ser analisada por redes convolucionais unidimensionais (1D CNNs).
- *Vantagens:* Visualização intuitiva: O GIP facilita a visualização das sequências de DNA, permitindo a identificação de padrões visuais que podem ser relevantes para a análise.
- *Eficiência computacional:* As 1D CNNs são modelos de aprendizado de máquina relativamente eficientes, capazes de processar grandes conjuntos de dados de sequências de DNA com rapidez.
- *Desvantagens:*
 - > Perda de informação: A conversão em imagem pode levar à perda de informações sutis presentes na sequência original de bases nitrogenadas.
 - > Limitação de aplicações: O GIP pode ser mais adequado para tarefas de classificação ou reconhecimento de padrões, mas pode não ser ideal para análises mais complexas que exigem a preservação da informação original da sequência.

2. Chaos Game Representation (CGR) (Löchel; Heider, 2021): Fractais Desvendando o DNA

- *Descrição:* O CGR utiliza a geometria fractal do Chaos Game para gerar uma representação de sequências de DNA como uma nuvem de pontos num plano ou num cubo de projeção. A sequência é percorrida e cada base nitrogenada é mapeada para um ponto nessa área ou volume de projeção, criando padrões fractais vissíveis. Essa representação fractal captura características intrínsecas da sequência, como repetições e padrões de longo alcance. A projeção completa de cada sequência pode ser armazenada e usada posteriormente para comparação utilizando técnicas de processamento de imagens ou simples comparação de matrizes, ou podem ser usadas para extrair características integrais como o Centroide e a Frequência do caos, como em (Tanchotsrinon; Lursinsap; Poovorawan, 2015) usado para genotipagem de vírus da Hepatite C.
- *Vantagens:*
 - > Preservação de informações: O CGR preserva informações importantes da sequência original, incluindo padrões de longo alcance e repetições.
 - > Aplicações diversas: O CGR pode ser utilizado para diversas tarefas em bioinformática, como comparação de sequências, filogenia e detecção de mutações.
- *Desvantagens:*
 - > Interpretação complexa: A interpretação dos vetores CGR pode ser complexa e desafiadora, exigindo conhecimento em geometria fractal e bioinformática.

> Sensibilidade ao ruído: O CGR pode ser sensível ao ruído na sequência de DNA, o que pode afetar a precisão da representação.

3. Natural Vector Representation (NVR) (Yu; Yau, 2024): Contando Bases e k-mers para Descrever o DNA

- *Descrição:* A NVR transforma sequências de DNA em vetores numéricos contando a frequência de bases e k-mers (subsequências de tamanho k) em cada posição da sequência. Essa representação captura informações sobre a composição local e regional da sequência, permitindo a comparação e análise usando técnicas de aprendizado de máquina e estatística.
- *Vantagens:*
 - > Simplicidade: A NVR é uma representação simples e intuitiva, fácil de implementar e interpretar.
 - > Versatilidade: A NVR pode ser utilizada para diversas tarefas em bioinformática, como classificação de sequências, detecção de genes e análise de expressão gênica.
- *Desvantagens:*
 - > Perda de informações de longo alcance: A NVR se concentra na composição local e regional da sequência, podendo perder informações de longo alcance que podem ser importantes para algumas tarefas.
 - > Alta dimensionalidade: A NVR pode gerar vetores de alta dimensionalidade, o que pode dificultar a análise e interpretação dos dados.

4. Word2vec e Doc2vec (W2V) (Wahab *et al.*, 2021): Capturando Semântica no DNA

- *Descrição:* Inspiradas no processamento de linguagem natural, técnicas como Word2vec e Doc2vec convertem sequências de DNA em vetores numéricos. Essas técnicas consideram a ordem e a proximidade das bases nitrogenadas, capturando relações semânticas entre elas e permitindo a identificação de similaridades funcionais entre sequências.
- *Vantagens:*
 - > Captura de relações complexas: Word2vec e Doc2vec podem capturar relações complexas entre bases nitrogenadas que podem não ser aparentes em representações baseadas em simples contagens.
 - > Potencial para novas descobertas: Essas técnicas podem auxiliar na descoberta de novas funções e interações moleculares baseadas em similaridades identificadas nas sequências de DNA.
- *Desvantagens:*

> Interpretação desafiadora: A interpretação dos vetores gerados por Word2vec e Doc2vec pode ser complexa, exigindo técnicas avançadas de análise de dados.

> Necessidade de grandes quantidades de dados: Essas técnicas geralmente requerem grandes conjuntos de dados de sequências de DNA para um aprendizado eficiente.

5. Propriedades Termodinâmicas (THP) (SantaLucia; Hicks, 2004): Representando a Estabilidade do DNA

- *Descrição:* A representação baseada em propriedades termodinâmicas considera as características energéticas das ligações entre bases nitrogenadas. Essas propriedades, como energia livre de Gibbs e entropia, fornecem informações sobre a estabilidade da molécula de DNA e a probabilidade de certos padrões de sequências ocorrerem.

- *Vantagens:*

- > Informações sobre estabilidade: A representação termodinâmica oferece insights sobre a estabilidade estrutural do DNA, o que pode ser relevante para estudos de regulação gênica e interações proteína-DNA.

- > Complementaridade com outras abordagens: A representação termodinâmica pode ser usada em conjunto com outras abordagens para obter uma visão mais abrangente das sequências de DNA.

- *Desvantagens:*

- > Complexidade da modelagem termodinâmica: A modelagem termodinâmica de DNA pode ser complexa e requer ferramentas computacionais especializadas.

- > Interpretação nem sempre intuitiva: A interpretação dos resultados termodinâmicos pode exigir conhecimento específico em biofísica molecular.

2.7.1.1 **Discussão:**

A escolha da melhor representação de sequências de DNA depende do contexto específico da análise e dos objetivos do estudo. É importante considerar fatores como:

- A natureza da tarefa (classificação, detecção de genes, análise de expressão gênica etc.)
- O tamanho e a complexidade dos dados de sequenciamento
- A disponibilidade de recursos computacionais

Além disso, diferentes representações podem ser combinadas para capturar informações complementares sobre as sequências de DNA. A pesquisa em bioinformática

segue explorando novas técnicas para representação de sequências, ampliando o arsenal de ferramentas disponíveis para desvendar os segredos do código genético.

2.7.2 Construindo Conjuntos de Treinamento com Atributos Categóricos

A representação natural de uma sequênciade DNA como atributos categóricos é a cadeia de carateres formadas pelos nucleotídeos em cada posição, que são representados por carateres seguindo o código IUPAC [REF]. Os nucleotídeos bem identificados nessas sequências são representados pelos carateres A, G, C e T, correspondendo às bases nitrogenadas Adenina, Guanina, Citosina e Tiamina, respectivamente. Essa é a representação usada pelos métodos filogenéticos tradicionais, que não é a usada neste trabalho.

A nossa abordagem é baseada nas seguintes considerações:

1. *A diversidade da realidade é melhor descrita quanto maior a variabilidade dos atributos independentes considerados para avaliar os sujeitos da população estudada:*

Considere um atributo a_1 que possui $n_1 > 1$ variantes. Então, as amostras serão classificadas em n_1 classes ou categorias ou classes. Considere agora outro atributo a_2 que possui $n_2 > n_1$ variantes. É óbvio que as amostras serão classificadas num número maior de categorias ou classes, permitindo uma descrição mais detalhada (nítida) da realidade se usarmos o atributo a_2 em lugar do a_1 .

2. *As mutações em regiões não codificantes dos vírus não produzem nenhuma mudança fenotípica/funcional que tenha influência na genotipagem:*

De acordo com isto somente tem sentido na genotipagem viral considerar as regiões codificantes (CDS/ORF) do genoma dos vírus estudados.

Combinando as duas considerações acima, nossa abordagem considera como atributos os códons em cada posição das ORFs ou de partes delas, que codificam todas ou alguma(s) proteína(s) viral(is) de interesse para a genotipagem.

- **Vantagens:**

- > Melhor descrição da realidade: Enquanto existem apenas 4 nucleotídeos e 20 aminoácidos, existem 61 códons codificantes (descontando os 3 códons de parada). Ou seja, os códons são a melhor escolha natural de acordo com o primeiro critério acima.
- > Maior compactação: O número de atributos por amostra é reduzido 3 vezes com relação ao número de nucleotídeos.

> Reversibilidade: A sequência original da amostra pode ser gerada sem ambiguidade a partir da sequência de atributos (códons) gerada na transformação de DNA para a sequência de atributos categóricos.

- **Desvantagem:** Para trabalhar com códons como atributos todas as sequências codificantes alvo (não necessariamente completas) precisam estar alinhadas e no mesmo quadro de leitura, preferencialmente no quadro correto para poder fazer estatísticas de códons (uso de códons).

A necessidade de alinhamento das sequências tem sido considerado uma série limitação das técnicas de filogenia. De fato, a complexidade computacional (tempo de computação) de um alinhamento múltiplo é elevado, com complexidade supra-linear com o número e comprimento das sequências sendo alinhadas. Contudo:

- Diversos algoritmos modernos de alinhamento, muito eficientes, seriais e paralelos (usando principalmente multi-core e GPUs) têm sido desenvolvidos e estão disponíveis online e/ou como bibliotecas ou APIs. Como nosso objetivo não está diretamente relacionado com o alinhamento múltiplo de sequências, sugerimos que o leitor interessado neste assunto consulte (Chao; Tang; Xu, 2022).
- Para fins de análise genotípica, é suficiente o alinhamento de cada sequência do conjunto de treinamento contra a sequência de referência, o que transforma a complexidade computacional para linear. As sequências que precisam ser classificadas usando o modelo treinado também precisam apenas serem alinhadas contra a sequencia de referência, o que não constitui um gargalo de processamento.
- Desta forma, cada sequência do conjunto de treinamento ou nova sequência a ser classificada, precisa apenas ser alinhada contra a sequência de referência. Isto permite que o conjunto de treinamento possa ser atualizado com frequência sem alto custo computacional, toda vez que uma nova sequência (com padrão diferente daqueles presentes no conjunto de treinamento) seja encontrada. Atualizando periodicamente o conjunto de treinamento constitui de fato uma atualização do acervo da diversidade genética da espécie estudada, pelo que é recomendável atualizar o conjunto de treinamento regularmente com todas as sequências processadas e de qualidade.

2.7.2.1 *Conversão das Sequências de Nucleotídeos a Sequências de Códons*

Para simplificar a notação dos códons, atribuímos a cada triplete de nucleotídeos de cada sequência do dataset de treinamento, alinhada com a sequência de referência no quadro de leitura, um número inteiro de 1 a 64, mas que é usado como atributo categórico, não como número em si.

Isto é feito em duas etapas: Primeiro fazemos uma numeração dos nucleotídeos $N = \{A, G, C, T\}$ da forma: $n[A] = 1, n[G] = 2, n[C] = 3$ e $n[T] = 4$.

Um códon vem dado pelas bases nas três posições: $C = N_1 N_2 N_3$, onde $N_i \in \{A, G, C, T\}$ é a base na posição $i = 1, 2, 3$, do códon.

O segundo passo é transformar cada nucleotídeo do códon para sua versão numerada, ou seja $n[C] = [n(N_1), n(N_2), n(N_3)]$.

Por último, com $n[C]$ dado atribuímos ao códon um numeral $c \in [1, 64]$ usando a fórmula seguinte:

$$c = n(N_3) + 4(n(N_2) - 1) + 16(n(N_1) - 1) \quad (2.1)$$

Exemplos:

- Dado o códon $C = AAA$. Então $n[C] = [n(A), n(A), n(A)] = [1, 1, 1]$, pelo que o numeral atribuído é $c = 1 + 4(1 - 1) + 16(1 - 1) = 1$
- Dado o códon $C = TTT$. Então $n[C] = [n(T), n(T), n(T)] = [4, 4, 4]$, pelo que o numeral atribuído é $c = 4 + 4(4 - 1) + 16(4 - 1) = 64$

Usando a fórmula 2.1 transformamos qualquer sequência codificante de L nucleotídeos numa sequência de $L/3$ códons numerados de 1 a 64.

2.7.2.1.1 Observações:

1. **Excluindo sequências idênticas:** Incluir sequências redundantes no conjunto de treinamento apenas aumenta o volume de dados e o tempo de preicssamento, sem introduzir nenhuma informação relevante, pelo que sequências idênticas devem ser filtradas.
2. **Excluindo sequências com stop codons:** Como as sequências codificantes não podem possuir códons de parada no quadro de leitura, se for encontrado algum desses códons (TAA, TAG ou TGA , que correspondem aos numerais 49, 50 e 53, respectivamente) a sequência é excluída.
3. **Tratamento de gaps:** É comum que existam inserções e deleções de códons nas sequências alinhadas. Quando um códon é inserido em uma das sequências, isso causa o surgimento de gaps, representados com $[-]$ nas outras sequências alinhadas. Da mesma forma, quando um códon é deletado de uma sequência, aparece um gap $[-]$ na posição do códon deletado nessa sequência. Neste trabalho atribuímos o numeral 65 aos gaps detectados.

4. Tratamento de nucleotídeos indeterminados: Chamamos nucleotídeos indeterminados a qualquer caráter na sequência de DNA que não seja A, G, C ou T(U). O mais comum é o caráter N que indica qualquer nucleotídeo e ocorre quando a base não foi identificada corretamente durante o sequenciamento. Outros caracteres representam indeterminações menos fortes, como por exemplo R significa purina, que pode ser A ou G, e Y representa uma pirimidina, ou seja, C ou T. A lista de caracteres distintos de A, G, C, T, é definida pela notação IUPAC.

Em qualquer situação, é necessário definir uma política de aceitação/rejeição de sequências com esses caracteres. Incluir sequências com caracteres indeterminados no conjunto de treinamento é factível, mas introduz ruído diminuindo a qualidade classificatória do modelo treinado.

A regra deve ser apenas considerar sequências sem caracteres indeterminados na hora de formar o conjunto de treinamento. Contudo, se por causa de falta de sequências for decidido considerar sequências com caracteres indeterminados, os códons que contêm bases indeterminadas devem ser eliminados de todo o alinhamento, deixando apenas sítios onde todos os códons estão determinados. Isto logicamente complica tanto a atualização/ampliação do conjunto de treinamento quanto o pre-processamento das sequências que serão classificadas usando o modelo treinado.

Na seção 2.7.2.2 descrevemos o procedimento para extração de atributos a partir das sequências traduzidas para códons que formam o conjunto de treinamento.

2.7.2.2 Extração de Atributos das Sequências de Códons

Denotemos por \mathcal{S}_n o conjunto de treinamento formado por N sequências com L nucleotídeos alinhados no quadro de leitura, e por \mathcal{S}_c a versão do conjunto de treinamento formado por N sequências com $M = L/3$ códons, tal que $\mathcal{S}_c[i, j]$ denota o códon na sequência $i = 1, 2, \dots, N$ na posição $j = 1, 2, \dots, M$.

Lembrando que os códons são numerais de 1 a 65 (excepto códons de parada: 49, 50 e 53), ou seja, $\mathcal{S}_c[i, j] \in \{1, 2, \dots, 48, 51, 52, 54, \dots, 65\}$ para toda sequência i e posição j .

O processo de extração é bem simples, consistindo na varredura de todas as posições de esquerda a direita, identificando posições polimórficas, ou seja, onde há códons distintos.

Os atributos tem o formato $[posição : valor]$ que é compatível com o algoritmo de classificação utilizado CLOPE (Yang; Guan; You, 2002). O algoritmo Python utilizado é descrito a seguir:

Listing 2.1 – Finding polymorphic positions and generating the training set for CLOPE

```
# Finding polymorphic positions
```

```

pos = [] # Initializing list of polymorphic positions
for j in range(M): # Loop through positions
    if len(set(Sc[:,j])) > 1: # if there is more than one codon type
        pos.append(j) # add position j to the list

# Generating the training set for CLOPE
T = [] # Initializing the training dataset (list of lists of attributes)
for i in range(N): # Loop through sequences
    t = [] # Initializing the attribute list for sequence i
    for p in pos: # loop over polymorphic positions
        t.append(str(p) + ":" + str(Sc[i,p])) # Updating the attribute
        list of sequence i
    T.append(t) # Updating the training set

```

2.7.2.2.1 Estrutura e Papéis dos Dados do Modelo de Referência:

O Modelo de Referência/Treinamento é uma estrutura de dados heterogênea que contem:

1. A sequência de referência, $S_{n,ref}$, utilizado para alinhar novas sequências. Estas novas sequências podem ser candidatas a serem incorporadas no conjunto de treinamento ou enviadas para classificação.
2. O conjunto de sequências de treinamento em nucleotídeos, S_n , mantido como backup para futuras análises se for preciso.
3. O conjunto de sequências de treinamento em códons, \mathcal{S}_c , utilizada para a busca de posições polimórficas toda vez que uma nova sequência é adicionada ao conjunto de treinamento.
4. O vetor de posições polimórficas, pos , utilizado para a extração de atributos das sequências enviadas para classificação.
5. A matriz de atributos, T , é um inventário dinâmico da diversidade da espécie viral correspondente formatada da forma que o método de clusterização CLOPE lê o conjunto de treinamento. Contem apenas informação das posições polimórficas. Devido a isto, toda vez que o vetor de posições polimórficas pos for atualizado, a matriz T precisa ser atualizada.

Por se tratar de um inventário, ela por si só serve para pesquisar se uma sequência de entrada é nova ou é idêntica a alguma outra já incluída no modelo. Em outras palavras, T é projetada para ser uma base de conhecimento da bio-diversidade genética viral a nível de códons.

Devido a sua importância, a seção 2.7.2.2.2 é dedicada a explicar em detalhe o processo de atualização.

2.7.2.2.2 Atualização do Conjunto de Treinamento:

A atualização do modelo com uma nova sequência de nucleotídeos, S_{new} , é feito em dois passos:

1. Adição da nova sequência:

- Verificar se S_{new} não possui caracteres indeterminados e se tem comprimento compatível com o dataset. Em caso positivo, continue.
- Alinhar S_{new} com $S_{n,ref}$ e validar o alinhamento (se foi completo, ou seja, se o início e o fim de $S_{new}^{aligned}$ coincidem com o de $S_{n,ref}$).
- Verificar se $S_{new}^{aligned}$ é uma nova sequência, ou seja, se já não existe uma idêntica em \mathcal{S}_n . Se for redundante, finalizar o processo. Em caso contrário, adicionar $S_{new}^{aligned}$ ao dataset \mathcal{S}_n e continuar o pipeline.
- Traduzir $S_{new}^{aligned}$ para sequência de códons: $S_{c,new} = [c_1, c_2, \dots, c_M]$ e adicioná-la ao dataset \mathcal{S}_c :

Listing 2.2 – Adding new sequence to the dataset

```
Sc.append(Sc_new)
N = len(Sc) # update dataset size
```

- Construir o vetor de atributos extraindo os códons das posições polimórficas, da forma descrita acima.

Listing 2.3 – Updating training set for CLOPE

```
t_new = [] # Init the attribute list of the new sequence
for p in pos: # loop over polymorphic positions
    t_new.append(str(p) + ":" + str(Sc[-1,p])) # Updating the
                                                attribute list
```

- Adicionar o vetor de atributos à matriz de treinamento T :

Listing 2.4 – Add new feature vector to the training set

```
T.append(t_new) # Updating the training set
```

2. Atualização da Estrutura do Dataset:

- Verificar se a nova sequência criou novas posições polimórficas: Para isto usamos o algoritmo a seguir:

Listing 2.5 – Finding new polymorphic positions

```
pos_new = [] # Initializing list of polymorphic positions
for j in range(M): # Loop through positions
```

```

if len(set(Sc[:,j])) > 1: # if there is more than one codon
    type
        pos_new.append(j) # add position j to the list
# Finding positions in pos_new that are not in pos
new_positions = [p for p in pos_new if p not in pos]

```

Se a lista *new_positions* está vazia finalize o pipeline. Em caso contrário, continue

- b) Atualização do vetor de posições polimórficas:

Listing 2.6 – Updating list of polymorphic positions

```

# Merging the lists while maintaining order
pos = sorted(pos + new_positions)

```

- c) Recriar o dataset de treinamento de CLOPE incluindo as novas posições polimórficas:

Listing 2.7 – Updating training set for CLOPE

```

T = [] # Initializing the training dataset (list of lists of
       attributes)
for i in range(N): # Loop through sequences
    t = [] # Initializing the attribute list for sequence i
    for p in pos: # loop over polymorphic positions
        t.append(str(p) + ":" + str(Sc[i,p])) # Updating the
                                         attribute list of sequence i
    T.append(t) # Updating the training set

```

3. **Salvar o novo modelo:** Em caso de atualização do modelo, salvar e registrar a data da atualização.

2.7.3 Processo de Treinamento

Dado que os conjuntos de treinamento são coleções de amostras retiradas de um universo maior e que muitas dessas amostras possuem semelhanças que permitem agrupá-las de acordo com certos critérios qualitativos/quantitativos definidos, surge a necessidade de dividir as amostras do conjunto de treinamento em classes ou categorias. Além disso, é necessário replicar o mecanismo utilizado para essa divisão, com o objetivo de atribuir novas amostras à classe mais adequada.

Neste contexto, o treinamento consiste na construção de um mecanismo de divisão e de associação de amostras em classes ou categorias distintas. Esse mecanismo deve ser capaz de identificar e separar as amostras com base em suas características compartilhadas, utilizando critérios quantitativos previamente definidos. Após a construção desse mecanismo, ele pode ser utilizado para classificar novas amostras, assegurando que sejam atribuídas à classe ou categoria mais apropriada com base em suas características. Dessa forma, o modelo de treinamento não só organiza as amostras existentes de maneira eficiente,

mas também oferece uma metodologia robusta para a classificação de futuras amostras, garantindo a consistência e precisão na categorização.

A pesar de termos abordado na seção 2.5.2 a diferenciação entre o Treinamento Supervisionado e Não-Supervisionado, a seguir faremos uma descrição mais detalhada e uma reclassificação dos métodos de treinamento que é essencial para o entendimento do trabalho como um todo. Em particular, segundo (Chapelle; Schölkopf; Zien, 2006) vamos diferenciar 4 tipos de treinamento:

1. **Treinamento Supervisionado:** No treinamento supervisionado, o modelo é construído com base em um conjunto de dados que inclui amostras e suas respectivas etiquetas. O objetivo é que o modelo aprenda a reproduzir essas etiquetas da forma mais fiel possível. Isso é alcançado ajustando os parâmetros do modelo de modo que ele possa prever corretamente a classe ou categoria de novas amostras, baseando-se nas características observadas nas amostras etiquetadas durante o treinamento.

O treinamento supervisionado é amplamente utilizado em bioinformática quando há disponibilidade de dados anotados. Alguns exemplos incluem:

- **Classificação de Doenças:**

> Aplicação: Identificação de subtipos de câncer a partir de perfis de expressão gênica.

> Descrição: Modelos supervisionados são treinados com dados de expressão gênica onde as amostras são etiquetadas com o subtipo de câncer correspondente. O modelo aprende a prever o subtipo de câncer para novas amostras com base em seus perfis de expressão gênica.

- **Previsão de Estrutura Proteica:**

> Aplicação: Previsão de estruturas secundárias de proteínas a partir de sequências de aminoácidos.

> Descrição: Utiliza-se um conjunto de dados de proteínas com sequências conhecidas e suas estruturas secundárias anotadas para treinar o modelo a prever a estrutura de novas sequências.

- **Identificação de Genes:**

> Aplicação: Identificação de genes de interesse em estudos de associação genômica ampla (GWAS).

> Descrição: Modelos são treinados para prever a presença ou ausência de genes associados a doenças com base em variáveis genéticas conhecidas e associadas a essas doenças.

2. **Treinamento Não-Supervisionado:** No treinamento não-supervisionado, não são fornecidas etiquetas para as amostras. Em vez disso, o modelo busca identificar padrões,

estruturas ou agrupamentos dentro dos dados com base em características intrínsecas das amostras. O objetivo é descobrir relações ou agrupamentos naturais dentro do conjunto de dados, sem qualquer orientação externa. Um exemplo comum de treinamento não-supervisionado é a análise de clusters, onde o modelo agrupa amostras semelhantes sem conhecer previamente suas categorias.

O treinamento não-supervisionado é útil em bioinformática quando se lida com dados não anotados ou quando se deseja descobrir padrões desconhecidos. Exemplos incluem:

- **Clusterização de Dados de Expressão Gênica:**

> Aplicação: Descoberta de novos grupos de genes co-expresos ou de novos tipos de células.

> Descrição: Algoritmos de clusterização, como k-means ou análise de componentes principais (PCA), são usados para agrupar genes ou células com perfis de expressão similares sem usar etiquetas.

- **Análise de Variação Genética:**

> Aplicação: Identificação de subpopulações dentro de uma população com base em variações genéticas.

> Descrição: Técnicas como análise de componentes principais (PCA) ou t-SNE são usadas para reduzir a dimensionalidade dos dados e visualizar a estrutura da população sem etiquetas prévias.

- **Descoberta de Motivos de Sequência:**

> Aplicação: Identificação de motivos de DNA ou RNA que são conservados em um conjunto de sequências.

> Descrição: Algoritmos como MEME (Multiple EM for Motif Elicitation) são usados para encontrar padrões de sequências conservadas sem a necessidade de etiquetas.

3. **Treinamento Auto-Supervisionado (Self-Supervised):** O treinamento auto-supervisionado é um tipo de treinamento não-supervisionado onde o modelo gera suas próprias etiquetas com base em partes do dado não etiquetado. Nesse tipo de treinamento, o modelo pode usar informações internas dos dados para criar pseudo-etiquetas, que são então usadas para treinar o modelo. Este método permite que o modelo aprenda de maneira supervisionada usando dados não etiquetados.

O treinamento auto-supervisionado é vantajoso quando há uma grande quantidade de dados não anotados e se deseja aproveitar esses dados para treinar modelos. Exemplos incluem:

- **Previsão de Interações Proteína-Proteína:**

> Aplicação: Prever interações entre proteínas a partir de sequências de aminoácidos.

> Descrição: Modelos auto-supervisionados podem ser treinados para prever partes faltantes de sequências de proteínas ou para gerar representações de proteínas que são usadas posteriormente para prever interações.

- **Análise de Sequências Genômicas:**

> Aplicação: Prever elementos funcionais no genoma, como *enhancers* ou *silencers*.

> Descrição: Modelos auto-supervisionados são treinados com tarefas preditivas, como a predição de segmentos faltantes de sequências genômicas, que ajudam a modelar a estrutura e função do DNA sem etiquetas explícitas.

- **Representações de Dados Ómicos:**

> Aplicação: Aprender representações de dados multi-ômeicos (genômica, transcriptômica, proteômica) que podem ser usadas em várias tarefas subsequentes.

> Descrição: Modelos auto-supervisionados são usados para aprender embeddings que capturam informações relevantes dos dados ómicos, que podem então ser aplicados em tarefas supervisionadas ou não supervisionadas.

4. **Treinamento Semi-Supervisionado:** O treinamento semi-supervisionado ou não-supervisionado guiado, é uma abordagem híbrida. Embora as etiquetas das amostras não sejam usadas diretamente para treinar o modelo, elas são utilizadas para avaliar a qualidade dos agrupamentos ou clusters formados pelo modelo. A pureza dos clusters, que mede o grau de homogeneidade dos clusters em relação às etiquetas reais, é uma métrica importante nesse contexto. Essa avaliação é então usada para otimizar os parâmetros do modelo, orientando o processo de treinamento para melhorar a formação dos clusters. Esse tipo de treinamento busca um equilíbrio entre a descoberta de padrões intrínsecos nos dados e a utilização de informações de etiquetas para refinar esses padrões.

O treinamento semi-supervisionado é útil quando há uma combinação de dados anotados e não anotados. Ele permite que o modelo aproveite as etiquetas disponíveis para guiar o aprendizado e, ao mesmo tempo, utilize o grande volume de dados não anotados para melhorar a performance. Alguns exemplos incluem:

- **Anotação Funcional de Genes:**

> Aplicação: Anotação de funções gênicas em novos genomas.

> Descrição: Utiliza um pequeno conjunto de genes anotados com funções conhecidas e um grande conjunto de genes não anotados. O modelo aprende a partir dos genes anotados e generaliza esse conhecimento para prever as funções

dos genes não anotados, utilizando informações adicionais dos dados não anotados para melhorar a precisão.

- **Identificação de Regiões Funcionais no Genoma:**

> Aplicação: Descoberta de *enhancers* e outras regiões regulatórias.

> Descrição: Usa um conjunto de regiões genômicas com funções conhecidas (etiquetadas) e um grande conjunto de regiões não caracterizadas (não etiquetadas).

O modelo aprende a partir das regiões conhecidas e utiliza esse conhecimento para prever as funções das regiões não caracterizadas, aproveitando os dados não anotados para melhorar a robustez das predições.

Essas distinções são fundamentais para entender como diferentes abordagens de treinamento podem ser aplicadas em cenários variados, cada uma com suas próprias vantagens e desafios.

A abordagem adotada neste projeto é o Treinamento Semi-Supervisionado.

Nas seções 2.7.3.1 e 2.7.4 a seguir, descrevemos os processos de treinamento do método CLOPE e de classificação de uma nova sequência usando o modelo treinado, respectivamente.

2.7.3.1 *Treinamento do Classificador CLOPE*

CLOPE (*Clustering with sLOPE*) é um algoritmo de clusterização projetado especificamente para dados categóricos. Ele se destaca por sua simplicidade e eficiência em termos de custo computacional e é particularmente útil quando se trabalha com grandes conjuntos de dados categóricos.

2.7.3.2 *Características Principais:*

- **Entrada:**

> Conjunto de transações, onde cada transação é composta por uma lista não ordenada e de tamanho variável de "itens" definidos nominalmente, por exemplo "manga", "carro", "coca-cola 1.5l". No nosso caso, como os itens são códons, temos apenas 60 e poucos deles, mas, como é importante a posição deles, criamos é uma lista de P itens categóricos formados por pares (*posição* : códon), onde a *posição* está ordenada de menor a maior². Ou seja nossas transações são da forma $t = [(p : A_p), p = 1, 2, \dots, P] \in T$, representando uma sequência do conjunto de treinamento, onde A_p é o numeral do códon na posição polimórfica p nessa sequência³.

² por conveniência para a utilização posterior do modelo com sequências parciais - fora do escopo deste estudo

³ A construção do conjunto de treinamento T foi descrita na seção 2.7.2

> Parâmetro de repulsão, r , que controla o compromisso entre a compactação intra-cluster e a separação inter-cluster. É crucial para determinar a formação dos clusters. Valores baixos de r favorecem clusters mais compactos, enquanto valores altos de r favorecem clusters mais separados.

- **Objetivo:** Maximizar uma função de lucro que considera tanto a compactação dos clusters quanto a separação entre eles.

2.7.3.3 *Algoritmo:*

1. Inicialização: Começa com uma distribuição inicial aleatória das transações no conjunto T .
2. Alocação Inicial: As transações são lidas na ordem sendo atribuídas a algum dos clusters existentes ou a um cluster novo, dependendo de qual opção agrupa mais lucro ao agrupamento.
3. Iterações: As transações são iterativamente movidas entre clusters para maximizar a função de lucro. Em cada iteração todas as transações, sorteadas de forma aleatória, são testadas. O teste consiste em que cada transação é avaliada sendo transferida para todos os outros clusters, assim como criando um cluster novo com ela. As transações são movimentadas para a alternativa que agrupa mais valor, podendo ficar no mesmo cluster se for a melhor opção. A função de lucro é projetada para equilibrar a densidade dos clusters e a diversidade dentro de cada cluster. As iterações finalizam quando nenhuma transação é transferida de um cluster para outro.

Para poder executar os passos 2 e 3 do algoritmo acima, se utilizam três funções de avaliação de valor: (1) Função de Valor do Cluster, (2) Função de Mudança de Valor do Cluster com a Incorporação de uma Transação e (3) Função de Valor da Criação de um Cluster. Estas funções são descritas a seguir.

2.7.3.3.1 **Função de Valor do Cluster**

Dado um cluster C_k que contém N_k transações $[t_1, t_2, \dots, t_{N_k}]$, a função de valor do mesmo é definida como o produto do gradiente (C_k) pelo peso w_k do cluster:

$$V(C_k) = \text{grad}(C_k) w_k \quad (2.2)$$

onde

$$\begin{aligned} \text{grad}(C_k) &= \frac{S_k}{W_k^r} \\ w_k &= \frac{N_k}{N} \end{aligned}$$

sendo:

- $N = \sum_{k=1}^K N_k$: o numero total de transações no grupamento com K clusters
- $S_k = \sum_{n=1}^{N_k} |t_n|$: a soma dos tamanhos das transações no cluster C_k . Como no nosso caso todas as transações tem o mesmo tamanho P (número de sítios polimórficos), então $S_k = PN_k$. Ou seja, S_k é uma medida do tamanho do cluster e como S_k está no numerador da equação do valor, significa que clusters maiores tem mais valor que clusters menores.
- W_k : o número de itens distintos no cluster. No nosso caso W_k é o número de pares $(p : Ap)$ distintos nas N_k transações alocadas no cluster C_k . Lembrando que um par $(p : A_p)$ representa uma posição polimórfica e um códon, W_k é uma medida da variabilidade genética intra-cluster. Como W_k está no denominador da equação do valor, significa que clusters com menor variabilidade tem maior valor que clusters com maior variabilidade genética.
- $r \geq 1$: o parâmetro de repulsão. É o parâmetro a ser otimizado utilizando as etiquetas para avaliar critérios quantitativos de qualidade do agrupamento. Ou seja, as etiquetas não influenciam o agrupamento em si, por isso que nosso método é classificado como semi-supervisionado. As etiquetas são utilizadas para otimizar o parâmetro de repulsão.

Fazendo as substituições indicadas, e eliminando a constante P , a função de valor do cluster C_k vem dado de forma simplificada por:

$$V(C_k) = \frac{N_k^2}{W_k^r} \quad (2.3)$$

2.7.3.3.2 Mudança no valor do cluster com a adição de uma transação

Para adicionar uma transação t_i a um cluster existente C_k , a mudança no valor do cluster é calculada como:

$$\Delta V_{\text{add}}(t_i, C_k) = \frac{(N_k + 1)^2}{(W_k + \omega_{i,k})^r} - V(C_k) \quad (2.4)$$

onde $\omega_{i,k}$ é o número de novos itens que a transação t_i trás com ela para o cluster k e $V(C_k)$ é o valor atual do cluster C_k definida pela equação 2.3.

2.7.3.3.3 Mudança no valor do cluster com a exclusão de uma transação

Ao excluir uma transação t_i de um cluster existente C_k , a mudança no valor do cluster é calculada como:

$$\Delta V_{\text{remove}}(t_i, C_k) = \frac{(N_k - 1)^2}{(W_k - \delta_{i,k})^r} - V(C_k) \quad (2.5)$$

onde $\delta_{i,k}$ é o número de itens únicos da transação t_i no cluster k e $V(C_k)$ é o valor atual do cluster C_k definida pela equação 2.3.

2.7.3.3.4 Valor da criação de um novo cluster com uma transação nova

Durante a inicialização do agrupamento, as novas sequências são adicionadas a clusters existentes ou a um cluster novo. Neste contexto é necessário definir o valor da criação de um cluster novo, para comparar com o valor da adição da transação aos clusters existentes.

Ao criar um novo cluster C_{new} com uma nova transação t_i , estamos criando um cluster com uma única transação, pelo que $N_{new} = 1$ e qualquer transação no nosso caso tem P itens distintos, pelo que o valor da criação do novo cluster é:

$$\Delta V(C_{new}(t_i)) = \frac{N_{new}}{|t_i|^r} = \frac{1}{P^r} \quad (2.6)$$

2.7.3.3.5 Decidindo a movimentação de uma transação de um cluster para outro

Considere uma transação t_i num cluster C_{k1} que possa ser movimentada para outro cluster C_{k2} . Neste caso precisamos calcular a mudança total de valor derivada da remoção de um e da adição no outro, ou seja:

$$\Delta V_{\text{move}}(t_i, C_{k1}, C_{k2}) = \Delta V_{\text{remove}}(t_i, C_{k1}) + \Delta V_{\text{add}}(t_i, C_{k2}) \quad (2.7)$$

Fazendo as substituições indicadas obtemos:

$$\Delta V_{\text{move}}(t_i, C_{k1}, C_{k2}) = \frac{(N_{k1} - 1)^2}{(W_{k1} - \delta_{i,k1})^r} - V(C_{k1}) + \frac{(N_{k2} + 1)^2}{(W_{k2} + \omega_{i,k2})^r} - V(C_{k2}) \quad (2.8)$$

Como $\Delta V_{\text{remove}}(t_i, C_{k1})$ é definido para a sequência t_i a ser movimentada, a busca tem que ser feita calculando apenas o ganho $\Delta V_{\text{add}}(t_i, C_{k2})$ nos outros clusters, para achar o maior ganho possível.

Contudo a decisão requer que $\Delta V_{\text{move}}(t_i, C_{k1}, C_{k2})$ seja positiva, pelo que se

$$\Delta V_{\text{remove}}(t_i, C_{k1}) > 0$$

então a transferência é realizada, mas, se $\Delta V_{\text{remove}}(t_i, C_{k1}) < 0$ então somente se realiza a transferência caso o máximo ganho com a adição a outro cluster for superior à perda com a remoção do cluster atual. Em caso contrário a transação t_i permanece no cluster $k1$.

2.7.3.3.6 Decidindo a remoção de uma transição de um cluster para criar um novo

Considere uma transação t_i num cluster C_k que possa ser removida para criar um novo cluster C_{new} . Neste caso precisamos calcular o ganho total de forma

$$\Delta V_{\text{rem-new}}(t_i, C_k, C_{new}) = \Delta V_{\text{remove}}(t_i, C_k) + \Delta V(C_{new}(t_i)) \quad (2.9)$$

Fazendo as substituições indicadas obtemos:

$$\Delta V_{\text{rem-new}}(t_i, C_k, C_{new}) = \frac{(N_{k1} - 1)^2}{(W_{k1} - \delta_{i,k1})^r} - V(C_{k1}) + \frac{1}{P^r} \quad (2.10)$$

Fazendo os cálculos se

$$\Delta V_{\text{rem-new}}(t_i, C_k, C_{new}) \leq 0$$

a transação t_i permanece no cluster C_k . Em caso contrário, t_i é removida do cluster C_k e um novo cluster é criado com ela.

2.7.3.4 Otimização:

O treinamento do classificador CLOPE consiste em achar uma estrutura de clusters ótima, segundo critérios de qualidade estabelecidos e avaliados com base em etiquetas associadas a cada amostra.

Nas seções seguintes 2.7.3.4.1, 2.7.3.4.2 e ?? descrevemos a métrica usada para avaliar a qualidade de um agrupamento, o processo iterativo para gerar distintos agrupamentos com o mesmo parâmetro de repulsão r , e o processo de busca do parâmetro de repulsão ótimo, respectivamente.

2.7.3.4.1 Métrica para avaliar qualidade do agrupamento baseado em etiquetas:

Para definir o que é uma boa estrutura de clusters precisamos introduzir métricas para quantificar critérios de qualidade e poder escolher a melhor estrutura entre todas as analisadas.

Existem métricas internas (que não utilizam etiquetas) e externas (baseadas em etiquetas) como foram descritas na seção 2.5.3.

No projeto focamos apenas a pureza dos clusters. Um cluster totalmente puro contém sequências de uma única etiqueta, ou seja, não mistura genótipos no nosso caso. O problema é que existe uma relação de compromisso entre pureza e número de clusters. Para entender isto, considere que cada sequência seja alocada a um cluster, ou seja, que os clusters contenham apenas uma sequência. Neste caso todos os clusters seriam totalmente puros, mas teríamos muitos clusters. Na medida em que o número de clusters se reduz aumenta a probabilidade do surgimento de clusters que misturam etiquetas.

Na nossa aplicação não é problema se se formam vários clusters puros com a mesma etiqueta. Isto simplesmente refletiria que existe uma diversidade sensível nas amostras com essa etiqueta, ou seja que existem subgrupos dentro dessa categoria etiquetada.

Para avaliar a pureza considere que no conjunto de treinamento T as sequências são classificadas em G genótipos distintos e etiquetadas com $g = 1, 2, \dots, G$. Considere que CLOPE identificou $K > 1$ clusters e que alocou em cada um deles um conjunto de N_k transações $T_k = \{t_{k,1}, t_{k,2}, \dots, t_{k,N_k}\}$, nos clusters $k = 1, 2, \dots, K$.

Agora, se denotamos por $d_{k,g}$ o número de transações no cluster k , ou seja, em T_k , que estão anotadas com o genótipo g , considerando os K clusters e os G genótipos, podemos preencher uma Matriz de Distribuição

$$D = \begin{bmatrix} d_{1,1} & d_{1,2} & \cdots & d_{1,G} \\ d_{2,1} & d_{2,2} & \cdots & d_{2,G} \\ \vdots & \vdots & \ddots & \vdots \\ d_{K,1} & d_{K,2} & \cdots & d_{K,G} \end{bmatrix} \quad (2.11)$$

onde k indica uma linha (cluster) e g uma coluna (genótipo) de D .

De acordo com isto, a pureza do cluster k pode ser avaliada como:

$$p_k = \begin{cases} \frac{\max_g(d_{k,g})}{N_k} \in [0, 1] & \text{para } N_k > 0 \\ 0 & \text{em caso de cluster vazio} \end{cases} \quad (2.12)$$

Note, que no caso que o cluster k seja puro, ou seja, que tenha apenas transações de um único genótipo $g*$, se cumpre que $\max_g(d_{k,g}) = d_{k,g*} = N_k$ pelo que $p_k = 1$.

Podemos medir a falta de pureza absoluta do cluster k como

$$h_k = N_k - \max_g(d_{k,g}) \geq 0$$

e a falta total de pureza do agrupamento como

$$H = \sum_k h_k \geq 0$$

pelo que a pureza do agrupamento pode ser estimada como

$$\mathcal{P} = 1 - \frac{H}{N} \in [0, 1] \quad (2.13)$$

onde $N = N_1 + N_2 + \dots + N_K$ é o número total de transações em T .

Baseado nas métricas definidas em 2.12 e 2.13, introduzimos uma medida de qualidade do agrupamento da forma:

$$Q = \left[p_k^{\min} \ p_k^{\text{avg}} \ \mathcal{P} \right]^{1/3} \quad (2.14)$$

onde $p_k^{\min}, p_k^{\text{avg}}$ são a pureza mínima e média dos clusters.

A métrica combinada Q mede a extensão em que os clusters contêm uma única classe de pontos de dados. Alta Q indica que os clusters são compostos predominantemente por uma única classe, tornando-os mais homogêneos.

2.7.3.4.2 Processo repetitivo buscando a melhor solução com repulsão dada:

O algoritmo de treinamento realiza um número prefixado R de repetições do agrupamento utilizando o algoritmo CLOPE. Em cada repetição calcula a métrica de qualidade Q e armazena a maior métrica Q_{best} e o melhor agrupamento.

2.7.3.4.3 Processo de busca da repulsão ótima:

O algoritmo de treinamento varia o parâmetro de repulsão r a ser otimizado, começando num valor mínimo r_{min} e aumentando com passo Δ_r até atingir um valor máximo r_{max} prefixado.

A experiência demomstrou que a qualidade dos agrupamentos Q_{best} começa a aumentar com o aumento de r , até que começa a diminuir de forma permanente. Por este motivo, o algoritmo de busca do r ótimo é interrompido a primeira vez que Q_{best} decresce, retornando como valor ótimo o valor de repulsão anterior à descida, ou seja, para o qual foi obtido o máximo Q_{best} .

2.7.3.5 Resultado do Treinamento:

Quando o algoritmo de otimização do parâmetro de repulsion r do método CLOPE finaliza, ele retorna o valor ótimo r_{optm} e também o agrupamento de maior qualidade Q_{best} . O agrupamento consiste numa lista que contém o número k do cluster ao qual cada

transação (sequência) no conjunto de treinamento T foi alocado. Denotemos o agrupamento por

$$\mathcal{G} = [k_1, k_2, \dots, k_N]$$

onde $k_i \in [1, K]$ representa o cluster ao qual a transação (sequência) i foi alocada.

Junto com o agrupamento, o método retorna a matriz de distribuição D desse agrupamento definida em 2.11.

No final do treinamento \mathcal{G} e D são adicionados ao modelo de referência descrito na seção 2.7.2.2.1.

2.7.4 Pipeline de Classificação de Sequências

Nesta seção descrevemos brevemente o pipeline para classificar uma sequência de entrada usando o modelo CLOPE treinado.

Na seção 2.7.2.2.2 foi descrito o processo para adicionar uma nova sequência ao conjunto de treinamento de CLOPE. O processamento inicial para classificar uma sequência de entrada S_{input} é muito similar. A seguir os passos do processamento:

1. Verificar se S_{input} não possui caracteres indeterminados e se tem comprimento compatível com o dataset. Em caso positivo, continue.
2. Alinhar S_{input} com $S_{n,ref}$ e validar o alinhamento (se foi completo, ou seja, se o início e o fim de $S_{input}^{aligned}$ coincidem com o de $S_{n,ref}$).
3. Classificação Direta: Verificar se no conjunto de treinamento \mathcal{S}_n existe uma sequência idêntica a $S_{input}^{aligned}$. Se existir, retornar a distribuição de probabilidade genotípica do cluster CLOPE ao qual a sequência idêntica do dataset foi alocada. Em caso contrário, continue.

A distribuição de probabilidade genotípica vem dada pelo percentual de cada genótipo no cluster identificado. Sendo k o cluster ao qual pertence a sequência do dataset que é idêntica à sequência de entrada, a probabilidade da sequência de entrada pertencer a qualquer genótipo $g = 1, 2, \dots, G$ vem dada por $d_{k,g}/N_k$, onde $N_k = \sum_g d_{k,g}$.

4. Traduzir $S_{input}^{aligned}$ para sequência de códons: $S_{c,input} = [c_1, c_2, \dots, c_M]$.
5. Construir o vetor de atributos extraiendo os códons das posições polimórficas:

Listing 2.8 – Getting feature vector for CLOPE

```
t_input = [] # Init the attribute list of the new sequence
for p in pos: # loop over polymorphic positions
    t_input.append(str(p) + ":" + str(Sc_input[-1,p])) # Updating
        the attribute list
```

6. Buscar o cluster CLOPE mais adequado para a sequência de entrada:

- a) Inicialização do CLOPE: Distribuir as transações nos clusters segundo o melhor agrupamento do treinamento \mathcal{G} .
- b) Alocação: Atribuir a transação de entrada a algum dos clusters identificados no treinamento ou a um cluster novo, dependendo de qual opção agrupa mais lucro ao agrupamento. Diferenciamos dois casos:

> *A sequência de entrada é atribuída a um cluster existente:* Retornar a distribuição de probabilidade genotípica do cluster CLOPE atribuído à sequência de entrada.

> *A sequência de entrada é atribuída a um novo cluster:* Neste caso se retorna que a sequência parece ser de um novo genótipo. Este caso, que evidencia a capacidade natural de descoberta de novas cepas virais do método descrito, precisa ser analisado por especialistas, para determinar se a sequência lida é confiável ou pode ser de baixa qualidade. Em caso de ser considerada de alta qualidade (confiável) pelos especialistas, o processo padrão recomendado, consiste em:

- Notificar os órgãos de vigilância sanitária sobre a descoberta
- Nomear o novo genótipo
- Adicionar a nova sequência ao conjunto de treinamento como descrito na seção 2.7.2.2.2
- Atualizar a Matriz de Distribuição D e o vetor de agrupamento \mathcal{G} com o novo cluster e novo genótipo.

2.8 Estrutura e Funcionamento de Ferramentas de Bioinformática na WEB

2.8.1 Visão Geral

Um sistema bioinformático web genérico deve ser robusto, escalável e capaz de lidar com grandes volumes de dados e processamentos demorados. A estrutura deve contemplar:

- **Gerenciamento de Usuários e Filas:**
 - Controle de acesso.
 - Priorização de tarefas.
 - Notificações de status e conclusão.
- **Acesso a Dados:**
 - Integração com bancos de dados locais e remotos.

- Garantia de segurança e eficiência no acesso aos dados.
- **Camadas de Processamento:**
 - Backend poderoso para cálculos pesados.
 - Camada intermediária para gerenciamento de filas e comunicação.
 - Frontend amigável para interação com o usuário.
- **Comunicação Robusta:**
 - Comunicação assíncrona entre frontend e backend para lidar com tarefas longas.
- **Notificação de Resultados:**
 - Envio de emails com links para resultados ou anexos com os dados finais.

2.8.2 Tecnologias Mais Utilizadas

- **Linguagens de Programação:**

Backend: Python, R, Java, C++.
Frontend: JavaScript (React, Vue.js, Angular).
- **Frameworks:**

Backend: Django (Python), Spring Boot (Java).
Frontend: Next.js (React), Nuxt.js (Vue.js), Angular Material (Angular).
- **Gerenciamento de Filas:**

Celery (Python). RabbitMQ (mensagens).
- **Bancos de Dados:**

Locais: PostgreSQL, MySQL.
Remotos: MongoDB, NoSQL.
- **Armazenamento em Nuvem:**

Amazon S3, Google Cloud Storage.
- **Comunicação:**

REST APIs. WebSockets.
- **Ferramentas de Autenticação:**

OAuth, JWT.

- **Monitoramento:**

Prometheus, Grafana.

- **Softwares para Alinhamento de Sequências:**

BLAST. Clustal Omega. SAMtools. GATK.

Na figura 5 mostramos um diagrama da estrutura genérica deste tipo de sistemas.

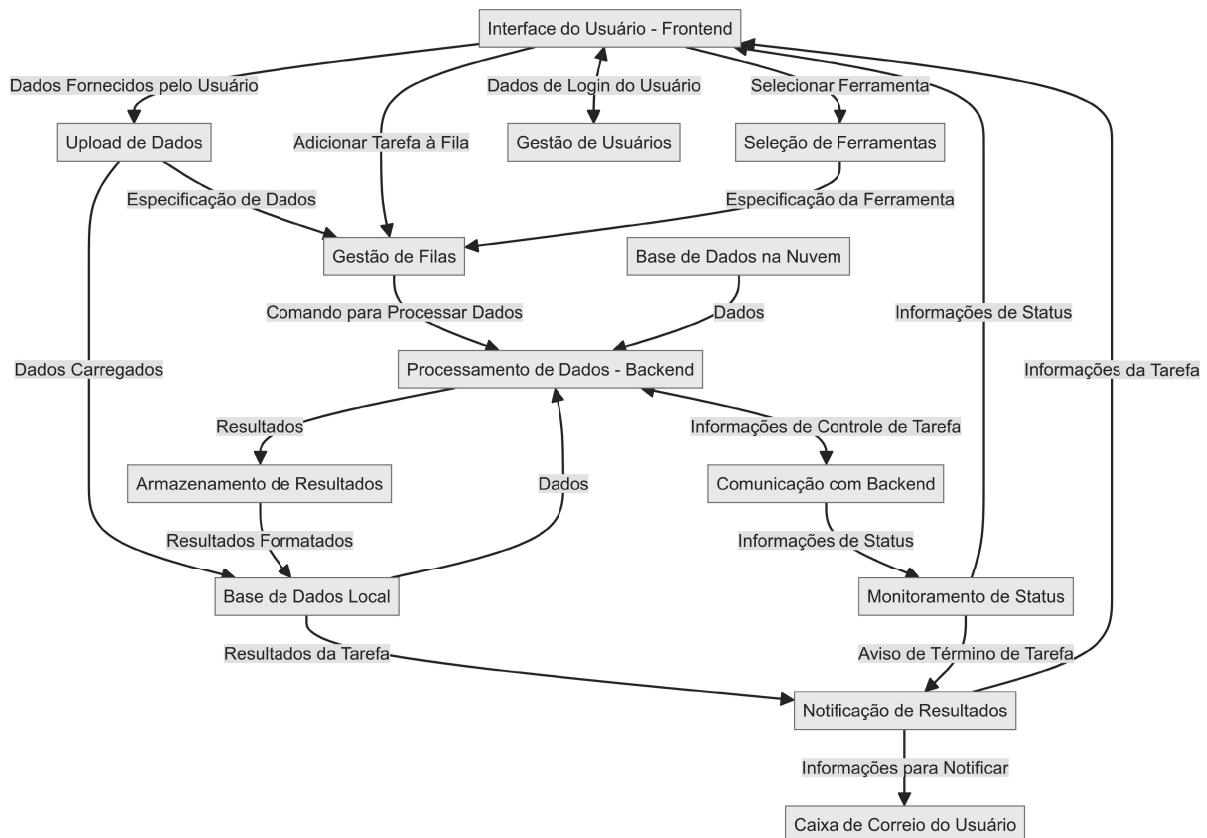


Figura 5 – Diagrama Estrutural e Funcional de um Sistema web para Bioinformática