UNIVERSIDADE DO ESTADO DA BAHIA

DEPARTAMENTO DE CIÊNCIAS EXATAS E DA TERRA

CURSO DE GRADUAÇÃO EM SISTEMAS DE INFORMAÇÃO

CÂNDIDO LUIZ DO NASCIMENTO JÚNIOR

ANÁLISE DA PROTEÍNA SPIKE DO SARS-COV-2 UTILIZANDO O ALGORITMO CBUC

SALVADOR

2021

CÂNDIDO LUIZ DO NASCIMENTO JÚNIOR

ANÁLISE DA PROTEÍNA SPIKE DO SARS-COV-2 UTILIZANDO O ALGORITMO CBUC

Monografia apresentada ao curso de Sistemas de Informação do Departamento de Ciências Exatas e da Terra da Universidade do Estado da Bahia - UNEB, como requisito à obtenção do grau de bacharel em Sistemas de Informação. Área de Concentração: Ciência da Computação

Orientadora: Profa. Dra. Maria Inés Valderrama Restovic

Coorientador: Prof. Dr. Diego Gervasio Frías Suárez

SALVADOR

2021

FICHA CATALOGRÁFICA
Sistema de Bibliotecas da UNEB

N244a Nascimento Júnior, Cândido Luiz do

Análise da proteína spike do Sars-Cov-2 Utilizando o algoritmo CBUC / Cândido Luiz do Nascimento Júnior. - Salvador, 2021.
59 fls.

Orientador(a): Profa. Dra. Maria Inés Valderrama Restovic.
Coorientador(a): Prof. Dr. Diego Gervasio Frías Suárez.
Inclui Referências
TCC (Graduação - Sistemas de Informação) - Universidade do Estado da Bahia. Departamento de Ciências Exatas e da Terra. Campus I. 2021.

1.Bioinformática. 2.SARS-CoV-2. 3.Filogenia. 4.Árvore Filogenética. 5.Codon Based Unsupervised Classification.

CDD: 004

CÂNDIDO LUIZ DO NASCIMENTO JÚNIOR

ANÁLISE DA PROTEÍNA SPIKE DO SARS-COV-2 UTILIZANDO O ALGORITMO CBUC

Monografia apresentada ao curso de Sistemas de Informação do Departamento de Ciências Exatas e da Terra da Universidade do Estado da Bahia - UNEB, como requisito à obtenção do grau de bacharel em Sistemas de Informação. Área de Concentração: Ciência da Computação

Aprovada em:

BANCA EXAMINADORA

Profa. Dra. Maria Inés Valderrama Restovic (Orientadora)
Universidade do Estado da Bahia – UNEB

Prof. Dr. Diego Gervasio Frías Suárez (Coorientador)
Universidade do Estado da Bahia – UNEB

Prof. Dr. Ernesto de Souza Massa Neto
Universidade do Estado da Bahia – UNEB

Prof. Dr. Alexandre Lenz
Universidade do Estado da Bahia – UNEB

AGRADECIMENTOS

Agradeço aos meus pais por todo apoio não somente durante esta graduação, mas em todos momentos da minha vida.

Agradeço aos meus colegas de curso Renato Andrade, José Cleiton, Matheus Tanure, Pedro Rosário, Alex Freire, Thiago Armede, Henrique Pacheco, Diego dos Santos e Yan Gabriel e Cleber del Rei, irmãos de batalha, que estiveram comigo durante essa longa jornada onde compartilhamos conhecimento e nos ajudamos diante das dificuldades enfrentadas.

Agradeço aos grandes mestres e doutores que compartilharam comigo seus conhecimentos durante a graduação, conhecimentos valiosos que complementam quem eu sou hoje.

Em especial, agradeço a professora Maria Inês pela oportunidade de trabalhar com este projeto e pela incrível orientação ao logo desta pesquisa. Não só pela disposição de orientar, como também de motivar e acreditar no meu potencial.

Por fim, gostaria de agradecer ao grande mestre Jorge Farias, com sua didática que nos tirava da zona de conforto e nos fazia pensa a fundo sobre o problema. Foram aulas nas quais tive o prazer de assistir, e com certeza tem um papel importante na maneira que eu trabalho hoje.

"Toda elegância é discreta, assim como toda virtude é silenciosa."

(Luiz Felipe Pondé)

RESUMO

As árvores filogenéticas têm um papel importante na biologia moderna porque elas provêm uma maneira concisa de visualizar a evolução dos descendentes partindo de ancestrais comuns. Durante a evolução da linhagem de um organismo os descendentes podem se divergir e "separar", esses eventos são conhecidos como cladogênese, no qual se refere a origem de um novo ramo. Um clado é um pedaço de uma árvore filogenética que contém uma linhagem ancestral e todos os descendentes dessa linhagem. Os clados formados em uma árvore filogenética nos passam uma importante informação sobre os agrupamentos das sequências. O procedimento de classificação feito atualmente pode levar muito tempo. O algoritmo CBUC - Codon Based Unsupervised Classification, inicialmente implementado em Scilab pelo Prof. Dr. Diego Gervasio Frías Suárez, nesse trabalho foi implementado em Python, consegue genotipar as sequências e encontrar agrupamentos. O SARS-CoV-2 - *Severe Acute Respiratory Syndrome Coronavirus* 2, vírus causador da doença COVID-19 - *Coronavirus Disease-2019*, é um vírus altamente transmissível e se espalhou rapidamente pelo mundo, escalando de um surto para uma pandemia. O objetivo geral deste trabalho compreende a análise da proteína *spike* do SARS-CoV-2 utilizando o algoritmo CBUC e confrontar os agrupamentos gerados pelo CBUC com os agrupamentos gerados pelo método *Maximum Likelihood*. Nesse trabalho também foi desenvolvida uma ferramenta para a coleta das sequências que foram analisadas pelas implementações do CBUC. Os resultados da implementação em Python conseguiram encontrar agrupamentos coerentes com a árvore filogenética em um curto período de tempo e indica que a proposta do CBUC é promissora em genotipagem de sequências genéticas.

Palavras-chave: Bioinformática. SARS-CoV-2. Filogenia. Árvore Filogenética. Codon Based Unsupervised Classification.

ABSTRACT

Phylogenetic trees play an important role in modern biology because they provide a concise way to visualize evolution as descendants from common ancestors. During evolution, evolutionary lineages may diverge and “split”, these events are known as cladogenesis, which refers to the origin of a new branch. A clade is a piece of a phylogenetic tree that includes an ancestral lineage and all descendants of that lineage. The clades formed in a phylogenetic tree provide us an important information about the grouped sequences. Current used methods of classification may take a lot of time. The CBUC - Codon Based Unsupervised Classification algorithm, initially implemented in Scilab by Prof. Dr. Diego Gervasio Frías Suárez, in this work, implemented in Python, it manages to genotype sequences and find clusters. SARS-CoV-2 - Severe Acute Respiratory Syndrome Coronavirus 2, the organism that causes COVID-19 - Coronavirus Disease-2019, is a highly transmissible virus that has spread rapidly around the world, escalating from an outbreak to a pandemic. The general objective of this work comprises the analysis of the spike protein of SARS-CoV-2 using the CBUC algorithm and confronting the clusters generated by CBUC with the clusters generated by the Maximum Likelihood method. In this work, a tool was also developed to collect the sequences that were analyzed by the CBUC implementations. The results of the Python implementation were able to find clusters consistent with the phylogenetic tree in a short period of time and indicate that the CBUC proposal is promising in genotyping genetic sequences.

Keywords: Bioinformatics. SARS-CoV-2. Phylogeny. Árvore Filogenética. Phylogenetic Tree. Codon Based Unsupervised Classification.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

API	*Aplication Programming Interface*
CBUC	*Codon Based Unsupervised Classification*
COVID-19	*Coronavirus Disease*-2019
CSV	*Comma-Separeted Values*
DNA	*Deoxyribonucleic Acid*
HTML	*HyperText Markup Language*
ICTV	*International Committee on Taxonomy of Viruses*
MCMC	Markov Chain Monte Carlo
MERS	*Middle East Respiratory Syndrome*
ML	*Maximum Likelihood*
mRNA	*Messenger Ribonucleic Acid*
NCBI	*National Center for Biotechnology Information*
NIH	*National Institutes of Health*
NJ	Neighbor Joining
OMS	Organização Mundial de Saúde
PSRM	*Parametric State Representation Method*
RNA	*Ribonucleic Acid*
SARS	*Severe Acute Respiratory Syndrome*
SARS-CoV-2	*Severe Acute Respiratory Syndrome Coronavirus 2*
tRNA	*Transfer Ribonucleic Acid*

SUMÁRIO

1 INTRODUÇÃO

As árvores filogenéticas têm um papel importante na biologia moderna porque elas provêm uma maneira concisa de visualizar a evolução dos descendentes partindo de ancestrais comuns (BAUM; SMITH., 2013). Uma árvore filogenética retrata a linhagem evolutiva de um conjunto de táxons. Os táxons são tipicamente espécies ou grupo de espécies vivas, mas também podem ser organismos fósseis, organismos individuais, genes ou populações.

Durante a evolução de uma linhagem, os descendentes podem se divergir e "separar", esse evento é conhecidos como cladogênese, no qual se refere a origem de um novo ramo (clado no grego antigo) na árvore filogenética. Um clado é um pedaço de uma árvore filogenética que contém uma linhagem ancestral e todos os descendentes dessa linhagem. Clados tem uma propriedade de monofilia (do grego clã único) e pode ser chamado de grupo monofilético. O clado ou grupo monofilético pode ser identificado como um pedaço de uma árvore filogenética que pode ser cortado fora a partir de um único ponto.

Os clados formados em uma árvore filogenética nos passam uma importante informação dos agrupamentos das sequências, quando quer-se saber a qual grupo monofilético uma sequência específica pertence, coloca-se a sequência em um conjunto de dados com outras sequências da mesma espécie, e verifica-se os agrupamentos formados e em qual agrupamento a sequência ficou. Porém esse procedimento pode levar muito tempo além de ser custoso.

Para contribuir com o processo de tipagem de sequências, tão importante, como por exemplo, a identificação de novas cepas virais, o Prof. Dr. Diego Gervasio Frías Suárez desenvolveu o algoritmo *Codon Based Unsupervised Classification* (CBUC) que é uma adaptação do *Parametric State Representation Method* (PSRM) para um conjunto de dados genômicos. O PSRM foi desenvolvido utilizando a linguagem de programação de alto nível Scilab, na versão 6.0, esse método teve origem no monitoramento de equipamento industrial. E foi originalmente desenvolvido para processar múltiplas séries temporais. E dentre outras coisas o CBUC consegue agrupar as sequências em famílias de forma semelhante ao agrupamento tradicional feito pela árvore filogenética. O professor Diego desenvolveu o CBUC com objetivo de fazer uma tipagem de maneira mais rápida que as ferramentas atuais.

Em dezembro de 2019 a Organização Mundial de Saúde (OMS) foi informada sobre um surto de pneumonia em Wuhan, uma cidade localizada na China. Em 11 de Fevereiro de 2020 a OMS nomeou oficialmente o surto atual de coronavírus como *Coronavirus Disease*-2019 (COVID-19) (SUN et al., 2020) e o *International Committee on Taxonomy of Viruses* (ICTV) nomeou o vírus como *Severe Acute Respiratory Syndrome Coronavirus 2* (SARS-CoV-2) (HU et al., 2021).

Sendo altamente transmissível, a doença COVID-19 se espalhou por todo mundo rapidamente, ultrapassando o número de infectados do *Severe Acute Respiratory Syndrome* (SARS) e do *Middle East Respiratory Syndrome* (MERS) juntos. E com essa disseminação acelerada pelo mundo a OMS em 11 de março de 2020 caracterizou o surto global de COVID-19 como uma pandemia (OMS, 2020).

Desde sua aparição, o coronavírus já sofreu bastante mutações e criou várias variantes. Classificar uma nova sequência de SARS-CoV-2 de forma rápida ajuda a descobrir de qual variante ela faz parte e assim analisar quais as melhores medidas que devem ser tomadas.

Os organismos possuem uma sequência em particular que define quais são as moléculas que compõem as suas proteínas, os aminoácidos. Eles são formados por três bases nitrogenadas chamadas de nucleotídeos, isso observado experimentalmente por CRICK et al. (1961). A sequência inteira que codifica um organismo em particular é chamada de genoma. São denominados códons a sequência de três nucleotídeos que compõem um aminoácido, elemento estrutural de uma proteína. Uma proteína é identificada por vários códon.

Grosjean e Fiers (1982) concluíram em sua pesquisa em organismos unicelulares, como *Escherichia coli*, que o *Messenger Ribonucleic Acid* (mRNA) não só contém informações para especificar uma sequência de aminoácidos em particular, mas também que a sua estrutura determina a eficiência da tradução. E isto é determinado tanto pela frequência de iniciação quanto pela devida escolha dos códons. O uso de códons é marcadamente diferente em genes altamente expressos comparados com genes que codificam proteínas raras. Dessa forma se o vírus quiser se replicar mais eficientemente, ele terá que sintetizar moléculas de mRNA com estruturas e códons aos quais a célula hospedeira produz eficientemente.

Ikemura (1985) em seu trabalho encontrou uma correlação entre a frequência do uso de códons e o conteúdo do *Transfer Ribonucleic Acid* (tRNA) *isoacceptor*, em organismos

unicelulares, como *Escherichia coli*, *Salmonella typhimurium*, e *Saccharomyces cerevisiae*. tRNA *isoacceptor* é um tRNA que carrega o mesmo aminoácido mas geralmente possui diferentes códons para aquele mesmo aminoácido. Para explicar essa correlação, ele analisou a frequência do uso do tRNA nos genes individualmente, e concluiu que nos genes da *Escherichia coli*, a escolha do uso de códons é restringida pela disponibilidade do tRNA.

Os evidentes padrões de similaridade na escolha de códons sinônimos apoiam fortemente a ideia da seleção restritiva imposta pelo conteúdo do tRNA *isoacceptor*. Discutindo um pouco sobre questões evolutivas, o processo celular de síntese de proteínas gasta uma grande quantidade de energia e de massa. Se códons que não são frequentemente utilizados fossem usados em um gene altamente expresso, os ribossomos teriam que executar uma tarefa antieconômica para encontrar o tRNA adequado para o grande número de moléculas de mRNA deste gene. Assim esses organismos ao se multiplicar tendem a utilizar códons mais frequentes, utilizando menos energia, gerando uma diferenciação no seu genoma e criando novas variantes.

Ainda não foram publicadas evidências do funcionamento desse algoritmo, então o objetivo geral deste trabalho compreende a análise da proteína *spike* do SARS-CoV-2 utilizando o algoritmo CBUC e confrontar os agrupamentos gerados pelo CBUC com os agrupamentos gerados pela filogenia tradicional, neste trabalho utilizamos o *Maximum Likelihood* (ML) para gerar a árvore e identificar os grupos monofiléticos e o organismo escolhido para verificar o funcionamento do algoritmo foi o vírus SARS-CoV-2. Os objetivos específicos deste trabalho compreendem:

- Desenvolver ferramenta para a coleta de sequências de forma automática do GenBank;
- Implementar o algoritmo PSRM na linguagem Python;
- Analise comparativa dos resultados do CBUC escrito em Python com a implementação original em Scilab e com o método ML da filogenia tradicional.

A estrutura deste trabalho é descrita a seguir. O Capítulo 2 é constituído pela fundamentação teórica do trabalho, discute os conceitos de bioinformática e filogenia, apresenta o método PSRM e são mostrados alguns trabalhos recentes de análise do genoma do SARS-CoV-2, encontrados a partir da revisão da literatura.

No Capítulo 3 é apresentada a metodologia do trabalho, abordagem quantitativa, juntamente com a identificação da amostra de dados, as definições do instrumentos de coleta e

de tratamento da amostra, e o método de análise dos dados.

O projeto em si é descrito no Capítulo 4, onde são detalhados os *softwares* implementados, buscador de sequências e o algoritmo CBUC, e também são detalhados os ambientes de desenvolvimento dos mesmos.

O Capítulo 5 reúne a discussão dos resultados gerados a partir da análise realizada comparando as implementações do CBUC com a filogenia tradicional. E por fim no Capítulo 6 são dadas as considerações finais com base nas observações feitas.

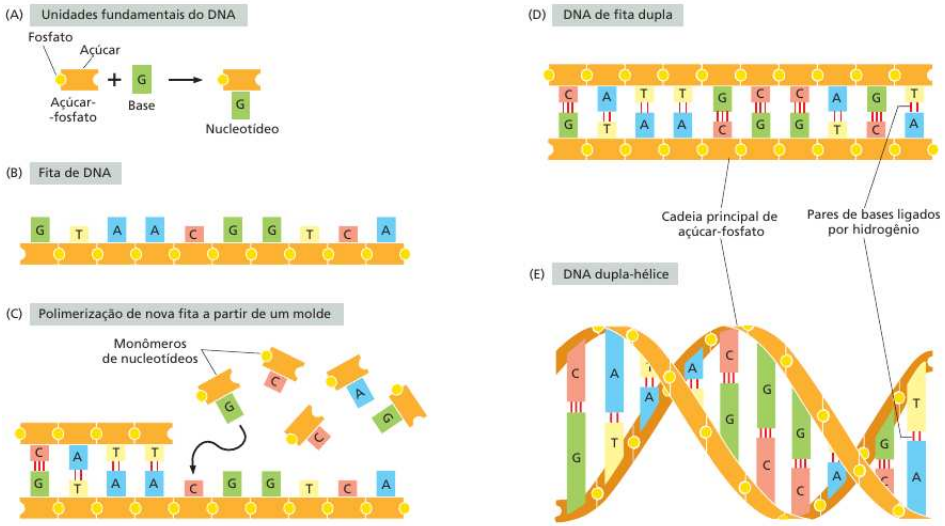
2 FUNDAMENTAÇÃO TEÓRICA

2.1 BIOLOGIA MOLECULAR

A Biologia Molecular é um ramo da biologia que lida com as interações bioquímicas celulares envolvidas na duplicação do material genético e na síntese proteica. Consiste principalmente em estudar as interações entre os vários sistemas da célula, partindo da relação entre o *Deoxyribonucleic Acid* (DNA), o *Ribonucleic Acid* (RNA) e a síntese de proteínas, e o modo como essas interações são reguladas.

> Todas as células vivas da Terra armazenam suas informações hereditárias na forma de moléculas de DNA de fita dupla – longas cadeias poliméricas pareadas não ramifica-das, formadas sempre pelos mesmos quatro tipos de monômeros. Esses monômeros, compostos químicos conhecidos como nucleotídeos, são nomeados a partir de um alfabeto de quatro letras – A, T, C e G – e estão ligados um ao outro em uma longa sequência linear que codifica a informação genética, assim como as sequências de 1s e 0s que codificam as informações em um arquivo de computador (ALBERTS et al., 2017).

Figura 1 – Bases que formam o DNA.



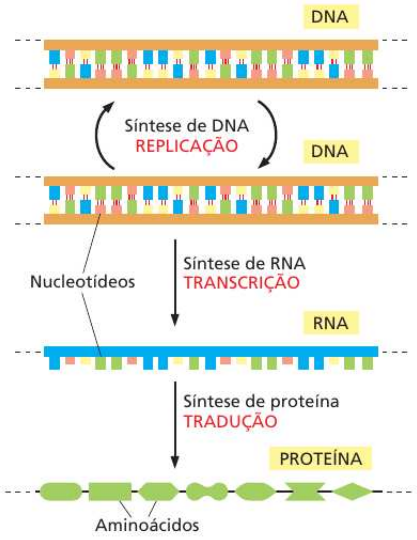
Fonte – Alberts et al. (2017)

O DNA é formado a partir de nucleotídeos, e cada um consiste em uma molécula de açúcar-fosfato com uma cadeia lateral nitrogenada, ou base, ligada a ele. As bases são de

quatro tipos (adenina, guanina, citosina e timina), correspondendo a quatro nucleotídeos distintos nomeados A, G, C e T.

A partir desse mecanismo de armazenamento de informações as células conseguem fazer o processo de replicação, no qual uma parte do DNA é duplicada ilustrado na Figura 1). Essa parte, que é uma uma fita dupla de DNA, é separada em dois gerando duas fitas que servem de molde para criar uma nova fita de DNA. A célula não apenas consegue duplicar o DNA a partir dele mesmo, como também consegue sintetizar outras moléculas: RNA e proteínas. Esse processo começa com a transcrição, no qual a dupla fita de DNA é separada e é usada como molde para síntese de moléculas menores, muito parecidas, os RNAs. Depois no processo mais complexo de tradução, muitas dessas moléculas de RNA controlam a síntese das proteínas. Este é um importante processo, conhecido como dogma central da biologia molecular relatado por Francis Crick em 1957, ilustrado na Figura 2.

Figura 2 – Processo de síntese de moléculas a partir do DNA.



Fonte – Alberts et al. (2017)

No RNA, a cadeia principal é formada pela ribose em vez da desoxirribose, e uma das quatro bases é diferente, a uracila (U) no lugar da timina (T). Durante a transcrição, os monômeros de RNA são alinhados e selecionados para a síntese a partir de uma fita-molde de DNA. O resultado é uma molécula de polímero cuja sequência de nucleotídeos consiste em monômeros de RNA. Esses transcritos são produzidos em massa e são descartáveis, eles atuam transferindo informação genética, servindo como moléculas de mRNA especificando a ordem que deve ser feita a ligação dos aminoácidos no processo de síntese de proteínas.

As moléculas de proteína são cadeias poliméricas longas não ramificadas, assim como as moléculas de DNA e de RNA. Os monômeros de uma proteína, os aminoácidos, são diferentes daqueles do DNA e do RNA, e existem 20 deles. Cada uma das moléculas de proteína é um polipeptídeo, gerado pela ligação de seus aminoácidos. De acordo com Saladin (2012) um peptídeo são cadeias entre 2 e 50 nucleotídeos e um polipeptídeo é uma longa e contínua cadeia não ramificada de peptídeos de aproximadamente 50 aminoácidos.

A sequência contida na molécula de mRNA é lida em grupos de três, formando os códons. Uma proteína pode ser identificada por mais de um códon. No total há 64 códons possíveis, porém apenas 20 deles ocorrem de maneira natural. Dessa forma existem casos no qual vários códons correspondem ao mesmo aminoácido. Esse código genético é lido por pequenas moléculas de RNA, os RNAs transportadores. O tRNA é uma molécula adaptadora composta de RNA, segundo Sharp e Li (1987) sua principal função está no seu envolvimento na tradução do mRNA (RNA mensageiro) para proteína no ribossomo. As funções de tradução incluem: a iniciação da síntese de peptídeos, o alongamento das cadeias de polipeptídeos e o anexo da cadeia crescente de nucleotídeos ao ribossomo.

As proteínas possuem muitas funções: manutenção de estruturas, geração de movimentos, percepção de sinais, etc. As funções das proteínas são dadas de acordo com o sua própria sequência de aminoácidos determinada geneticamente.

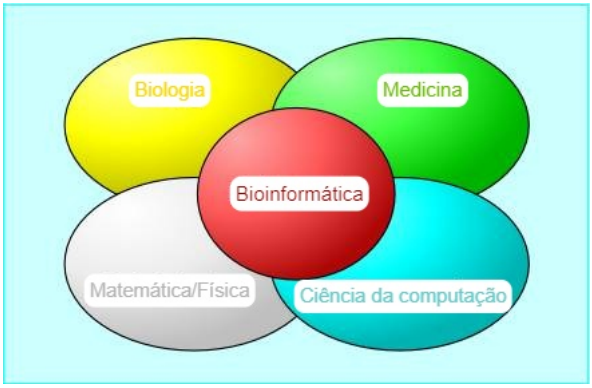
Utilizando métodos químicos, os cientistas conseguem ler a sequência completa dos nucleotídeos em qualquer molécula de DNA, assim conseguem obter toda informação do DNA, a sequencia inteira de DNA que codifica um organismo em particular é chamada de genoma. Dessa maneira é possível sequenciar os genomas dos organismos, e com esses dados sequenciados é possível fazer análises e predições das funções das proteínas.

2.2 BIOINFORMÁTICA

Bioinformática é definida como conjunto de ferramentas de computação e de análise para coletar e interpretar dados biológicos. É um campo interdisciplinar que engloba ciência da computação, matemática, física e biologia (Figura 3). Bioinformática é essencial para gestão dos dados biológicos modernos e da medicina (BAYAT, 2002).

Uma tarefa fundamental de quem trabalha com bioinformática é a análise de sequên-

Figura 3 – Interação das disciplinas que formam a bioinformática.



Fonte – Adaptada de Bayat (2002)

cias de DNA e de proteínas, usando as diversas ferramentas disponíveis. O sítio do *National Center for Biotechnology Information* (NCBI) possui várias ferramentas de bioinformática e de banco de dados genômicos gratuitos. Permitindo que pesquisadores do mundo todo compartilhem suas análises e dados.

A análise e a interpretação dos dados biológicos não fica só no nível do genoma, mas também no nível do proteoma e do transcriptoma. Proteômica se refere a análise do valor total de proteínas expressadas pela célula, e transcriptômica se refere a análise das transcrições do mRNA produzidas pela célula. No geral, os pesquisadores que trabalham com bioinformática utilizam dessa grande massa de dados disponível gratuitamente junto com as ferramentas. Assim utilizando-se da análise, da predição e da classificação das proteínas, eles elaboram hipóteses sobre o funcionamento das mesmas.

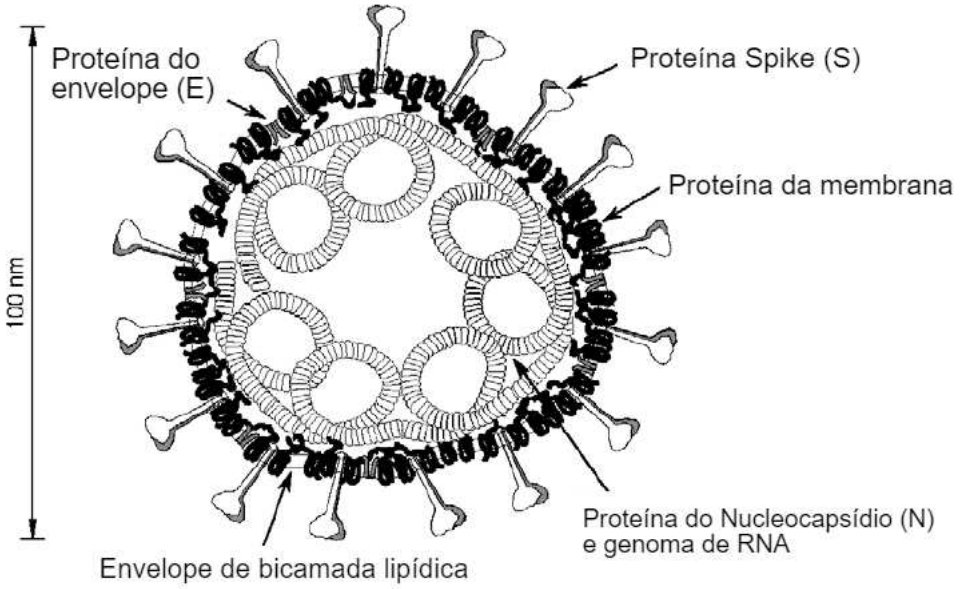
2.3 CORONAVÍRUS

Coronavírus são vírus de RNA envelopado que estão amplamente distribuídos entre os humanos, outros mamíferos e pássaros, causando infecções agudas e persistentes (KNIPE; HOWLEY, 2013). Os coronavírus são classificados atualmente em um dos gêneros da família *Coronaviridae* e foram estudados principalmente por causar doenças reparatórias e intestinais em animais domésticos. Eles não eram considerados patogênicos em humanos até o surto de SARS em 2002 e 2003 na província de Guangdong na China (CUI et al., 2019) e (DROSTEN et al., 2003). Dez anos depois, um outro coronavírus, MERS, surgiu nos países do oriente médio (ZAKI et al., 2012).

Os coronavírus são aproximadamente esféricos, como podemos ver na Figura 4, e

eles podem mudar de forma moderadamente.

Figura 4 – Coronavírus com proteínas mínimas estruturais.

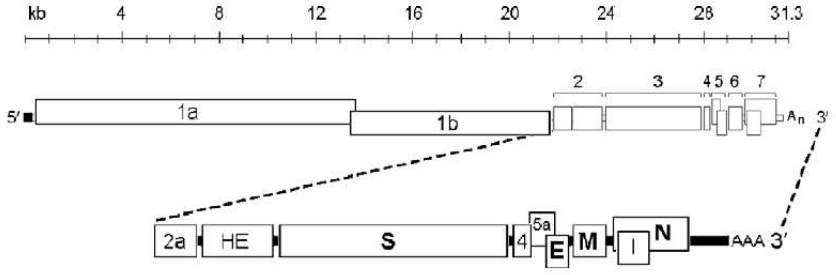


Fonte – Adaptada de Masters (2006)

Existem três proteínas que compõem o envelope viral. A mais proeminente delas é a glicoproteína S, também conhecida como *spike*. A glicoproteína M é a mais abundante dos coronavírus, e formam o corpo do envelope. A proteína do envelope P é um pequeno polipeptídeo que é menos abundante. A proteína do nucleocapsídio, N, é a proteína que compõe a espiral do nucleocapsídio.

O genoma dos coronavírus é não segmentado, fita simples de moléculas de RNA no sentido positivo, isto é, no mesmo sentido do mRNA. Ao contrário da maioria dos mRNAs eucariontes, os genomas dos coronavírus é muito comprido, os genomas do coronavírus estão entre as maiores moléculas maduras conhecidas na biologia (MASTERS, 2006).

Figura 5 – Organização do genoma dos coronavírus.



Fonte – Retirada de Masters (2006)

Na Figura 5 podemos ver como estão organizadas as proteínas no genoma dos

coronavírus e qual parte do genoma codifica qual proteína.

2.4 FILOGENIA

Em relação à evolução, os termos árvore, árvore filogenética e filogenia são usados de maneira que podem ser sinônimos para histórico de ramificação evolutiva ou para gráficos que representam esses históricos evolutivos.

Árvores filogenéticas nos dão uma maneira de entender a continuidade da vida de indivíduos para espécies. Cada organismo vivo no planeta tem um ou dois ancestrais diretos, e esses ancestrais também tem ancestrais. Assim, pode-se traçar o histórico evolutivo de um organismo voltando no tempo de ancestral para ancestral. Caminhando na árvore de dois organismos diferentes, chegará em um momento no qual os caminhos vão convergir em um ancestral comum, que é o ancestral dos dois organismos de onde a caminhada começou. A ancestralidade comum nos dá uma maneira natural de entender as conexões entre os indivíduos de diferentes populações e espécies, e o diagrama de árvore provê uma maneira de visualizar e resumir ancestralidade comum. A estrutura em árvore trabalha em diversas escalas, desde herança de genes únicos, até uma população inteira de espécies relacionadas e até linhagens.

Figura 6 – Partes de uma árvore filogenética.

Fonte – Autor

Podemos ver na Figura 6 as partes de uma árvore filogenética, um diagrama de árvore é feito de linhas chamadas ramos ou arestas, conectados pelos nós. O diagrama precisa ser direto, indo apenas em uma direção no tempo e também precisa ser acíclico, as linhagens que divergem nunca se juntam subsequentemente. O texto no final da árvore pode indicar espécies individuais que representam uma espécie em particular ou um conjunto de espécies que constituem um

ramo na árvore da vida, no caso dessa árvore em específico são os identificadores das sequências com o identificador do *GenBank* e o país. O item representado por esses textos são as folhas ou terminais. Os ramos representam as linhagens em evolução onde os nós correspondem aos eventos de "separação". Um nó marca o último ancestral comum dos organismos das linhagens filhas, enquanto os ramos internos conectam dois nós, e os ramos externos conectam um nó e uma folha.

2.4.1 Inferência Hennigiana

No meio do século 20, o entomologista alemão Willi Hennig e seus colegas desenvolveram o primeiro método formal para reconstrução de filogenia, descrito em seu livro de 1966 "Phylogenetic Systematic". O método de inferência filogenética Hennigiana consiste em identificar os conjuntos de grupos mais próximos que compartilham um estado de uma característica e inferindo que eles formam um clado. Aplicando este método conseguimos desenhar uma árvore que contém todos os clados com suas características.

Antes de Hennig os cientistas careciam de protocolos bem definidos para reconstrução filogenética. O método Hennigiano foi bastante popular durante a época dos anos 60 a 70, porém não é mais usado. O principal problema da inferência Hennigiana é que faz premissas irreais sobre características evolutivas e não provem uma maneira clara de procedimento quando essas premissas não atendem.

O método Hennigiano assume que homoplasia é ausente, mas a homoplasia acontece. Homoplasia é quando uma característica específica aparece mais de uma vez na mesma árvore. Algumas vezes duas características evolutivas acontecem em paralelo. A lógica de Hennig não pode ser usada para deduzir a verdadeira árvore nesses casos, então foi criado um método para substituir a inferência Hennigiana e o método proposto foi o ***Maximum Parsimony***.

2.4.2 Critério de *Maximum Parsimony*

Uma vez que nós temos conhecimento que algumas características mostram homoplasia, uma maneira lógica de proceder é permitir que a homoplasia ocorra, mas diminuir a quantidade que acontece. O critério de *Maximum Parsimony* sustenta que a melhor estimativa de filogenia é a árvore que explica todos dados observados chamando a última homoplasia, ou seja, o menor estado da característica muda. A implementação mais simples de *parsimony* segue em

três passos:

- Para uma única árvore, nós consideramos cada característica em turno e determinamos o número mínimo de mudanças no estado da característica, ou passos, que são necessários para contar a distribuição de estados entre as folhas;
- Somamos o número de passos necessários para cada característica. O número de passos necessários para explicar todas as características evolutivas é chamado de comprimento da árvore;
- Repetimos as etapas do processo para todas as árvores alternativas e então identificamos a árvore com o menor comprimento, que é a mais curta ou a árvore mais parcimônia.

2.4.3 Abordagem baseada em modelos matemáticos

O núcleo de qualquer modelo matemático de característica evolutiva é o modelo de substituição, que especifica a maneira em que as características são permitidas de evoluir entre estados, bem como a taxa dos diferentes tipos de mudanças evolutivas. Todos os modelos amplamente usados em filogenia são modelos de Markov de tempo contínuo, isto é, eles descrevem um processo no qual a probabilidade de um evento ocorrer em alguma janela de tempo é dependente apenas do estado naquele tempo e independente de como ele chegou naquele estado. Os modelos de evolução de sequência de DNA possuem apenas quatro estados, que são correspondentes às quatro bases, A, T, G e C.

Inferência filogenética é essencialmente uma tentativa de determinar há quanto tempo atrás um par de organismos ou grupo de organismos compartilharam o mesmo ancestral comum. O modelo mais simples foi desenvolvido por Jukes e Cantor e é conhecido como Jukes-Cantor ou modelo de evolução molecular JC. Este modelo assume que todas as quatro bases ocorrem na mesma frequência, cada tipo de substituição (A por C, A por T) ocorre na mesma frequência e a taxa de substituição é a mesma para todas posições dos nucleotídeos da sequência estudada.

2.4.3.1 Métodos de distância

O fato principal subjacente aos métodos de distância é que se nós soubéssemos a verdadeira distância entre cada par de organismos (definido como a média de substituições por sítio em uma sequência de DNA), então essas distâncias corresponderiam em uma única árvore. A distância evolutiva entre dois organismos quaisquer é a soma dos comprimentos de todos os

ramos no caminho entre esses dois organismos, portanto, saber a verdadeira distância evolutiva é equivalente a saber a árvore. Os objetivos dos métodos de distância são determinar as distâncias entre os organismos e usá-las para inferir a verdadeira árvore filogenética.

2.4.3.2 *Maximum Likelihood*

O critério do *ML* não é específico para filogenia, mas sim uma abordagem geral usada através da estatística. A aplicação do ML em filogenia consiste em procurar pela árvore que tem a maior probabilidade de dar origem ao dado observado. No caso da filogenia os dados observados serão as características de cada organismo (a matriz de estado de características) e as hipóteses são todas as possíveis árvores. É também necessário um modelo matemático de evolução de características que será aplicado pelo ML.

2.4.3.3 Inferência Bayesiana

A inferência Bayesiana avalia as árvores baseado na probabilidade posterior delas, a probabilidade de que a árvore é verdadeira, os dados, os modelos evolutivos e as crenças prévias. O princípio de calcular a probabilidade posterior foi desenvolvido por Reverend Thomas Bayes no século XVIII, ele provou matematicamente que a probabilidade de uma hipótese de acordo com algum dado, é igual a probabilidade dos dados, dada a hipótese (*likelihood*), podemos ver a definição formal na equação (2.1).

$$Pr(H|D) = \frac{Pr(D|H) \times Pr(H)}{Pr(D)} \tag{2.1}$$

Aplicando os princípios em filogenia ficaria como na equação (2.1):

$$Pr(Arvore|Dados) = \frac{Pr(Dados|Arvore) \times Pr(Arvore)}{Pr(Dados)} \tag{2.2}$$

A probabilidade anterior de uma árvore em particular, Pr(Árvore), é a probabilidade (depois de olhar para os dados) que entre todas as árvores possíveis é verdadeira. Calcular a probabilidade dos dados segundo uma árvore, Pr(Dados|Árvore), envolve determinar o *likelihood* da árvore. O maior desafio é calcular a probabilidade anterior dos dados, Pr(Dados), envolve

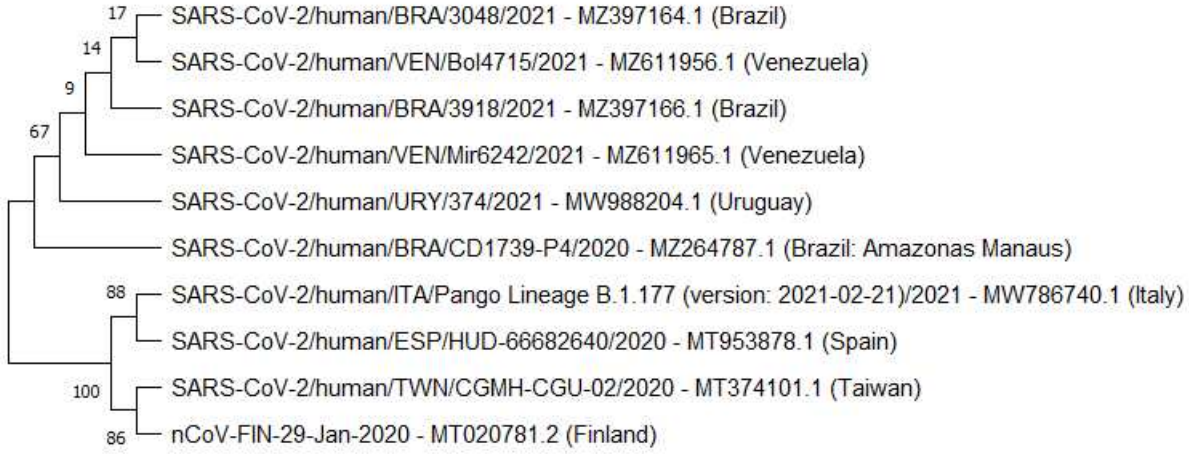
a soma de todas as árvores. Para vencer esse problema é necessário usar o método de análise Markov Chain Monte Carlo (MCMC). Ele explora o fato de que não podemos calcular facilmente o verdadeiro posterior, mas podemos calcular os posteriores relativos de diferentes árvores.

2.4.4 Análise *Nonparametric Bootstrap*

Análise *Nonparametric Bootstrap*, ou apenas *bootstrap*, é amplamente usado para avaliar a força de uma clado. As tentativas de bootstrap avaliam as chances de recuperar um clado específico novamente, se pudéssemos testar um novo conjunto de características. A ideia é obter novos conjuntos de dados randomizando o conjunto inicial.

Em uma análise de *bootstrap* completa, a proporção das amostras de dados do *bootstrap* que uma árvore produz com um dado clado é contada. Este é o *bootstrap score* para aquele clado, se 80 de 100 amostras formaram um clado X, então o clado X tem um *bootstrap score* de 80%, na Figura 7 temos um exemplo uma árvore gerada utilizando 500 replicações de *boostrap* gerada pelo MEGA, e podemos observar que o clado que contém a Itália, Espanha, Finlândia e Taiwan possui um *score* de 100%, ou seja das 500 vezes essas quatro sequências se juntaram todas as vezes.

Figura 7 – Árvore gerada com 500 replicações de *bootstrap*.



Fonte – Autor

2.5 PSRM

O *Parametric State Recognition (PSR) Method* é uma abordagem de aprendizado não supervisionada de agrupamento, originalmente desenvolvido para processar múltiplas séries

temporais e que teve origem no monitoramento de equipamento industrial por um *Digital Twin*. *Digital twin* é uma representação virtual que serve como contrapartida digital em tempo real de um objeto ou processo físico (GRIEVES; VICKERS, 2017).

As entradas que o método espera são uma lista de números normalizados misturados e de características categóricas que são os parâmetros que serão analisados. O pré-processamento é formado por duas camadas, uma derivativa que calcula as diferenças finitas para frente, para trás e centradas, e a integrativa que calcula a média acumulativa e baseada em janela.

Os hiper-parâmetros são o número de variáveis, quando for maior que um, determina a resolução do método e está relacionado ao número de características por agrupamento, e as combinações e os tipos de camadas de pré-processamento. Os parâmetros de conhecimento são o mapa de agrupamentos (MOC), dicionário de estados (CDIC) são as assinaturas de microestados e o dicionário de famílias (FDIC) são as ligações de microestado para macroestado. E a saída do método são os micro e macro estados gerados. O PSRM agrupa os dados de entrada e gera micro e macro agrupamentos.

2.5.1 Etapa de treinamento

O conjunto bruto de dados consiste em instâncias de características bem limitadas, isto é, com dados valores mínimos e máximos para variáveis numéricas e classes numéricas constantes para as categóricas. As características brutas são normalizadas e eventualmente derivadas e integradas seguidas da normalização no processo de pré-processamento. Durante o processo de retreinamento e classificação, instâncias com características desvinculadas são desconsideradas. No caso que amostras desconsideradas sejam encontradas, para estender a faixa viável de algumas características devem ser decididas se: os limites de algumas características são atualizados, que implica em retreinamento, ou, características truncadas para os limites existentes, ou, deixa assim por que a amostra é fora da curva.

A etapa de treinamento consiste em duas etapas: etapa de partição, na qual as amostras de treinamento são distribuídas em micro agrupamentos; etapa aglomerativa, na qual os micro-agrupamentos são unidos para formar os macro-agrupamentos.

2.5.1.1 Etapa de partição

Depois de ler as ***N*** amostras do conjunto de dados de treinamento, um histograma com *C* variáveis é construído para cada característica numérica normalizada. Características categóricas não são consideradas nesta etapa. Um algoritmo de busca de agrupamentos é executado em cada característica numérica identificando picos e vales. Cada pico representa um agrupamento para cada característica. ***Nf*** denota o número de agrupamentos em cada característica ***F***.

Características sem ou com apenas um pico são excluídas de cada conjunto de características. Uma distribuição de probabilidade, ***pdf***, é ajustada para cada pico sequencialmente, usando o domínio entre os vales adjacentes, de tal maneira que a soma dos ***pdfs*** se aproxime do histograma da característica. Tais ***pdfs*** são usados como funções de pertinência para identificar o agrupamento de um dado valor de uma característica pertencente a ***cf*** no intervalo [1, ***Nf***]. Olhando para as interceptações entre os ***pdfs*** adjacentes, os limites de cada agrupamento, ***Nf***, são guardados no MOC.

Para cada classe de características categóricas é atribuído um número de agrupamentos tal qual ***Nf*** é igual ao número de classes de características categóricas. Uma parte reservada do MOC contém o número de associações classe-agrupamento para cada classe categórica. A coleção de m inteiros ***cf***, para ***f*** = { 1, 2, 3, ..., ***m*** }, isto é { ***c1***, ***c2***, ..., ***cm*** } para cada uma das ***n*** amostras de dados, se define uma assinatura de micro-agrupamento no espaço da característica. Todas as novas micro-assinaturas são armazenadas no dicionário de agrupamentos, CDIC, recebendo um índice ordinal. A amostra que apresenta a nova micro-assinatura é rotulada com o novo índice. Amostras com micro-assinaturas que já estão no CDIC, são rotuladas com os índices correspondentes. Note que, no final o número de características do primeiro passo, ***m***, pode ser menor que o número original de características, porque as características que não possuem pelo menos dois picos no histograma são excluídas. Em outras palavras, o primeiro passo também é de distinção de características. As características também podem ser ordenadas pelo número de agrupamento ***Nf*** que está diretamente relacionado com a importância da característica no processo de agrupamento.

2.5.1.2 Etapa aglomerativa

No final da primeira etapa nós temos no CDIC uma lista com $\boldsymbol{u}$ menor ou igual a $\boldsymbol{n}$ assinaturas de micro-agrupamentos, onde cada assinatura é um vetor de $\boldsymbol{m}$ inteiros. Calcular a distância entre cada par de assinaturas de micro-agrupamentos usando uma métrica apropriada para os dados híbridos, numéricos e categóricos, a distância PSRM nesse trabalho, ou qualquer distância no caso de serem todas as características numéricas. As distâncias são organizadas no triângulo superior da matriz de $\boldsymbol{u}$ x $\boldsymbol{u}$ elementos, tal qual a distância na linha $\boldsymbol{i}$ e coluna $\boldsymbol{j}$, $\boldsymbol{j}$ maior que $\boldsymbol{i}$, é a distância entre as assinaturas de micro-agrupamentos $\boldsymbol{i}$ e $\boldsymbol{j}$ na lista de CDIC. Dada matriz triangular de distâncias, qualquer método hierárquico de agrupamento pode ser aplicado.

Porém, aqui nós aplicamos um tipo de método de vizinho mais próximo consistindo em, inicializar um conjunto de dados de dicionário família (FDIC) no qual as linhas são famílias e as colunas são a lista de assinaturas, seus índices k no conjunto $\{ 1, 2, \ldots, \boldsymbol{u} \}$ pertencendo a cada família. E depois transformar a matriz de distâncias em um conjunto de dados com três colunas de distancia, $\boldsymbol{i}$, $\boldsymbol{j}$, e ordená-la de forma crescente pela distância. Percorrer o conjunto formado perguntando:

- $\boldsymbol{i}$ está no FDIC?
  - Se **sim**: $\boldsymbol{j}$ está no FDIC?
    - Se **não**: adiciona $\boldsymbol{j}$ na mesma família que $\boldsymbol{i}$ no FDIC
  - Se **não**: $\boldsymbol{j}$ está no FDIC?
    - Se **sim**: adiciona $\boldsymbol{i}$ na mesma família que $\boldsymbol{j}$ no FDIC
    - Se **não**: adiciona uma nova família e inclui $\boldsymbol{i}$ e $\boldsymbol{j}$ como membros.

É altamente recomendado incluir a coluna no conjunto de dados (CDIC) para atualizar a família de cada micro-agrupamento durante o processo de procurar família. No final da segunda etapa nós temos no FDIC uma lista de $\boldsymbol{M}$ macro-agrupamentos com seus membros, dados como índices na lista de micro-agrupamentos CDIC.

2.5.2 Etapa pós-treino

Uma vez que os macro-agrupamentos foram encontrados, calcularemos para cada um deles: usando a matriz de distância, coleta-se a distribuição de distância dos micro-agrupamentos internos, e calcula-se a média, $\boldsymbol{md}$, e a variância, $\boldsymbol{sd}$. A maior distância entre os membros do

macro-agrupamentos $\boldsymbol{c}$, $\boldsymbol{dc}$, c = { 1, 2 ... $\boldsymbol{M}$ } e $\boldsymbol{D}$ = max($\boldsymbol{dc}$). E os centroides, $\boldsymbol{Cc}$, $\boldsymbol{c}$ = { 1, 2, ... $\boldsymbol{M}$ }, como centro de massa dos macro-agrupamentos. Depois calcula-se uma matriz triangular superior com as distâncias entre os centroides, e então calcula a métrica de qualidade do agrupamento.

2.5.3 Operação e retreino

Uma vez que o FDIC está preenchido e a coluna "família" no conjunto de dados do CDIC está atualizada, com uma nova instância de dados o processo segue: para cada característica procura-se o número do agrupamento e constrói a amostra de assinatura $\boldsymbol{ss}$, procura pela assinatura $\boldsymbol{ss}$ no CDIC se encontra retorna o número da família correspondente, senão adiciona a assinatura $\boldsymbol{ss}$ no CDIC e calcula a distância de $\boldsymbol{ss}$ para todas as outras assinaturas no CDIC e encontra a assinatura mais próxima, $\boldsymbol{cs}$. $\boldsymbol{c*}$ denota o índice do macro-agrupamento onde $\boldsymbol{cs}$ está. Então se a distância de $\boldsymbol{cs}$ e $\boldsymbol{ss}$ for menor que $\boldsymbol{dc*}$ / 2 e a distância entre $\boldsymbol{ss}$ e $\boldsymbol{Cc*}$ for menor que $\boldsymbol{dc*}$ / 2 então atribui $\boldsymbol{ss}$ ao macro-agrupamento $\boldsymbol{c*}$, senão adiciona um novo macro-agrupamento no FDIC com $\boldsymbol{ss}$ como membro.

O número de macro-agrupamento não pode ser definido a priori. Quando se usa o método de encontrar a família PSRM, contudo há duas formas que indiretamente causam redução no número de macro-agrupamento: reduzindo as características a um número menor que o de agrupamentos, gradativamente até atingir o número desejado de agrupamentos, ou reduzindo o número de variáveis $\boldsymbol{C}$.

2.5.4 Pré-processamento

O pré-processamento das camadas de pode ser feito de maneira serial, paralela ou híbrida.

2.5.4.1 Paralelo

O pré-processamento paralelo consiste em: normalizar as características cruas, selecionar as características cruas, $\boldsymbol{R}$, cálculo das derivadas normalizadas das características selecionadas $\boldsymbol{D}$, cálculo das médias integrais normalizadas das características selecionadas $\boldsymbol{I}$, agrupa $\boldsymbol{R}$, $\boldsymbol{D}$ e $\boldsymbol{I}$ para formar o vetor de entrada.

2.5.4.2 Serial

O pré-processamento serial consiste inicialmente em: normalizar as características cruas e selecionar as características cruas, ***R***. Depois seguem dois processos: o pré-processamento ***D-I*** e o pré-processamento ***I-D***. O pré-processamento ***D-I*** faz o cálculo das derivadas normalizadas das características selecionadas ***D***, cálculo das médias integrais normalizadas de ***D*** gerando ***I***, agrupa ***R***, e ***I*** para formar o vetor de entrada. E o pré-processamento ***I-D*** faz o cálculo cálculo das médias integrais normalizadas das características selecionadas ***I***, cálculo das derivadas normalizadas de ***I*** gerando ***D*** e agrupa ***R*** e ***D*** para formar o vetor de entrada.

2.6 TRABALHOS CORRELATOS

Várias análises foram feitas da filogenia do SARS-CoV-2 utilizando os métodos tradicionais, como o Neighbor Joining (NJ) e o MCMC. Essas análises seguem o procedimento tradicional que é adicionar as sequências que se quer analisar em um conjunto de dados e gerar a árvore filogenética, e analisam em qual clado se agrupou.

Hu et al. (2021) analisaram o Sars-Cov-2 junto com as versões anteriores dele, MERS, SARS etc. Foi feita a análise filogenética com o método NJ e calculada a diferença entre os genomas utilizando o *software* MEGA 6. As sequências do SARS-CoV-2 se agruparam em um clado com ancestral comum com a versão do vírus que infecta os morcegos, dando indícios de sua provável origem. É observado também que a sequência é 96.6% idêntica com o coronavírus do morcego de identificador RaTG13.

O artigo de Yadav et al. (2020) estuda as duas primeiras sequências da Índia. As sequências para fazer a análise foram coletadas do GenBank e o programa usado para fazer as análise foi o MEGA com o método NJ. As duas sequências se agruparam em clados separados com sequências da China, mas a análise do genoma indicou que essas sequências não eram 100% idênticas.

O trabalho de Zehender et al. (2020) faz uma análise das sequências de SARS-CoV-2 da Itália que foram isoladas de alguns pacientes da Itália e o método utilizado para a análise foi o MCMC. As sequências para formar o conjunto de dados foram coletados do banco GISAID. Três das sequências da Itália formaram um clado com as sequências da Alemanha, Finlândia, México e Brasil.

Os pesquisadores dessas análises podem querer verificar o agrupamento de novas sequências, e para isso tem que criar uma árvore filogenética, em um procedimento demorado e custoso, para analisar os agrupamentos. A proposta do CBUC é para auxiliar nessas classificação de uma nova sequência para verificar se ela faz parte de uma família que já existe ou é uma família nova. De uma forma mais rápida e aproveitando os treinamentos anteriores do algoritmo.

3 METODOLOGIA

Tendo em vista que o objetivo deste trabalho é identificar padrões e agrupamentos das sequências da proteína *spike* do SARS-CoV-2, este trabalho fundamenta-se na abordagem conhecida como pesquisa quantitativa que consiste em fazer uso de métricas e de recursos estatísticos para mensurar os resultados obtidos, com o propósito final de validá-los e de representá-los. Para uma pesquisa quantitativa deve-se identificar a amostra, definir os instrumentos de coleta de dados e os procedimentos de análise de dados (CRESWELL, 2007).

Este trabalho analisa a proteína *spike* do SARS-CoV-2, e compara os agrupamento monofiléticos do SARS-CoV-2 encontrados com a filogenia tradicional, ML, utilizando 500 replicações de *bootstrap*. As sequências serão coletadas do *GenBank* da biblioteca *nucleotide*. O *GenBank* é um banco de dados público de sequências de nucleotídios ele foi construído e distribuído pela *National Institutes of Health* (NIH) *(SAYERS et al., 2021)*. As sequências foram primeiramente coletadas dos artigos, e depois o conjunto de dados foi complementado posteriormente.

Para coletar mais facilmente as sequências precisou-se construir uma ferramenta automática para coletar as sequências do *GenBank* e salvar em um banco de dados. Em trabalhos futuros, busca-se publicar essa ferramenta online, para que os pesquisadores possam fazer suas análises, então o algoritmo CBUC teve que ser implementado em outra linguagem de programação, para que futuramente possa-se publica-lo. A linguagem escolhida foi o Python, pelas bibliotecas que ele possui para fazer cálculos com vetores e matrizes e de plotagem de gráficos que é muito importante para a exibição dos resultados.

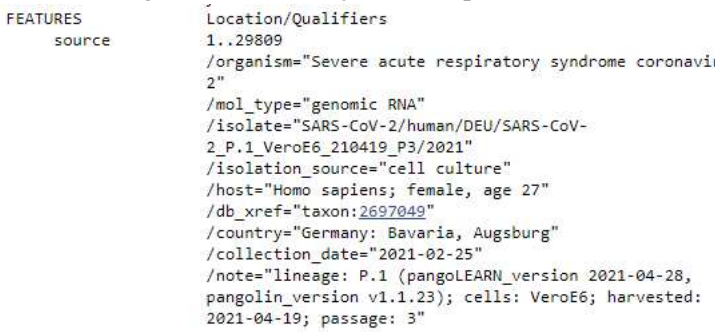
3.1 AMOSTRAGEM

As nossas amostras são as sequências da proteína *spike* do SARS-CoV-2, queremos amostras espalhadas em espaço e tempo para ter uma amostra bem diversa. Então alguns países principais foram escolhidos, não limitando-se apenas a eles, para a coleta das sequências: China, Itália, França, Inglaterra, Africa do Sul, Estados Unidos, Brasil, Austrália e Índia. Foram selecionadas sequências aleatórias de cada país, inicialmente aquelas que foram referenciadas em

artigos publicados, pois os artigos passaram por revisão por pares, e isso garante uma sequência mais confiável.

Para a análise foram selecionas as sequências ótimas da proteína *spike*. Nesse trabalho consideramos sequências ótimas, aquelas que no *GenBank* possuem suas regiões e proteínas identificadas e possui todos os nucleotídeos identificados, ou seja nenhum carácter diferente de A, T, G ou C. Na Figura 8 podemos ver as informações da sequência de identificador *MZ427312.1* e como são mostradas no *site* do *GenBank*.

Figura 8 – Informações da sequência *MZ427312.1*.

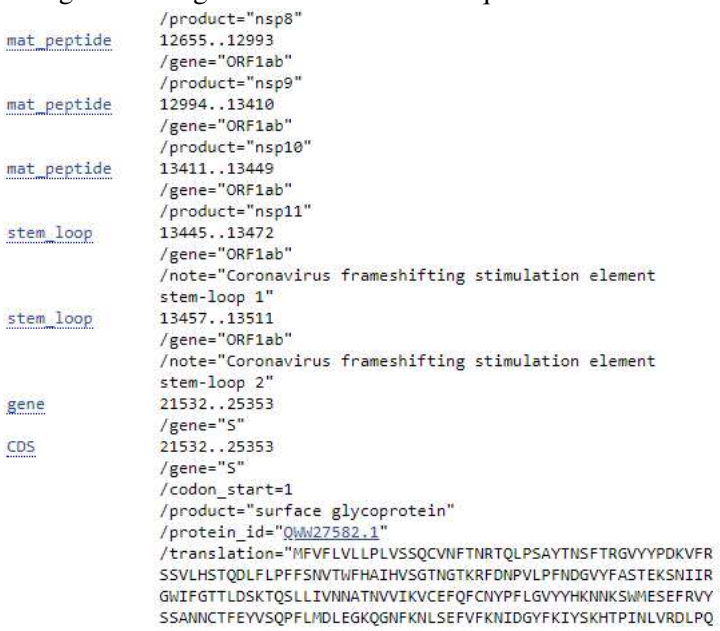


```
FEATURES             Location/Qualifiers
     source          1..29809
                     /organism="Severe acute respiratory syndrome coronavirus
                     2"
                     /mol_type="genomic RNA"
                     /isolate="SARS-CoV-2/human/DEU/SARS-CoV-
                     2_P.1_VeroE6_210419_P3/2021"
                     /isolation_source="cell culture"
                     /host="Homo sapiens; female, age 27"
                     /db_xref="taxon:2697049"
                     /country="Germany: Bavaria, Augsburg"
                     /collection_date="2021-02-25"
                     /note="lineage: P.1 (pangoLEARN_version 2021-04-28,
                     pangolin_version v1.1.23); cells: VeroE6; harvested:
                     2021-04-19; passage: 3"
```

Fonte – GenBank

Na figura 9 são mostradas algumas de suas regiões identificadas e a proteína *spike*. Essa sequência possui todos os nucleotídeos identificados então considerados essa uma sequência ótima.

Figura 9 – Regiões identificadas da sequência *MZ427312.1*.



```
                     /product="nsp8"
mat_peptide          12655..12993
                     /gene="ORF1ab"
                     /product="nsp9"
mat_peptide          12994..13410
                     /gene="ORF1ab"
                     /product="nsp10"
mat_peptide          13411..13449
                     /gene="ORF1ab"
                     /product="nsp11"
stem_loop            13445..13472
                     /gene="ORF1ab"
                     /note="Coronavirus frameshifting stimulation element
                     stem-loop 1"
stem_loop            13457..13511
                     /gene="ORF1ab"
                     /note="Coronavirus frameshifting stimulation element
                     stem-loop 2"
gene                 21532..25353
                     /gene="S"
CDS                  21532..25353
                     /gene="S"
                     /codon_start=1
                     /product="surface glycoprotein"
                     /protein_id="QWW27582.1"
                     /translation="MFVFLVLLPLVSSQCVNFTNRTQLPSAYTNSFTRGVYYPDKVFR
                     SSVLHSTQDLFLPFFSNVTWFHAIHVSGTNGTKRFDNPVLPFNDGVYFASTEKSNIIR
                     GWIFGTTLDSKTQSLLIVNNATNVVIKVCEFQFCNYPFLGVYYHKNNKSWMESEFRVY
                     SSANNCTFEYVSQPFLMDLEGKQGNFKNLSEFVFKNIDGYFKIYSKHTPINLVRDLPQ
```

Fonte – GenBank

Na Figura 10 é mostrada a sequência de identificador *MZ427313.1*, é uma sequência que possui também suas regiões bem identificadas, porém como podemos ver na Figura 11 que ela possui um sequenciamento incompleto, esta faltando uma boa parte da sequência e ela possui mais partes incompletas.

Figura 10 – Regiões identificadas da sequência *MZ427313.1*.

```
                     /product= RNA-dependent RNA polymerase
mat_peptide          16199..18001
                     /gene="ORF1ab"
                     /product="helicase"
mat_peptide          18002..19582
                     /gene="ORF1ab"
                     /product="3'-to-5' exonuclease"
mat_peptide          19583..20620
                     /gene="ORF1ab"
                     /product="endoRNAse"
mat_peptide          20621..21514
                     /gene="ORF1ab"
                     /product="2'-O-ribose methyltransferase"
CDS                  228..13445
                     /gene="ORF1ab"
                     /codon_start=1
                     /product="ORF1a polyprotein"
                     /protein_id="QWW27593.1"
                     /translation="MESLVPGFNEKTHVQLSLPVLQVRDVLVRGFGDSVEEVLSEARQ
```

Fonte – GenBank

Figura 11 – Parte não sequenciada da sequência *MZ427313.1*.

```
 421 aacagcccta tgtgttcatc aaacgttcgg atgctcgaac tgcacctcat ggtcatgtta
 481 tggttgagct ggtagcagaa ctcgaaggca ttcagtacgg tcgtagtggt gagacacttg
 541 gtgtccttgt ccctcatgtg ggcgaaatac cagtggctta ccgcaaggtt cttcttcgta
 601 agaacggtaa taaaggagct ggtggccata gttacggcgc cgatctaaag tcatttgact
 661 taggcgacga gcttggcac
     [gap 234 bp]    Expand Ns
 914               ctgctgc cgtgaacatg agcatgaaat tgcttggtac acggaacgtt
 961 ctgaaaagag ctatgaattg cagacacctt ttgaaattaa attggcaaat aaatttgaca
1021 ccttcaatgg ggaatgtcca aattttgtat ttcccttaaa ttccataatc aagactattc
1081 aaccaagggt tgaaaagaaa aagcttgatg gctttatggg tagaattcga tctgtctatc
1141 cagttcgctc accaaatgaa tgcaaccaaa tgtgcctttc aactctcatg aagtgtgatc
```

Fonte – GenBank

Na Figura 12 vemos que a sequência de identificador *OK091006.1* não possui alguns nucleotídeos identificados, apesar de possuir suas regiões e proteínas bem identificadas. Essas sequências não estão qualificadas para o nosso conjunto de dados.

Figura 12 – Nucleotídeos não identificados da sequência *OK091006.1*.

```
5881 ttataaattg gatggtgttg tttgtacaga aattgaccct aagttggaca attattataa
5941 gaaagacaat tcttatttca cagagcaacc aattgatctt gtaccaaacc aaccatatcc
6001 aaacgcaagc ttcgataatt ttaagtttgt atgtgataat atcaaatttg ctgatgattt
6061 aaaccagtta actggttata agaaacctgc ttcaagagag cttaaagtta cattttccc
6121 tgacttaaat ggtgatgtgg tggctattga ttataaacac tacacaccct cttttaagaa
6181 aggagctaaa ttgttacata aacctattgt ttggcatgtt aacaatgcaa ctaataaagc
6241 cacgtataaa ccaaatacct ggtgtatacg ttgtctttgg annnnnnnnn nnnnnnnnnn
6301 nnnnnnnnnn nnnnatgtac tgaagtcaga ggacgcgcag ggaatggata atcttgcctg
6361 cgaagatcta aaactagtct ctgaagaagt agtggaaaat cctaccatac agaaagacgt
6421 tcttgagtgt aatgtgaaaa ctaccgaagt tgtaggagac attatactta aaccagcaaa
6481 taatagttta aaaattacag aagaggttgg ccacacagat ctaatggctg cttatgtaga
6541 caattctagt cttactatta agaaacctaa tgaattatct agagtattag gtttgaaaac
6601 ccttgctact catggtttag ctgctgttaa tagtgtccct tgggatacta tagctaatta
6661 tgctaagcct tttcttaaca aagttgttag tacaactact aacatagtta cacggtgttt
```

Fonte – GenBank

3.2 COLETA E TRATAMENTO DA AMOSTRA

3.2.1 Armazenamento

Para guardar os dados das sequências do Sars-Cov-2, optou-se utilizar o banco de dados não relacional orientado a documentos desenvolvido na linguagem C++, MongoDB (MongoDB Inc., 2009). Como não haverá muitas entidades nem relacionamentos um banco não relacional foi escolhido.

3.2.2 Coleta e armazenamento das sequências

Foram selecionadas artigos da base do PUBMED, artigos que fazem análise do Sars-Cov-2. Depois foram salvas em *Comma-Separeted Values* (CSV) os identificadores das sequências que foram analisadas pelos artigos. Depois disso foi utilizada a ferramenta de busca criada para ler o arquivo de CSV e buscar no GenBank a sequência pelo nome e armazenar os dados modelados no banco.

Depois que os dados das sequências foram coletados do GenBank, então a ferramenta filtra pelo nome do organismo e hospedeiro, para armazenar no banco apenas os vírus Sars-Cov-2 e hospedeiros humanos. Assim as sequências filtradas são salvas no banco.

3.2.3 Amostra de dados e tratamento da amostra

Com as sequências armazenadas no banco, a ferramenta consulta o banco e busca todas as sequências armazenadas e seleciona no máximo dez sequências de cada país armazenado no banco. Então é criada uma amostra de dados no formato FASTA, formato baseado em texto para representar sequências de nucleotídeos ou de amino ácidos.

Com o conjunto de dados FASTA será feito um tratamento fazendo um alinhamento das sequências usando o algoritmo MUSCLE do software MEGA (Molecular Evolutionary Genetics Analysis across Computing Platforms) (KUMAR et al., 2018). MUSCLE é um programa para gerar múltiplos alinhamentos de aminoácidos e sequências de nucleotídeos (EDGAR, 2004). Por fim uma edição das sequências, fazendo um corte no início e no final das sequências caso tenha ficado algum carácter inválido depois do alinhamento.

3.3 ANÁLISE DE DADOS

A análise das sequências será feita com o algoritmo CBUC, implementado no Python, o Python foi escolhido pela quantidade de recursos de análise de dados que possui como as bibliotecas de plotagem de gráfico e pela biblioteca de operações com matrizes e vetores, *numpy*, entre outros recursos, que ajudaram na implementação do algoritmo. E também é utilizada a versão no Scilab. É também gerada a árvore filogenética desse conjunto de sequências para comparar os resultados com os resultados do CBUC no Python e no Scilab.

4 DESENVOLVIMENTO DO PROJETO

O projeto compreende a implementação de dois *softwares*: o buscador de sequências e o o algoritmo CBUC implementado em Python. O buscador recebe os arquivos CSV e salva as sequências no banco de dados, e também gera um conjunto FASTA com esse conjunto de dados. O conjunto de dados é aberto no programa MEGA, que faz o alinhamento das sequências e em seguida o CBUC recebe as sequências alinhadas e busca os padrões de códons e faz os agrupamentos.

4.1 BANCO DE DADOS

As sequências foram armazenados em um *cluster* do MongoDB no Atlas, que é uma solução em nuvem que o MongoDB fornece de banco de dados como serviço (MongoDB Inc., 2016). O banco guardou os seguintes dados das sequencias:

- Nome da sequência (nome dado a sequência isolada)
- Nome do organismo (nome do vírus)
- Genoma completo (sequência de nucleotídeos que codifica o vírus)
- País (local onde ela foi isolada)
- Hospedeiro (humano, animal, etc)
- Data de coleta da sequência
- Início e fim da proteína *spike* (de qual carácter começa e quando termina a proteína *spike*)
- Observações (que podem ter sido feitas sobre a sequência)

4.2 BUSCADOR DE SEQUÊNCIAS

Essa ferramenta busca os dados desejados de uma sequência de maneira automatizada do banco de dados *GenBank*. Então ela faz um *Web Scraping* para extrair essa informação dos *site* do *GenBank*. Como ele está lidando com os textos em *HyperText Markup Language* (HTML) com as informações das sequências, a linguagem utilizada para desenvolver-lo foi o TypeScript, é uma linguagem fortemente tipada que se baseia no JavaScript desenvolvida e mantida pela Microsoft (Microsoft, 2012). No geral ela é um super conjunto que adiciona tipagem estática no

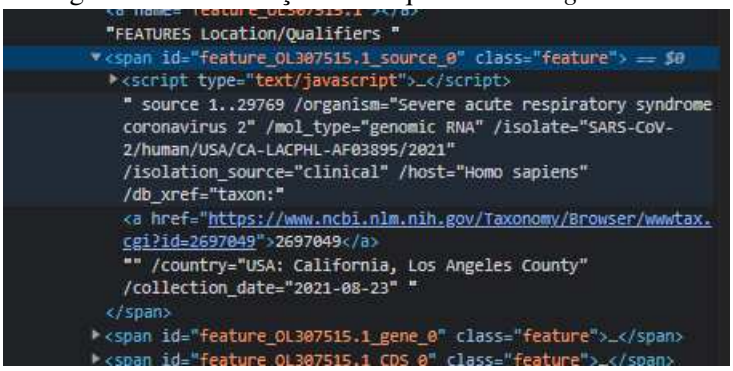
JavaScript.

O JavaScript é uma linguagem de alto nível que está de acordo com a especificação do ECMAScript (Ecma International, 2021). Ela é usada na construção de aplicações *web*, e possui várias bibliotecas de terceiros que ajudam no desenvolvimento.

A ferramenta foi desenvolvida da plataforma Node.js, essa plataforma foi criada para rodar códigos JavaScript fora de um navegador (OpenJS Foundation, 2009). Para a criação da ferramenta foi utilizado o *framework puppeteer*, esse *framework* oferece uma *Aplication Programming Interface* (API) de alto nível para controlar o navegador *Chrome*, dessa forma conseguimos navegar pelos *sites*.

A ferramenta lê o arquivo CSV recebido e cria uma lista com os identificadores das sequências, em seguida para cada identificador é buscado no *site* do GenBank as informações da sequência utilizando o *puppeteer*. Entendendo a estrutura HTML do *site* consegue-se utilizar as funções do *puppeteer* e coletar as informações gerais da sequência e o genoma nas *tags* HTML. Como mostrado na Figura 13 as informações das sequências ficam dentro de uma *tag* "*span*" com a classe "*feature*".

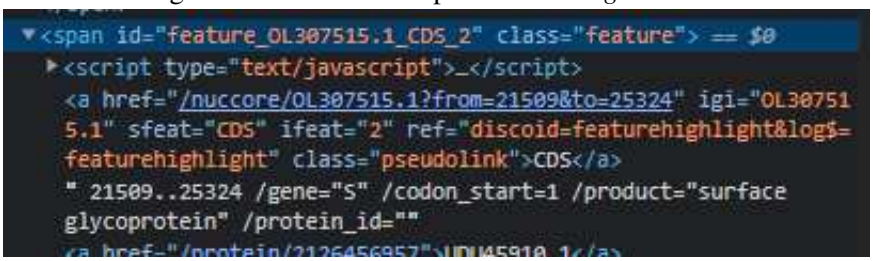
Figura 13 – Informações da sequência nas *tags* HTML.



Fonte – GenBank

As proteínas identificadas, assim como as informações da sequências, ficam dentro de uma *tag* "*span*" com a classe "*feature*" (Figura 14), as informações gerais são mostradas primeiro, depois são mostradas as proteínas identificadas na sequência na ordem em que aparecem na sequência, e no final é mostrada a sequência completa e também é mostrado o nome da proteína com o intervalo no qual ela está na sequência.

Figura 14 – Intervalo da proteína nas *tags* HTML.



Fonte – GenBank

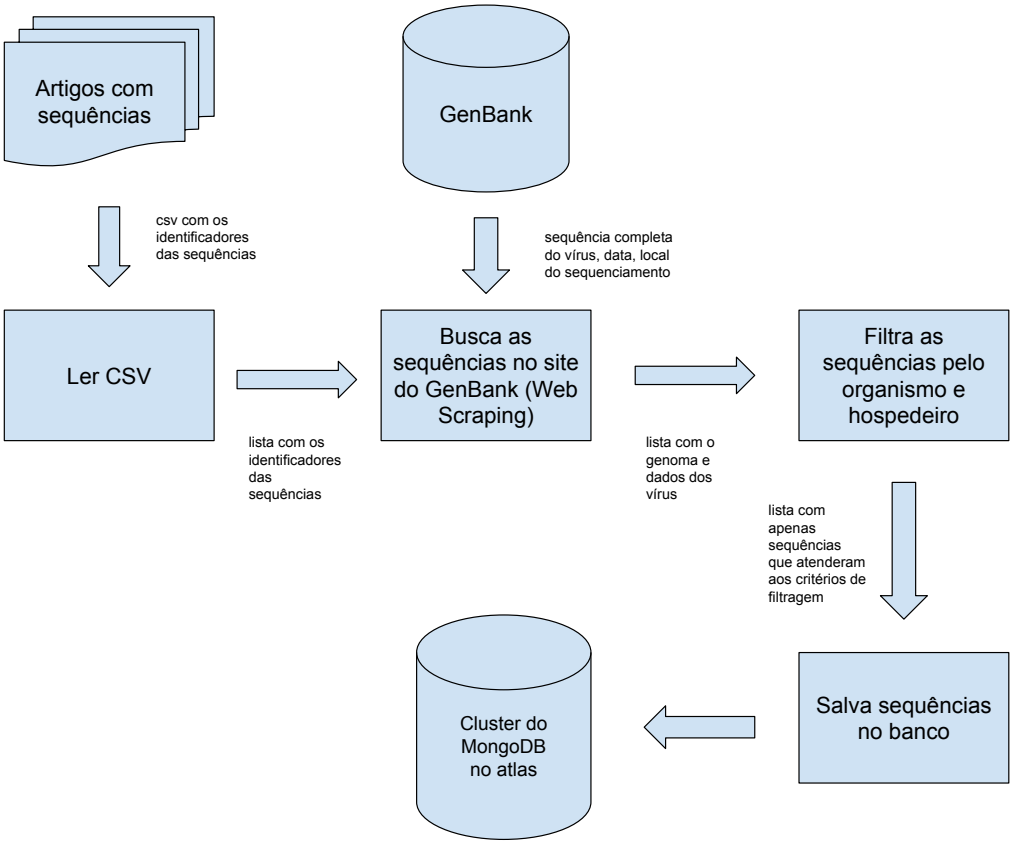
Como pode-se ver na Figura 15 a sequência completa fica separada por linhas dentro de *tags* "*spans*" com a classe "ff_line".

Figura 15 – Partes da sequência nas *tags* HTML.

Fonte – GenBank

Então são coletadas as informações gerais da sequência, seu genoma completo e a posição da proteína *spike* do *GenBank* pelo identificador. Depois de construir a lista com os dados coletados, são removidas as sequências que o hospedeiro não é "*homo sapiens*" e o organismo é diferente de "*severe acute respiratory syndrome coronavirus 2*", no armazenamento das sequências no banco as sequências ruins não são removidas. Esse processo de busca de sequência e armazenamento é ilustrado na figura 16.

Figura 16 – Processo de coleta de sequências.



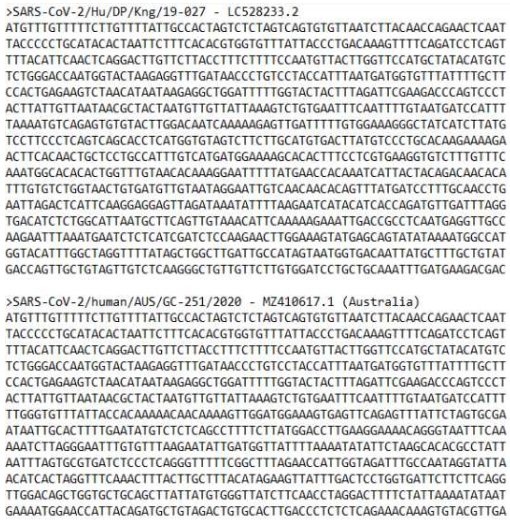
Fonte – Autor

4.3 GERAÇÃO DO CONJUNTO FASTA

Essa funcionalidade da ferramenta gera um conjunto FASTA com as sequências armazenadas no banco. São buscadas todas as sequências do banco, dessa lista de sequências são removidas as sequências que possuem um carácter diferente de A, T, G, ou C e não possuem a informação da localização da proteína *spike* no genoma. Com a lista das sequências ótimas as sequências da lista são cortadas na posição da proteína *spike* gerando uma nova lista apenas com a proteína e então um arquivo FASTA é construído.

Os arquivos FASTA são arquivo tipo baseados em texto usados para representar sequências de nucleotídeos ou aminoácidos. As linhas começam com um “>” (maior que) seguido de uma descrição para a sequência que vem a seguir, nas próximas linhas vêm a sequência até o próximo “>”, que indica outra sequência, na Figura 17 é mostrado um exemplo de um arquivo FASTA.

Figura 17 – Exemplo de arquivo FASTA



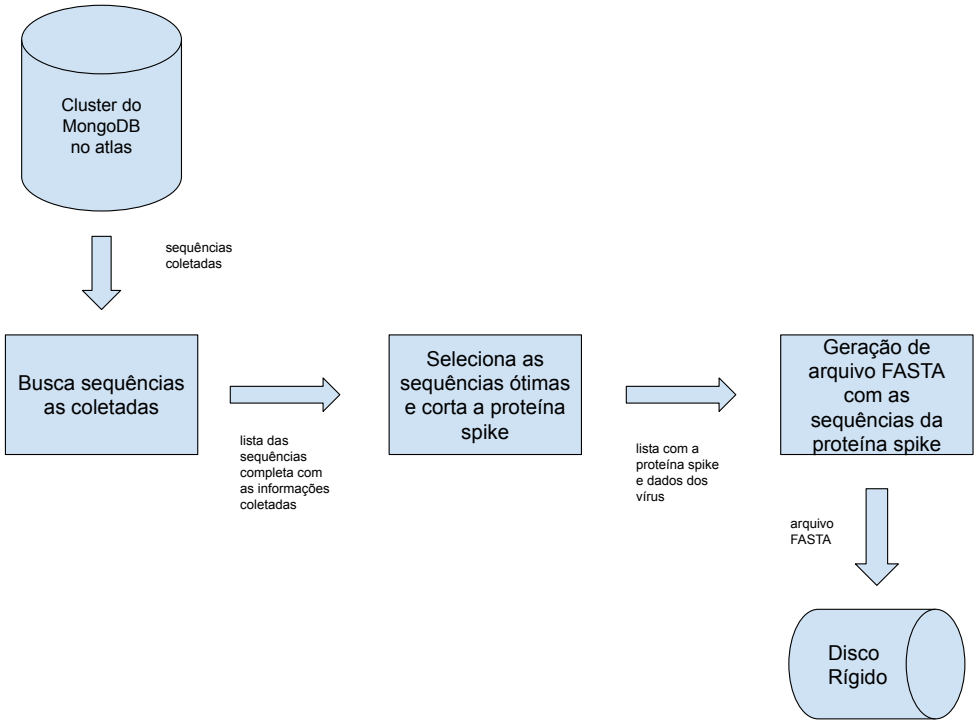
```
>SARS-CoV-2/Hu/DP/Kng/19-027 - LC528233.2
ATGTTTGTTTTTCTTGTTTTATTGCCACTAGTCTCTAGTCAGTGTGTTAATCTTACAACCAGAACTCAAT
TACCCCCTGCATACACTAATTCTTTCACACGTGGTGTTTATTACCCTGACAAAGTTTTCAGATCCTCAGT
TTTACATTCAACTCAGGACTTGTTCTTACCTTTCTTTTCCAATGTTACTTGGTTCCATGCTATACATGTC
TCTGGGACCAATGGTACTAAGAGGTTTGATAACCCTGTCCTACCATTTAATGATGGTGTTTATTTTGCTT
CCACTGAGAAGTCTAACATAATAAGAGGCTGGATTTTTGGTACTACTTTAGATTCGAAGACCCAGTCCCT
ACTTATTGTTAATAACGCTACTAATGTTGTTATTAAAGTCTGTGAATTTCAATTTTGTAATGATCCATTT
TAAAATGTCAGAGTGTGTACTTGGACAATCAAAAAGAGTTGATTTTTGTGGAAAGGGCTATCATCTTATG
TCCTTCCCTCAGTCAGCACCTCATGGTGTAGTCTTCTTGCATGTGACTTATGTCCCTGCACAAGAAAAGA
ACTTCACAACTGCTCCTGCCATTTGTCATGATGGAAAAGCACACTTTCCTCGTGAAGGTGTCTTTGTTTC
AAATGGCACACACTGGTTTGTAACACAAAGGAATTTTTATGAACCACAAATCATTACTACAGACAACACA
TTTGTGTCTGGTAACTGTGATGTTGTAATAGGAATTGTCAACAACACAGTTTATGATCCTTTGCAACCTG
AATTAGACTCATTCAAGGAGGAGTTAGATAAATATTTTAAGAATCATACATCACCAGATGTTGATTTAGG
TGACATCTCTGGCATTAATGCTTCAGTTGTAAACATTCAAAAAGAAATTGACCGCCTCAATGAGGTTGCC
AAGAATTTAAATGAATCTCTCATCGATCTCCAAGAACTTGGAAAGTATGAGCAGTATATAAAATGGCCAT
GGTACATTTGGCTAGGTTTTATAGCTGGCTTGATTGCCATAGTAATGGTGACAATTATGCTTTGCTGTAT
GACCAGTTGCTGTAGTTGTCTCAAGGGCTGTTGTTCTTGTGGATCCTGCTGCAAATTTGATGAAGACGAC

>SARS-CoV-2/human/AUS/GC-251/2020 - MZ410617.1 (Australia)
ATGTTTGTTTTTCTTGTTTTATTGCCACTAGTCTCTAGTCAGTGTGTTAATCTTACAACCAGAACTCAAT
TACCCCCTGCATACACTAATTCTTTCACACGTGGTGTTTATTACCCTGACAAAGTTTTCAGATCCTCAGT
TTTACATTCAACTCAGGACTTGTTCTTACCTTTCTTTTCCAATGTTACTTGGTTCCATGCTATACATGTC
TCTGGGACCAATGGTACTAAGAGGTTTGATAACCCTGTCCTACCATTTAATGATGGTGTTTATTTTGCTT
CCACTGAGAAGTCTAACATAATAAGAGGCTGGATTTTTGGTACTACTTTAGATTCGAAGACCCAGTCCCT
ACTTATTGTTAATAACGCTACTAATGTTGTTATTAAAGTCTGTGAATTTCAATTTTGTAATGATCCATTT
TTGGGTGTTTATTACCACAAAAACAACAAAAGTTGGATGGAAAGTGAGTTCAGAGTTTATTCTAGTGCGA
ATAATTGCACTTTTGAATATGTCTCTCAGCCTTTTCTTATGGACCTTGAAGGAAAACAGGGTAATTTCAA
AAATCTTAGGGAATTTGTGTTTAAGAATATTGATGGTTATTTTAAAATATATTCTAAGCACACGCCTATT
AATTTAGTGCGTGATCTCCCTCAGGGTTTTTCGGCTTTAGAACCATTGGTAGATTTGCCAATAGGTATTA
ACATCACTAGGTTTCAAACTTTACTTGCTTTACATAGAAGTTATTTGACTCCTGGTGATTCTTCTTCAGG
TTGGACAGCTGGTGCTGCAGCTTATTATGTGGGTTATCTTCAACCTAGGACTTTTCTATTAAAATATAAT
GAAAATGGAACCATTACAGATGCTGTAGACTGTGCACTTGACCCTCTCTCAGAAACAAAGTGTACGTTGA
```

Fonte – Autor

Então a ferramenta gera um arquivo FASTA com a lista filtrada, na descrição da sequência é colocado o nome identificador da sequência, o identificador da sequência no GenBank e o país onde a sequência foi isolada. O processo de geração do arquivo FASTA é ilustrado na Figura 18.

Figura 18 – Processo de geração do arquivo FASTA



Fonte – Autor

4.4 CBUC NO PYTHON

A implementação foi feita no ambiente Google Colab em Python 3(Google, 2015), ele oferece uma notebook na nuvem em uma plataforma Jupyter Notebook (Jupyter Project, 2014), com quantidades altíssimas de memória principal e disco e de fácil compartilhamento. Assim é possível desenvolver em Python pelo navegador utilizando uma máquina com alta capacidade computacional.

O programa recebe o arquivo no formato FASTA faz a leitura das sequência no arquivo. Com a lista das sequências ele transforma as sequências em uma lista de números normalizada, são selecionadas as trincas de caracteres da sequência e transforma em um número de 2 a 65 que representa um códon.

Com a lista de sequências normalizadas é criada a base de conhecimento, e calculada a matriz de frequência dos códons das sequências. A partir da matriz de frequência são criados os agrupamentos, e é gerado o mapa de agrupamentos e calculada a quantidade de agrupamentos em cada posição da sequência. Pelo agrupamento de cada códon da sequência é gerado o código identificador único de cada sequência. A frequência de cada padrão é calculada, com a frequência de cada padrão calculada é gerada a matriz de distância, e com a matriz de distância dos padrões são feitos os agrupamentos em famílias dos padrões. Todos esses cálculos são feitos com vetores da biblioteca *numpy*.

Os resultados dos agrupamentos de códons e os agrupamentos dos padrões de códons, as famílias, são exibidos no console utilizando a biblioteca *matplotlib*.

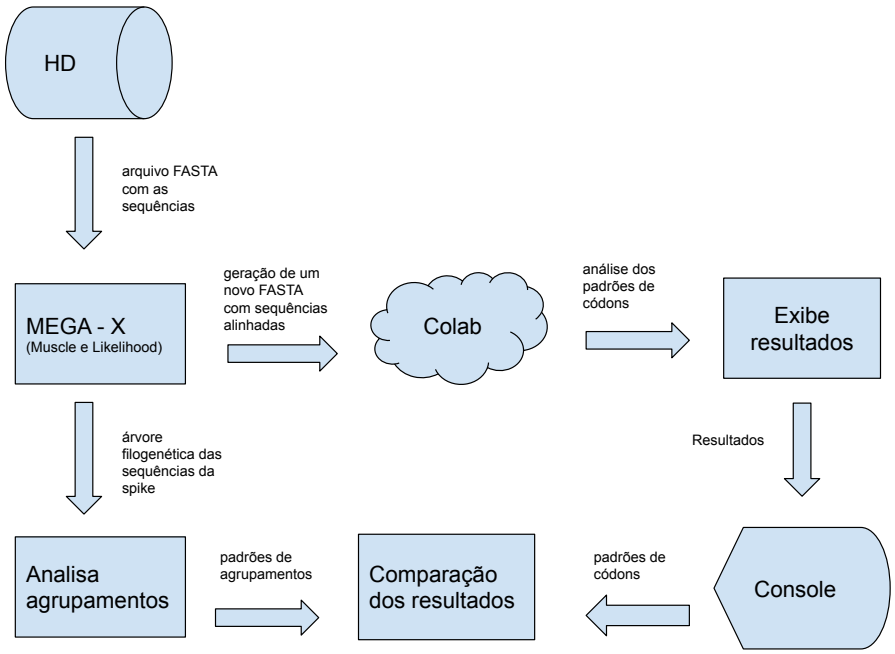
5 RESULTADOS

Ao final do processo de busca e armazenamento das sequências dos artigos selecionados, foram coletadas com a ferramenta 3.705 sequências do Sars-Cov-2. Para aumentar a diversidade do conjunto de dados foram selecionadas manualmente sequências ótimas do Sars-Cov-2 do GenBank de países de não estavam no banco no final foram selecionadas 152 sequências. Então no final o banco ficou com 3857 para gerar o conjunto de dados.

Foram selecionadas no máximo 10 de cada país para manter a variação de local de cada sequência e evitar que apareçam sequências repetidas. Ao final foi gerado um arquivo FASTA com 310 sequências que foi alinhado no MEGA e analisado pelo CBUC.

A árvore filogenética foi gerada utilizando o algoritmo *Maximum Likelihood* com 500 replicações de *bootstrap* (HUELSENBECK; CRANDALL, 1997), com o arquivo FASTA de 310 sequências. O processo de análise dos resultados é ilustrado na figura 19.

Figura 19 – Processo de análise das sequências.

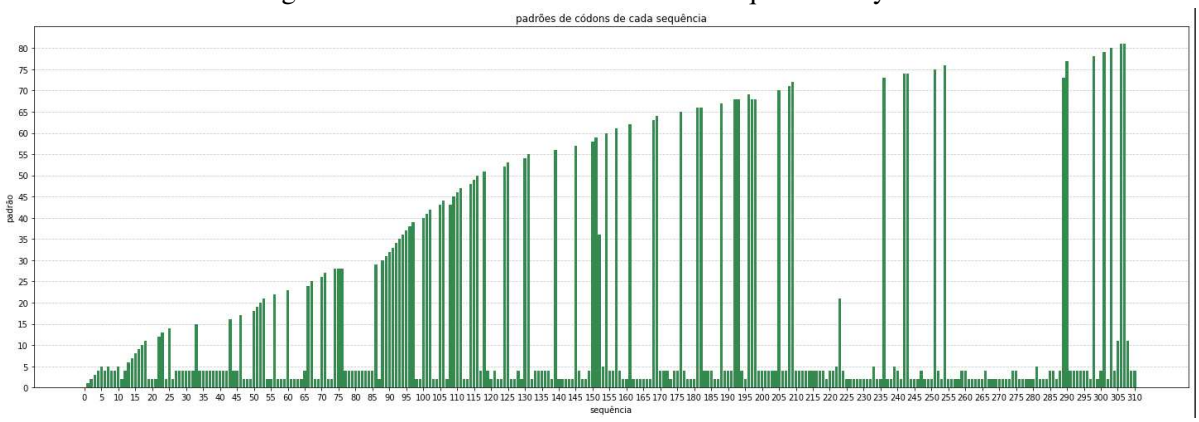


Fonte – Autor

5.1 RESULTADOS DO CBUC NO PYTHON

O algoritmo no Python aplicado ao conjunto de 310 sequências identificou 81 padrões de códons diferentes, na Figura 20 pode-se visualizar o gráfico informando o padrão de cada sequência, as sequências estão numeradas de 1 a 310 no eixo x e os padrões no eixo y.

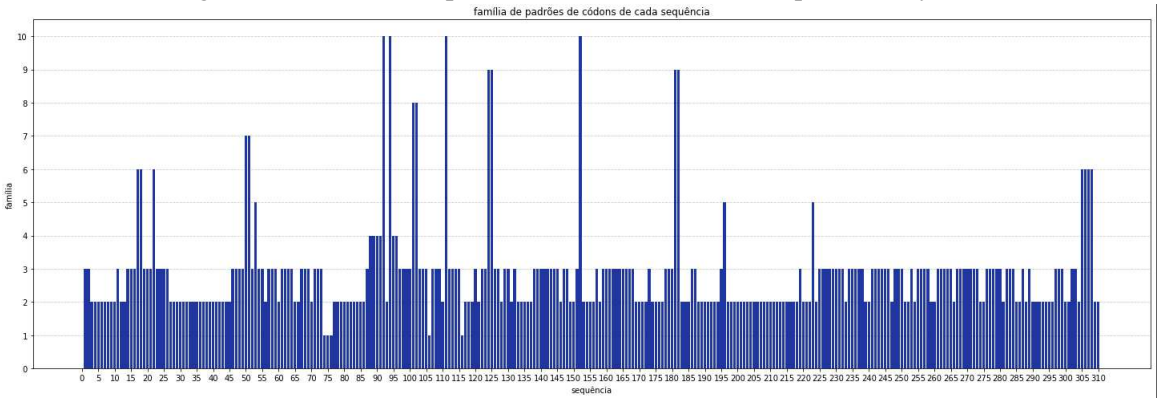
Figura 20 – Padrões de códons de cada sequência - Python



Fonte – Autor

Fazendo o agrupamento dos padrões, o programa conseguiu encontrar 10 famílias de padrões conforme visto o gráfico na Figura 21. A maioria das sequências se agruparam nas famílias 2 e 3. Foram selecionadas as 29 sequências pertencentes às famílias 1, 4, 5, 6, 7, 8, 9 e 10 (Tabela 1), e essas sequências foram identificadas em cores distintas na árvore filogenética gerada.

Figura 21 – Família de padrões de códons de cada sequência - Python



Fonte – Autor

Na Figura 22 é possível ver as marcações de duas das famílias identificadas, a família 1 (cor vermelho vinho) e a família 5 (cor laranja). Na Figura 23 pode-se ver na árvore as famílias 1 (cor vermelho vinho), família 4 (cor vermelha), família 6 (cor verde), família 7 (cor azul fraco),

Tabela 1 – Sequências das famílias 1, 4, 5, 6, 7, 8, 9 e 10 - Python.

No	Família	Identificador	País
74	1	SARS-CoV-2/human/GUM/GU-NHG-01/2020	Guam
104	1	SARS-CoV-2/human/IND/GBRC20/2020	India: Ahmedabad
106	1	SARS-CoV-2/human/IND/GBRC24a/2020	India: Ahmedabad
116	1	SARS-CoV-2/human/IND/GBRC16/2020	India: Surat
88	4	SARS-CoV-2/human/IND/509MN908947.3/2021	India
89	4	SARS-CoV-2/human/IND/528MN908947.3/2021	India
90	4	SARS-CoV-2/human/IND/565MN908947.3/2021	India
91	4	SARS-CoV-2/human/IND/Wuhan/2021	India
95	4	SARS-CoV-2/human/IND/510MN908947.3/2021	India
96	4	SARS-CoV-2/human/IND/512MN908947.3/2021	India
53	5	nCoV-FIN-29-Jan-2020	Finland
196	5	SARS-CoV-2/human/TWN/CGMH-CGU-02/2020	Taiwan
17	6	SARS-CoV-2/human/BRA/3048/2021	Brazil
18	6	SARS-CoV-2/human/BRA/3918/2021	Brazil
22	6	SARS-CoV-2/human/BRA/CD1739-P4/2020	Brazil: Manaus
305	6	SARS-CoV-2/human/URY/374/2021	Uruguay
306	6	SARS-CoV-2/human/VEN/Bol4715/2021	Venezuela
308	6	SARS-CoV-2/human/VEN/Mir6242/2021	Venezuela
50	7	SARS-CoV-2/human/CZE/CzechiaMotol 2448/2020	Czech Republic
51	7	SARS-CoV-2/human/CZE/CzechiaMotol 2174/2020	Czech Republic
101	8	SARS-CoV-2/human/IND/GBRC17a/2020	India: Ahmedabad
102	8	SARS-CoV-2/human/IND/GBRC17b/2020	India: Ahmedabad
108	8	SARS-CoV-2/human/IND/GBRC55/2020	India: Dhansura
124	9	SARS-CoV-2/human/ITA/TE.237004/2020	Italy
125	9	SARS-CoV-2/human/ITA/Pango Lineage B.1.177	Italy
181	9	SARS-CoV-2/human/ESP/HUD-66682640/2020	Spain
92	10	SARS-CoV-2/human/IND/GBRC648/2021	India
94	10	SARS-CoV-2/human/IND/hospital 2 295/2021	India
111	10	SARS-CoV-2/human/IND/AIIMS-Bhopal-S...	India: Madhya P...

Fonte – Autor

família 8 (cor azul forte), família 9 (cor azul lilás) e família 10 (cor azul rosa). A sequência da família 1 ficou isolada dos agrupamentos das famílias, e os restantes dos agrupamentos encontrados coincidiram com a árvore.

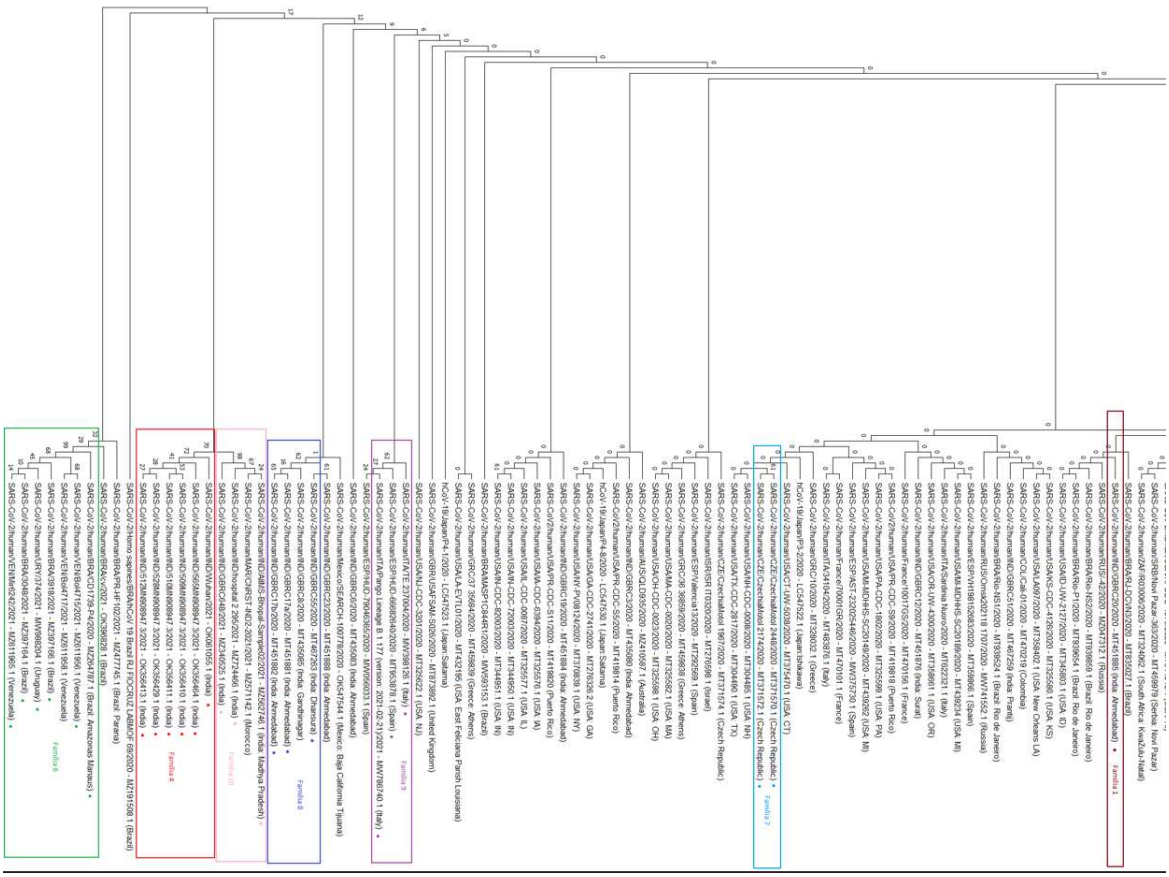
5.2 RESULTADOS DO CBUC NO SCILAB

A implementação no Scilab utilizando as 310 sequências e também identificou 81 padrões de códons diferentes, na Figura 24 pode-se visualizar o gráfico informando o padrão de cada sequência.

Figura 22 – Famílias 1 e 5 identificadas na árvore - Python

Fonte – Autor

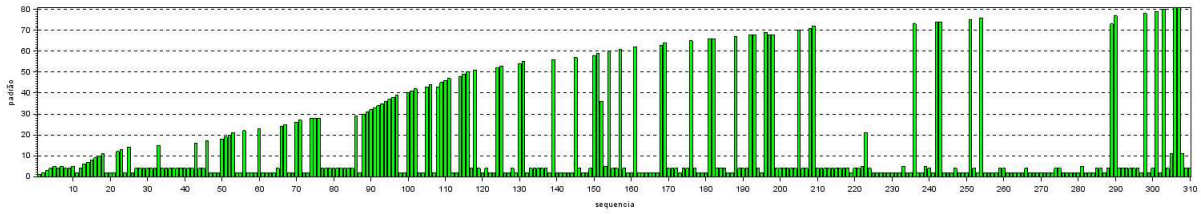
Figura 23 – Famílias famílias 1, 4, 5, 6, 7, 8, 9 e 10 identificadas na árvore - Python



Fonte – Autor

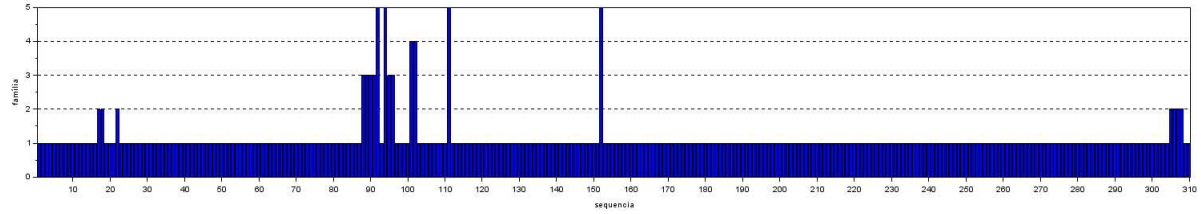
Fazendo o agrupamento dos padrões no Scilab o programa conseguiu encontrar 5 famílias de padrões que podem ser vistas no gráfico na Figura 25. A maioria das sequências se agruparam na família 1. Foram selecionadas as 16 sequências pertencentes às famílias 2, 3, 4 e 5 (Tabela 2 e Tabela 3).

Figura 24 – Padrões de códons de cada sequência - Scilab



Fonte – Autor

Figura 25 – Família de padrões de códons de cada sequência



Fonte – Autor

Tabela 2 – Sequências das famílias 2, 3 e 4 - Scilab.

No	Família	Identificador	País
17	2	SARS-CoV-2/human/BRA/3048/2021	Brazil
18	2	SARS-CoV-2/human/BRA/3918/2021	Brazil
305	2	SARS-CoV-2/human/URY/374/2021	Uruguay
306	2	SARS-CoV-2/human/VEN/Bol4715/2021	Venezuela
307	2	SARS-CoV-2/human/VEN/Bol4717/2021	Venezuela
308	2	SARS-CoV-2/human/VEN/Mir6242/2021	Venezuela
88	3	SARS-CoV-2/human/IND/509MN908947.3/2021	India
89	3	SARS-CoV-2/human/IND/528MN908947.3/2021	India
90	3	SARS-CoV-2/human/IND/565MN908947.3/2021	India
91	3	SARS-CoV-2/human/IND/Wuhan/2021	India
96	3	SARS-CoV-2/human/IND/512MN908947.3/2021	India
97	3	SARS-CoV-2/human/IND/GBRC62/2020	India: Ahmedabad
7	4	SARS-CoV-2/human/IND/GBRC17a/2020	India: Ahmedabad
8	4	SARS-CoV-2/human/IND/GBRC17b/2020	India: Ahmedabad

Fonte – Autor

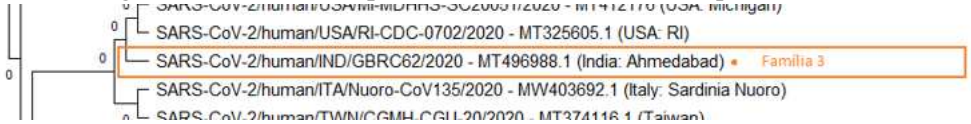
Na Figura 26 pode-se ver marcada a família 3 (cor laranja) com apenas uma sequência isolada. Na Figura 27 pode-se ver as famílias 2 (cor vermelho), família 3 (cor laranja), família 4 (cor verde), família 5 (cor azul fraco). Apenas a sequência da família 3 teve uma sequência que ficou isolada dos agrupamentos das famílias, e o restante dos agrupamentos encontrados coincidiram com a árvore.

Tabela 3 – Sequências da família 5 - Scilab.

No	Família	Identificador	País
92	5	SARS-CoV-2/human/IND/GBRC648/2021	India
94	5	SARS-CoV-2/human/IND/hospital 2 295/2021	India
111	5	SARS-CoV-2/human/IND/AIIMS-Bhopal-Sample...	India: Madhya P...
152	5	SARS-CoV-2/human/MAR/CNRST-IND2-2021...	Morocco

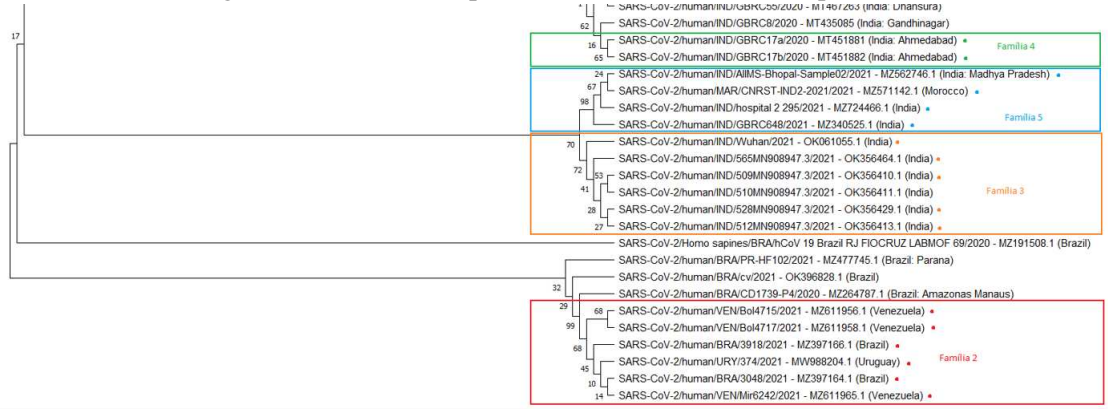
Fonte – Autor

Figura 26 – Família de padrões de códons de cada sequência



Fonte – Autor

Figura 27 – Família de padrões de códons de cada sequência



Fonte – Autor

5.3 RESULTADOS COM UM CONJUNTO DE SEQUÊNCIAS REDUZIDO

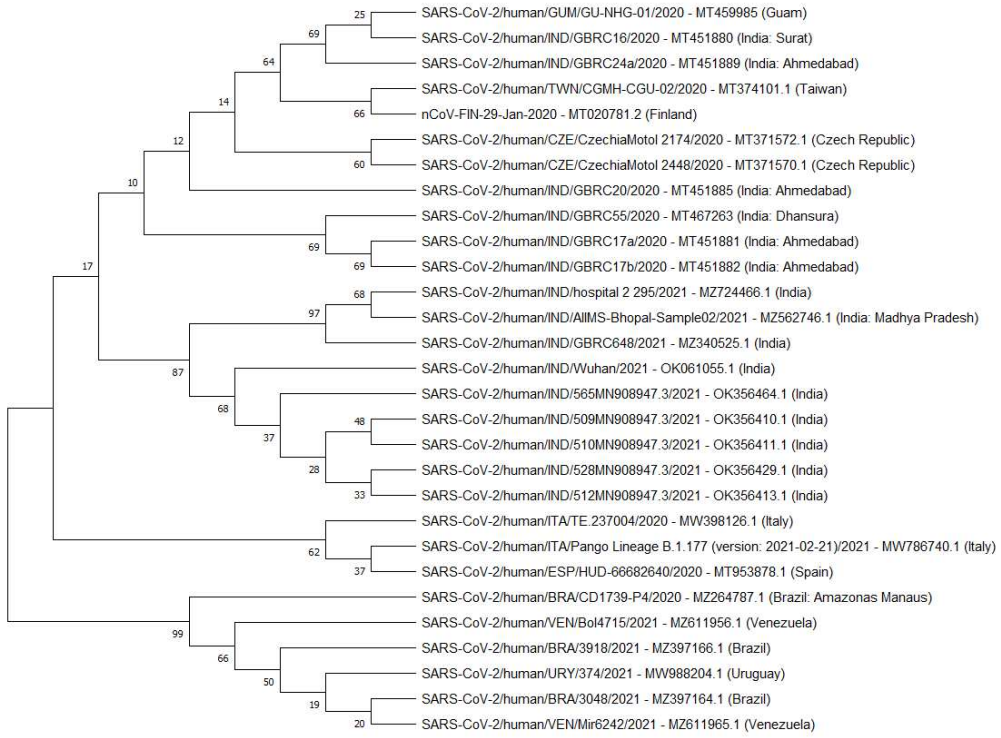
Para conseguir visualizar melhor as famílias de padrões e as famílias identificadas na árvore, foram selecionas as sequências das famílias 1, 4, 5, 6, 7, 8, 9 e 10 (Tabela 1), encontradas na análise do Python. Esse conjunto reduzido foi analisado com a implementação do Python e do Scilab. Foi gerada uma nova árvore filogenética com 500 replicações de *bootstrap* com esse conjunto reduzido apresentada na figura 28.

5.3.1 Resultados do Python com um conjunto de sequências reduzido

A implementação com o conjunto reduzido identificou 27 padrões de códons diferentes (Figura 29).

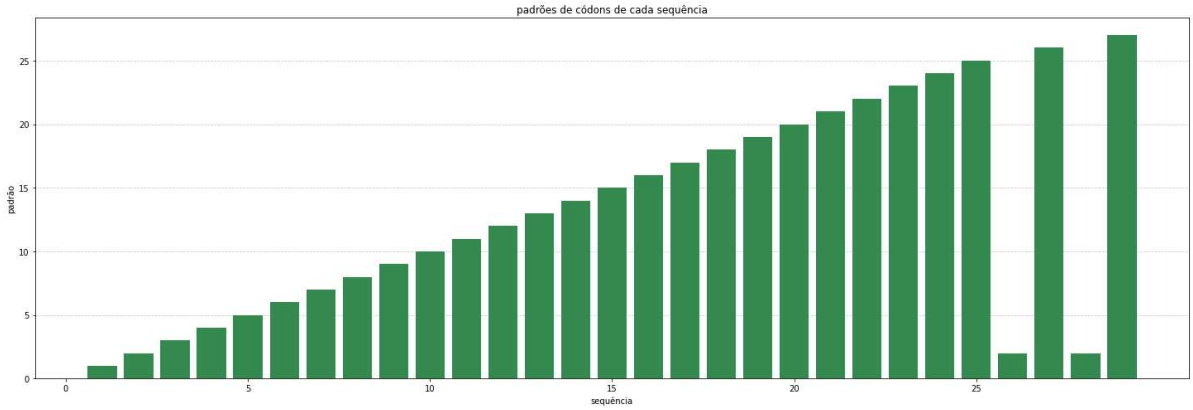
No agrupamento dos padrões foram encontradas 7 (Figura 30). Na Tabela 4, pode-se

Figura 28 – Padrões de códons de cada sequência - Python



Fonte – Autor

Figura 29 – Padrões de códons de cada sequência (conjunto reduzido) - Python

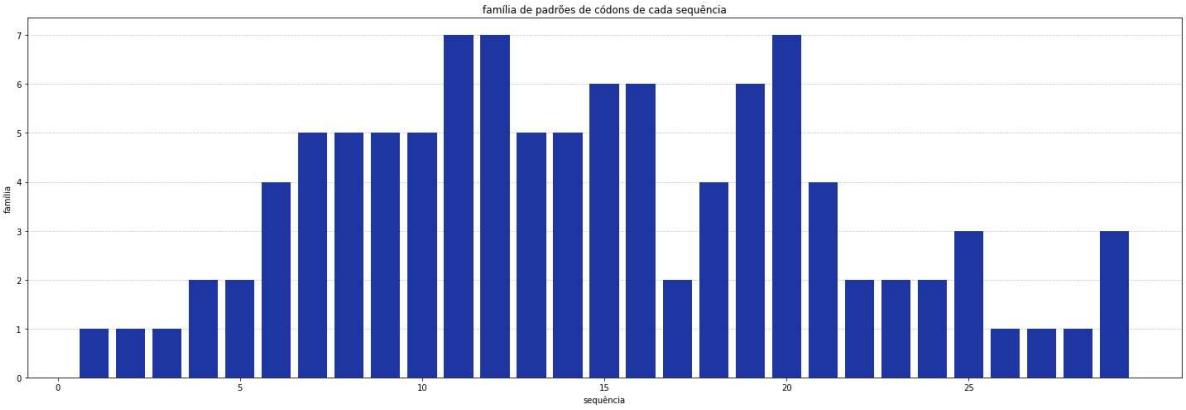


Fonte – Autor

ver as sequências que se agruparam com mais detalhes.

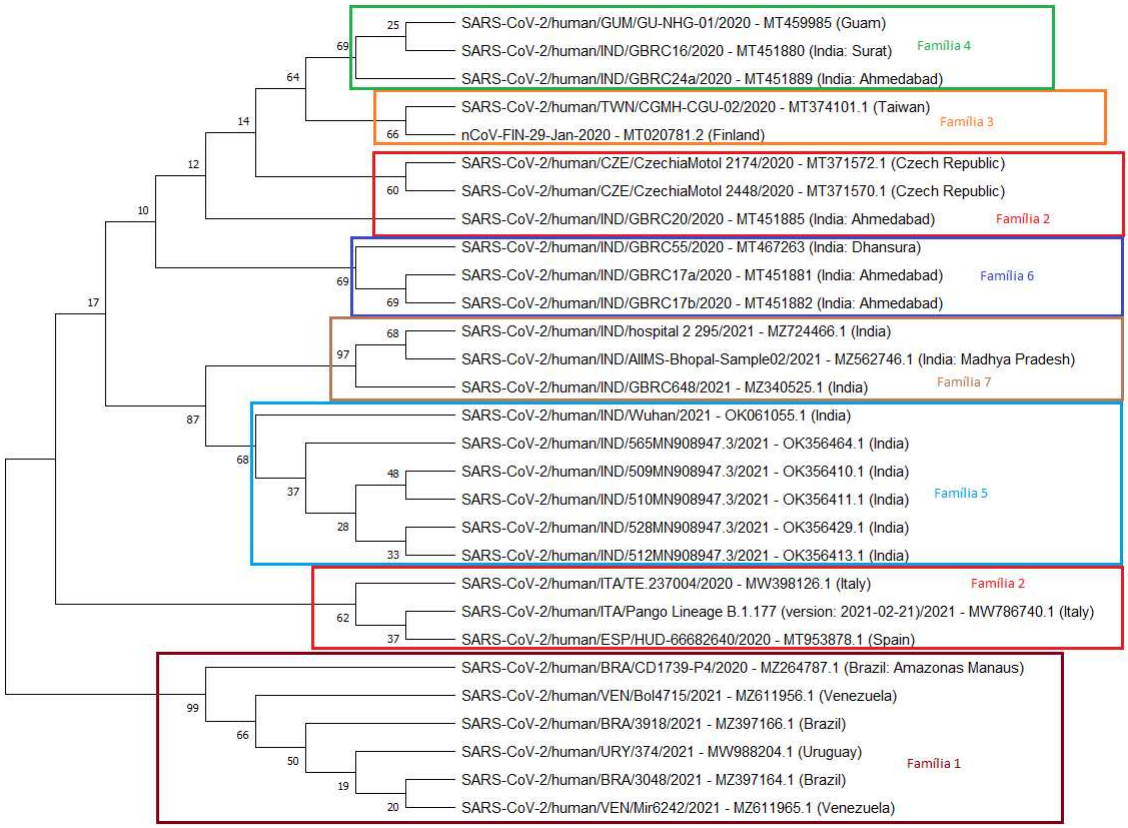
Na Figura 31 pode-se ver claramente todas as famílias marcadas, os agrupamentos encontrados coincidiram com a árvore, apenas 3 sequências da família 2 ficaram em outro clado.

Figura 30 – Família de padrões de códons de cada sequência (conjunto reduzido) - Python



Fonte – Autor

Figura 31 – Famílias Identificadas (conjunto reduzido) - Python



Fonte – Autor

5.3.2 Resultados do Scilab com um conjunto de sequências reduzido

Também foram encontrados 27 padrões de códons diferentes (Figura 32). Com esse conjunto de sequências os agrupamentos em famílias foram iguais ao do Python, foram encontrados 7 (Figura 33). Na Tabela 5, pode-se ver as sequências que se agruparam com mais detalhes.

Tabela 4 – Sequências das famílias 1 a 7 (conjunto reduzido) - Python.

No	Família	Identificador	País
1	1	SARS-CoV-2/human/BRA/3048/2021	Brazil
2	1	SARS-CoV-2/human/BRA/3918/2021	Brazil
3	1	SARS-CoV-2/human/BRA/CD1739-P4/2020	Brazil: Manaus
26	1	SARS-CoV-2/human/URY/374/2021	Uruguay
27	1	SARS-CoV-2/human/VEN/Bol4715/2021	Venezuela
28	1	SARS-CoV-2/human/VEN/Mir6242/2021	Venezuela
4	2	SARS-CoV-2/human/CZE/CzechiaMotol 2448/2020	Czech Republic
5	2	SARS-CoV-2/human/CZE/CzechiaMotol 2174/2020	Czech Republic
17	2	SARS-CoV-2/human/IND/GBRC20/2020	India: Ahmedabad
22	2	SARS-CoV-2/human/ITA/TE.237004/2020	Italy
23	2	SARS-CoV-2/human/ITA/Pango Lineage B.1.177/...	Italy
24	2	SARS-CoV-2/human/ESP/HUD-66682640/2020	Spain
25	3	nCoV-FIN-29-Jan-2020	Finland
29	3	SARS-CoV-2/human/TWN/CGMH-CGU-02/2020	Taiwan
6	4	SARS-CoV-2/human/GUM/GU-NHG-01/2020	Guam
18	4	SARS-CoV-2/human/IND/GBRC24a/2020	India: Ahmedabad
21	4	SARS-CoV-2/human/IND/GBRC16/2020	India: Surat
7	5	SARS-CoV-2/human/IND/509MN908947.3/2021	India
8	5	SARS-CoV-2/human/IND/528MN908947.3/2021	India
9	5	SARS-CoV-2/human/IND/565MN908947.3/2021	India
10	5	SARS-CoV-2/human/IND/Wuhan/2021	India
13	5	SARS-CoV-2/human/IND/510MN908947.3/2021	India
14	5	SARS-CoV-2/human/IND/512MN908947.3/2021	India
15	6	SARS-CoV-2/human/IND/GBRC17a/2020	India: Ahmedabad
16	6	SARS-CoV-2/human/IND/GBRC17b/2020	India: Ahmedabad
19	6	SARS-CoV-2/human/IND/GBRC55/2020	India: Dhansura
11	7	SARS-CoV-2/human/IND/GBRC648/2021	India
12	7	SARS-CoV-2/human/IND/hospital 2 295/2021	India
20	7	SARS-CoV-2/human/IND/AIIMS-Bhopal-Sample02...	India: Madhya P...

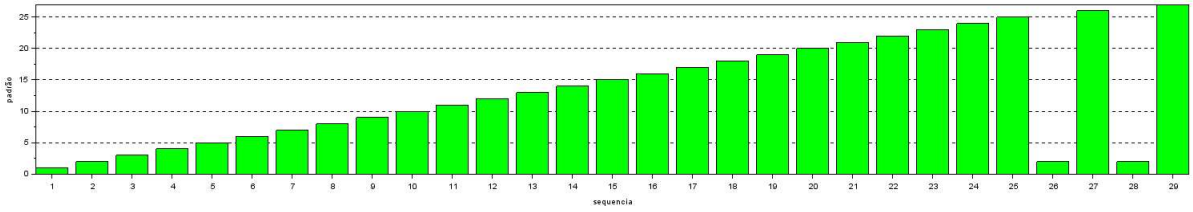
Fonte – Autor

Na Figura 34 pode-se ver as famílias marcadas, como os resultados foram iguais ao do Python os agrupamentos encontrados também coincidiram com a árvore, e a família 2 que é igual a família 3 do Python ficou com três sequências em um clado separado também.

5.4 RESULTADOS DO PYTHON X RESULTADOS DO SCILAB

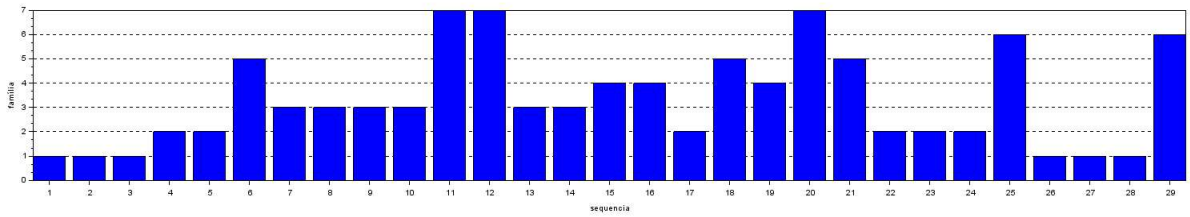
Com as 310 sequências a implementação em Python encontrou 5 famílias de padrões de códons a mais que o Scilab e as famílias encontradas a mais no Python foram coerentes com a árvore filogenética. Essa diferença acontece na ordenação na matriz de distâncias, quando

Figura 32 – Padrões de códons de cada sequência (conjunto reduzido) - Scilab



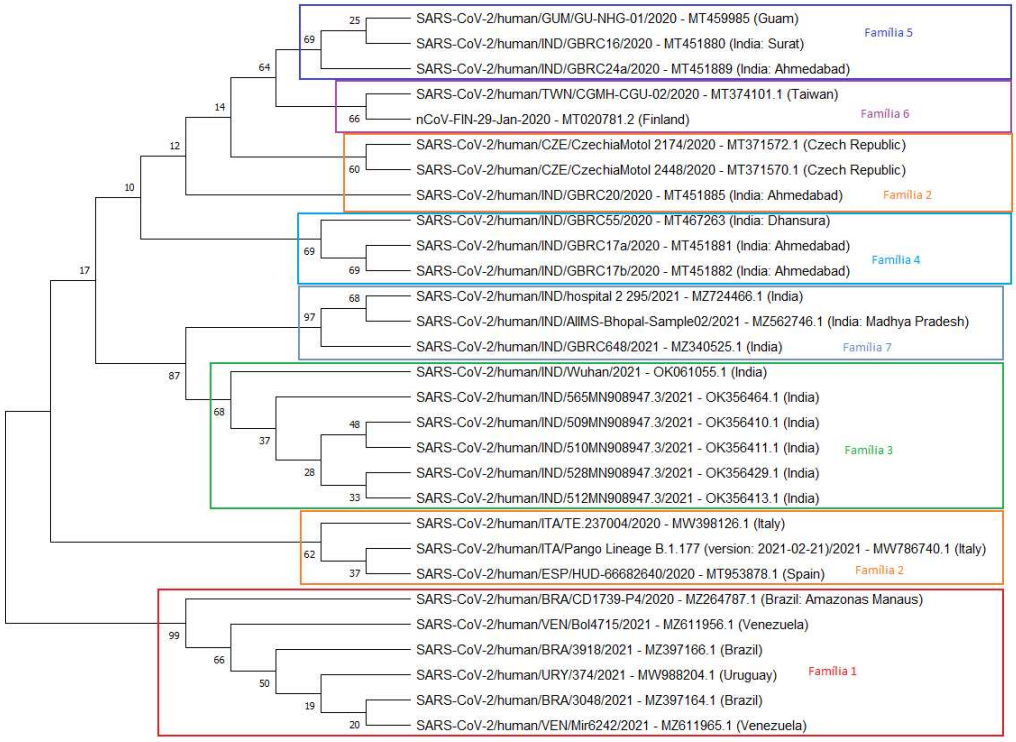
Fonte – Autor

Figura 33 – Família de padrões de códons de cada sequência (conjunto reduzido) - Scilab



Fonte – Autor

Figura 34 – Famílias Identificadas (conjunto reduzido) - Scilab



Fonte – Autor

existem duas distâncias iguais o Python retorna o índice da primeira que ele encontra na matriz, o Scilab retorna um índice diferente. E isso gerou essa diferença no número de família, porque as famílias são agrupadas com base nos índices da matriz de distâncias, e um índice retornado primeiro pode fazer com uma família não seja agrupada.

Tabela 5 – Sequências das famílias 1 a 7 (conjunto reduzido) - Scilab.

No	Família	Identificador	País
1	1	SARS-CoV-2/human/BRA/3048/2021	Brazil
2	1	SARS-CoV-2/human/BRA/3918/2021	Brazil
3	1	SARS-CoV-2/human/BRA/CD1739-P4/2020	Brazil: Manaus
26	1	SARS-CoV-2/human/URY/374/2021	Uruguay
27	1	SARS-CoV-2/human/VEN/Bol4715/2021	Venezuela
28	1	SARS-CoV-2/human/VEN/Mir6242/2021	Venezuela
4	2	SARS-CoV-2/human/CZE/CzechiaMotol 2448/2020	Czech Republic
5	2	SARS-CoV-2/human/CZE/CzechiaMotol 2174/2020	Czech Republic
17	2	SARS-CoV-2/human/IND/GBRC20/2020	India: Ahmedabad
22	2	SARS-CoV-2/human/ITA/TE.237004/2020	Italy
23	2	SARS-CoV-2/human/ITA/Pango Lineage B.1.177/...	Italy
24	2	SARS-CoV-2/human/ESP/HUD-66682640/2020	Spain
7	3	SARS-CoV-2/human/IND/509MN908947.3/2021	India
8	3	SARS-CoV-2/human/IND/528MN908947.3/2021	India
9	3	SARS-CoV-2/human/IND/565MN908947.3/2021	India
10	3	SARS-CoV-2/human/IND/Wuhan/2021	India
13	3	SARS-CoV-2/human/IND/510MN908947.3/2021	India
14	3	SARS-CoV-2/human/IND/512MN908947.3/2021	India
15	4	SARS-CoV-2/human/IND/GBRC17a/2020	India: Ahmedabad
16	4	SARS-CoV-2/human/IND/GBRC17b/2020	India: Ahmedabad
19	4	SARS-CoV-2/human/IND/GBRC55/2020	India: Dhansura
21	5	SARS-CoV-2/human/IND/GBRC16/2020	India: Surat
6	5	SARS-CoV-2/human/GUM/GU-NHG-01/2020	Guam
18	5	SARS-CoV-2/human/IND/GBRC24a/2020	India: Ahmedabad
25	6	nCoV-FIN-29-Jan-2020	Finland
29	6	SARS-CoV-2/human/TWN/CGMH-CGU-02/2020	Taiwan
11	7	SARS-CoV-2/human/IND/GBRC648/2021	India
12	7	SARS-CoV-2/human/IND/hospital 2 295/2021	India
20	7	SARS-CoV-2/human/IND/AIIMS-Bhopal-Sample02...	India: Madhya P...

Fonte – Autor

6 CONSIDERAÇÕES FINAIS

A implementação em Python conseguiu gerar agrupamentos em famílias, e os agrupamento foram coerentes com a árvore filogenética gerada com 500 replicações de *bootstrap*. Foi observado que os agrupamentos encontrados tiveram um *bootstrap score* maior que 60, a maioria dos agrupamentos que não foram encontrados da árvore tinha *bootstrap score* 0 o que não garante uma confiabilidade daqueles clados formados.

O buscador de sequências funcionou muito bem, coletando e armazenando as sequências do Sars-Cov-2. A maneira como foi implementado pode ser facilmente adaptado para coletar sequências de outros organismos.

A implementação em Python gera os resultados em um período de tempo de aproximadamente 50 segundos, o que é um resultado muito bom. E também consegue identificar mais famílias de padrões de códons do que a implementação no Scilab.

Dado fato que o Sars-Cov-2 é um vírus recente e ainda não possui grandes varições entre suas sequências, tivemos resultados positivos com os agrupamentos que foram encontrados, eles estavam coerentes com a árvore gerada, podemos dizer que o algoritmo CBUC é no mínimo promissor e pode ser usado para classificar grupos monofiléticos em tempo hábil.

A partir desse trabalho abrem-se várias possibilidades de trabalhos futuros. Podem ser feitos estudos com outros organismos, pode ser feito o trabalho para deixar o CBUC online, pode unificar o CBUC online com o buscador de sequências entre outros projetos.

REFERÊNCIAS

ALBERTS, B.; JOHNSON, A.; LEWIS, J.; MORGAN, D.; RAFF, M.; ROBERTS, K.; WALTER, P.; WILSON, J.; HUNT, T. **Biologia Molecular da Célula - 6ª Ed.** [S.l.]: Wolters Kluwer Health, 2017. ISBN 9788582714232.

BAUM, D. A.; SMITH., S. D. **Tree thinking : an introduction to phylogenetic biology**. [S.l.]: Roberts, 2013. ISBN 1936221160,9781936221165.

BAYAT, A. Science, medicine, and the future: Bioinformatics. **BMJ (Clinical research ed.)**, BMJ, v. 324, n. 7344, p. 1018–1022, Abr 2002. ISSN 1756-1833. 11976246[pmid]. Disponível em: <https://pubmed.ncbi.nlm.nih.gov/11976246>.

CRESWELL, J. **Projeto de pesquisa métodos qualitativo, quantitativo e misto**. [S.l.]: Artmed, 2007. ISBN 9788536308920.

CRICK, F. H. C.; BARNETT, L.; BRENNER, S.; WATTS-TOBIN, R. J. General nature of the genetic code for proteins. **Nature**, v. 192, n. 4809, p. 1227–1232, Dez 1961. ISSN 1476-4687. Disponível em: <https://doi.org/10.1038/1921227a0>.

CUI, J.; LI, F.; SHI, Z.-L. Origin and evolution of pathogenic coronaviruses. **Nature Reviews Microbiology**, v. 17, n. 3, p. 181–192, Mar 2019. ISSN 1740-1534. Disponível em: <https://doi.org/10.1038/s41579-018-0118-9>.

DROSTEN, C.; GüNTHER, S.; PREISER, W.; WERF, S. van der; BRODT, H.-R.; BECKER, S.; RABENAU, H.; PANNING, M.; KOLESNIKOVA, L.; FOUCHIER, R. A.; BERGER, A.; BURGUIèRE, A.-M.; CINATL, J.; EICKMANN, M.; ESCRIOU, N.; GRYWNA, K.; KRAMME, S.; MANUGUERRA, J.-C.; MüLLER, S.; RICKERTS, V.; STüRMER, M.; VIETH, S.; KLENK, H.-D.; OSTERHAUS, A. D.; SCHMITZ, H.; DOERR, H. W. Identification of a novel coronavirus in patients with severe acute respiratory syndrome. **New England Journal of Medicine**, v. 348, n. 20, p. 1967–1976, 2003. PMID: 12690091. Disponível em: <https://doi.org/10.1056/NEJMoa030747>.

Ecma International. **Especificação da linguagem ECMAScript 2022**. 2021. Disponível em: <https://tc39.es/ecma262/>. Acesso em: 16 de outubro de 2021.

EDGAR, R. C. MUSCLE: multiple sequence alignment with high accuracy and high throughput. **Nucleic Acids Research**, v. 32, n. 5, p. 1792–1797, 03 2004. ISSN 0305-1048. Disponível em: <https://doi.org/10.1093/nar/gkh340>.

Google. **Google Colab**. 2015. Disponível em: <https://colab.research.google.com>. Acesso em: 16 de outubro de 2021.

GRIEVES, M.; VICKERS, J. Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In: ____. **Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches**. Cham: Springer International Publishing, 2017. p. 85–113. ISBN 978-3-319-38756-7. Disponível em: <https://doi.org/10.1007/978-3-319-38756-7_4>.

GROSJEAN, H.; FIERS, W. Preferential codon usage in prokaryotic genes: the optimal codon-anticodon interaction energy and the selective codon usage in efficiently expressed genes. **Gene**, v. 18, n. 3, p. 199–209, 1982. ISSN 0378-1119. Disponível em: <https://www.sciencedirect.com/science/article/pii/0378111982901573>.

HU, B.; GUO, H.; ZHOU, P.; SHI, Z.-L. Characteristics of sars-cov-2 and covid-19. **Nature reviews. Microbiology**, Nature Publishing Group UK, v. 19, n. 3, p. 141–154, Mar 2021. ISSN 1740-1534. 33024307[pmid]. Disponível em: <https://pubmed.ncbi.nlm.nih.gov/33024307>.

HUELSENBECK, J. P.; CRANDALL, K. A. Phylogeny estimation and hypothesis testing using maximum likelihood. **Annual Review of Ecology and Systematics**, v. 28, n. 1, p. 437–466, 1997. Disponível em: <https://doi.org/10.1146/annurev.ecolsys.28.1.437>.

IKEMURA, T. Codon usage and tRNA content in unicellular and multicellular organisms. **Molecular Biology and Evolution**, v. 2, n. 1, p. 13–34, Jan 1985. ISSN 0737-4038. Disponível em: <https://doi.org/10.1093/oxfordjournals.molbev.a040335>.

Jupyter Project. **Jupyter Project**. 2014. Disponível em: <https://jupyter.org>. Acesso em: 16 de outubro de 2021.

KNIPE, D.; HOWLEY, P. **Fields Virology**. [S.l.]: Wolters Kluwer Health, 2013. ISBN 9781469830667.

KUMAR, S.; STECHER, G.; LI, M.; KNYAZ, C.; TAMURA, K. MEGA X: Molecular Evolutionary Genetics Analysis across Computing Platforms. **Molecular Biology and Evolution**, v. 35, n. 6, p. 1547–1549, 05 2018. ISSN 0737-4038. Disponível em: <https://doi.org/10.1093/molbev/msy096>.

MASTERS, P. S. The molecular biology of coronaviruses. In: . Academic Press, 2006, (Advances in Virus Research, v. 66). p. 193–292. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0065352706660053>.

Microsoft. **Documentação do TypeScript**. 2012. Disponível em: <https://www.typescriptlang.org>. Acesso em: 16 de outubro de 2021.

MongoDB Inc. **MongoDB**. 2009. Disponível em: <https://www.mongodb.com>. Acesso em: 16 de outubro de 2021.

MongoDB Inc. **MongoDB Atlas**. 2016. Disponível em: <https://www.mongodb.com/pt-br/cloud>. Acesso em: 16 de outubro de 2021.

OMS. **Coronavirus disease 2019 (COVID-19). Situation Report – 51**. 2020. Disponível em: <https://www.who.int/docs/default-source/coronaviruse/situation-reports/20200311-sitrep-51-covid-19.pdf?sfvrsn=1ba62e57_10>.

OpenJS Foundation. **Node.JS**. 2009. Disponível em: <nodejs.org>. Acesso em: 16 de outubro de 2021.

SALADIN, K. **Anatomy & Physiology: The Unity of Form and Function**. [S.l.]: McGraw-Hill, 2012. ISBN 9780071316385.

SAYERS, E. W.; CAVANAUGH, M.; CLARK, K.; PRUITT, K. D.; SCHOCH, C. L.; SHERRY, S. T.; KARSCH-MIZRACHI, I. GenBank. **Nucleic Acids Res**, v. 49, n. D1, p. D92–D96, 01 2021.

SHARP, P. M.; LI, W. H. The rate of synonymous substitution in enterobacterial genes is inversely related to codon usage bias. **Molecular Biology and Evolution**, v. 4, n. 3, p. 222–230, Maio 1987. ISSN 0737-4038. Disponível em: <https://doi.org/10.1093/oxfordjournals.molbev.a040443>.

SUN, P.; LU, X.; XU, C.; SUN, W.; PAN, B. Understanding of covid-19 based on current evidence. **Journal of medical virology**, John Wiley and Sons Inc., v. 92, n. 6, p. 548–551, Jun 2020. ISSN 1096-9071. 32096567[pmid]. Disponível em: <https://doi.org/10.1002/jmv.25722>.

YADAV, P. D.; POTDAR, V. A.; CHOUDHARY, M. L.; NYAYANIT, D. A.; AGRAWAL, M.; JADHAV, S. M.; MAJUMDAR, T. D.; SHETE-AICH, A.; BASU, A.; ABRAHAM, P.; CHERIAN, S. S. Full-genome sequences of the first two sars-cov-2 viruses from india. **The Indian journal of medical research**, Wolters Kluwer - Medknow, v. 151, n. 2 & 3, p. 200–209, 2020. ISSN 0971-5916. 32242873[pmid]. Disponível em: <https://doi.org/10.4103/ijmr.IJMR_663_20>.

ZAKI, A. M.; BOHEEMEN, S. van; BESTEBROER, T. M.; OSTERHAUS, A. D.; FOUCHIER, R. A. Isolation of a novel coronavirus from a man with pneumonia in saudi arabia. **New England Journal of Medicine**, v. 367, n. 19, p. 1814–1820, 2012. PMID: 23075143. Disponível em: <https://doi.org/10.1056/NEJMoa1211721>.

ZEHENDER, G.; LAI, A.; BERGNA, A.; MERONI, L.; RIVA, A.; BALOTTA, C.; TARKOWSKI, M.; GABRIELI, A.; BERNACCHIA, D.; RUSCONI, S.; RIZZARDINI, G.; ANTINORI, S.; GALLI, M. Genomic characterization and phylogenetic analysis of sars-cov-2 in italy. **Journal of Medical Virology**, v. 92, n. 9, p. 1637–1640, 2020. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/jmv.25794>.