UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE COMPUTAÇÃO
CURSO DE ENGENHARIA DE COMPUTAÇÃO

VITOR GREGORIO

FERRAMENTA DE EXTRAÇÃO DE CARACTERÍSTICAS DE SEQUÊNCIAS BIOLÓGICAS

TRABALHO DE CONCLUSÃO DE CURSO FINAL

CORNÉLIO PROCÓPIO
2021

VITOR GREGORIO

FERRAMENTA DE EXTRAÇÃO DE CARACTERÍSTICAS DE SEQUÊNCIAS BIOLÓGICAS

Trabalho de Conclusão de Curso Final apresentado ao Curso de Engenharia de Computação da Universidade Tecnológica Federal do Paraná, como requisito parcial para a obtenção do título de Bacharel.

Orientador: Alexandre Rossi Paschoal
Universidade Tecnológica Federal do Paraná

CORNÉLIO PROCÓPIO
2021

TERMO DE APROVAÇÃO

Ferramenta de extração de características de sequências biológicas

por

Vitor Gregorio

Este Trabalho de Conclusão de Curso de graduação foi julgado adequado para obtenção do Título de “Graduação em Engenharia de Computação” e aprovado em sua forma final pelos membros listados da banca da Universidade Tecnológica Federal do Paraná.

Cornélio Procópio, 11/05/2021

_____

Prof. Dr. Alexandre Rossi Paschoal

_____

Prof. Dr. André Yoshiaki Kashiwabara

_____

Prof. Dr. Fábio Fernandes da Rocha Vicente

“A Folha de Aprovação assinada encontra-se na Coordenação do Curso”

AGRADECIMENTOS

Agradeço ao professor Dr. Alexandre Rossi Paschoal pela oportunidade, confiança, paciência e aprendizado passado durante o período do projeto; à minha família que me apoiou financeiramente para eu realizar meus estudos; à Tatianne da Costa Negri, pelo suporte; e aos meus amigos, em especial Aline Bini e Ana Clara Bergamin e Beatriz Gambaro, pelo companheirismo e ajuda.

A ciência poderá ter achado a cura para a maioria dos males, mas não achou ainda remédio para o pior de todos: a apatia dos seres humanos. (KELLER, Helen).

RESUMO

GREGORIO, Vitor. Ferramenta de extração de características de sequências biológicas. 2021. 30 f. Trabalho de Conclusão de Curso Final – Curso de Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2021.

A bioinformática é uma área da ciência que busca analisar, interpretar e solucionar problemas biológicos. Na computação, a análise exploratória de dados permite visualizar e ter uma maior compreensão sobre os dados. Por exemplo, representações gráficas ou tabelares são formas amigáveis de ter esse visão. Nesse sentido, este TCC apesenta o resultado final da construção de uma ferramenta *Desktop* para análise exploratória de sequências biológicas, apelando para visualização destes dados. A ferramenta foi desenvolvida em Python, que permite gerar relatórios para visualização dos resultados. Foram implementadas nove características, sendo: tamanho da sequência, conteúdo GC e taxa (ratio) GC, contagem k-mer (e.g., dinucleotídeos e trinucleotídeos), *Dinucleotide-based Auto Covarience*, *Dinucleotide-based Cross Covarience*, *Trinucleotide-based Auto Covarience* e *Trinucleotide-based Cross Covarience*. Assim, bibliotecas como Biopython, Numpy, Tkinter e Matplotlib, foram usadas na construção da análise das sequências biológicas e criação de gráficos, através de uma interface intuitiva e usual. Por fim, está é uma ferramenta amigável em que o usuário pode inserir a sequência e exportar seus relatórios e gráficos em vários formatos para ser usado em seus trabalhos científicos.

Palavras-chave: Bioinformática. Python. Análise exploratória de dados.

ABSTRACT

GREGORIO, Vitor. Extraction tool for biological sequence features. 2021. 30 f. Trabalho de Conclusão de Curso Final – Curso de Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2021.

Bioinformatics is an area of science that seeks to analyze, interpret and solve biological problems. In the analysis, an exploratory data analysis allows you to visualize and have a greater understanding of the data. For example, graphical or tabular representations are friendly ways of having this vision. In this sense, this TCC presents the final result of the construction of a *Desktop* tool for exploratory analysis of biological sequences, appealing for visualization of this data. The tool was developed in Python, which allows generating reports to view the results. Nine characteristics were implemented, namely: sequence size, GC content and GC ratio, k-mer count (ex: dinucleotides and trinucleotides), *Dinucleotide-based Auto Covarience*, *Dinucleotide-based Cross Covarience* , *Trinucleotide-based Auto Covarience* and *Trinucleotide-based Cross Covarience*. Thus, libraries such as Biopython, Numpy, Tkinter and Matplotlib, were used in the construction of the analysis of biological sequences and creation of graphics, through an intuitive and usual interface. Finally, it is a user-friendly tool in which the user can insert the sequence and export his reports and graphs in various formats to be used in his scientific works.

Keywords: Bioinformatics. Python. Exploratory data analysis.

LISTA DE FIGURAS

LISTA DE ABREVIATURAS E SIGLAS

RNA	Ácido ribonucleico
CSV	Valores separados por virgula
miRNA	microRNA
ncRNA	RNA não-codificante
lncRNA	RNA longo não-codificante

SUMÁRIO

1 INTRODUÇÃO

Na bioinformática, os cientistas se deparam com muitos dados numéricos, oriundos de arquivos enormes ou grandes sequências biológicas, como o DNAs, por exemplo. Geralmente, esses dados estão armazenados em tabelas ou arquivos texto e similar, que na maioria dos casos, não facilitam a sua manipulação ou consulta pelo usuário. Por exemplo, ferramentas de predição, em geral, aceitam um ou mais arquivos FASTA como entrada, e fornecem como saída uma lista (tabela) com os resultados. Assim é cada vez maior a demanda do processamento de um grande volume de dados, e ainda, extrair o conhecimento das sequências biológicas para ajudar na tomada de decisão.

De acordo com a *International Data Corporation*, a cada dois anos, a quantidade de dados produzidos dobra, tornando cada vez mais complexo a interpretação dos dados (GREGO, 2014). Porém, com a ajuda da estatística, a análise exploratória se torna essencial, e o uso de representação gráficas torna-se uma ferramenta fundamental na etapa de interpretação dos dados (JERóNIMO, 2016).

Dado o cenário, o objetivo desse trabalho final foi desenvolver uma ferramenta amigável de análise exploratória de características em sequências biológicos *desktop*. Em particular, a entrada pode ser uma sequência de DNA ou RNA. Por meio de uma interface gráfica, a ferramenta irá contribuir na tomada de decisão sobre análise em sequências biológicas. A partir da entrada da sequência fornecida pelo usuário, gráficos e relatórios são apresentados. Nela serão apresentados gráficos e relatórios com os resultados da análise, além de arquivo para o usuário exportar a saída, visando facilitar a utilização dos dados por parte do usuário.

Espera-se que essa ferramenta ajude os biólogos, não especialistas, bioinformatas ou usuários de áreas similares, a interpretar e tomar as decisões na análise de características das sequências.

Por fim, neste documento será apresentado uma revisão e conceitos dos termos que serão utilizados. Em seguida, será explicado de forma mais aprofundada como funcionará a ferramenta, e as tecnologias necessárias para utiliza-la, além do método que será programada essa ferramenta. Por fim, será apresentada a interface criada, junto com os resultados obtidos na validação da ferramenta.

2 REVISÃO DE LITERATURA

2.1 BioSeq-Analysis 2.0

É a segunda versão de uma plataforma desenvolvida para a análise de diversas sequências biológicas utilizando aprendizado de máquina, pois a primeira versão só podia ser utilizada para análise de sequência, e agora pode ser utilizada para análise de nível residual também. Assim, com a extração de características, é possível realizar a predição final utilizando 26 diferentes tipos de características de DNA, RNA ou proteínas (LIU; GAO; ZHANG, 2019). E como resultado final, é apresentado os valores da sensibilidade, especificidade, precisão, *Mathew's Correlation Cofficient*, e a área abaixo da curva ROC, além do gráfico da curva ROC (*Receiver Operating Characteristic*), por se tratar de um processo de aprendizado de máquina.

A ferramenta possui a versão Web e também uma versão *desktop* para Linux e Windows, porém a sua versão *desktop* possui uma complexidade maior para realizar a instalação, mesmo seguindo o tutorial. Além disso, na versão *desktop*, não possui nenhuma interface que ajuda o usuário a utilizá-la, tendo que realizar todas as operações pelo terminal.

2.2 iLearn

É uma ferramenta de aprendizado de máquina com o mesmo objetivo que a *BioSeq-Analysis*, porém, buscaram apresentar algumas vantagens, como por exemplo uma quantidade maior de descritores de dados biológicos, ou a melhoria da análise das características, entre outras (CHEN et al., 2019). E como resultado, ela apresenta as mesmas características da *BioSeq-Analysis*, mas também mostra os valores da precisão, revocação, *F1-Score* e a área sob a curva de precisão-revocação.

Da mesma forma que a *BioSeq-Analysis*, a ferramenta *iLearn* possui uma versão Web e uma *desktop*, e que da mesma forma, possui uma dificuldade maior para ser instalada. E na versão Web, não é muito fácil de ser utilizada, e não é muito bem explicado quando o usuário se depara com um erro.

2.3 Análise exploratória de dados

A análise exploratória de dados é o processo de sintetização de dados utilizando valores numérico estatísticos, que podem ser apresentados via gráficos e tabelas, visando verificar a validade de premissas necessárias para a inferência estática; a validação e qualidade dos dados; ou identificar as estratégias analíticas e estatísticas apropriadas (TEO, 2009).

De acordo com Medri, existem diversas maneiras de realizar a análise de uma forma detalhada, onde o objetivo é adquirir o máximo de informação desses dados, para serem

utilizadas posteriormente (MEDRI, 2011). A análise exploratória busca utilizar os dados para serem examinados antes de aplicar em algum método estatístico (FONSECA, 2018).

Para Nist, a maioria das técnicas de análise exploratória estão relacionadas com a utilização de gráficos, histogramas, tabelas, fluxogramas e outras técnicas de estatísticas simples. Ele também salienta que a utilização dos gráficos e outras técnicas facilitam o entendimento dos resultados (NIST/SEMATECH, 2003).

2.4 Bioinformática

A bioinformática é a área interdisciplinar, que numa visão da computação, busca ajudar de forma rápida e eficiente a análise de dados biológicos, em específico os estudos da biologia molecular. De acordo com Pevsner, o National Institutes of Health define a bioinformática como a pesquisa, o desenvolvimento e a aplicação de recursos e técnicas computacionais para aumentar o uso de dados biológicos, incluindo as tarefas de aquisição, armazenamento, análise ou visualização dos dados (PEVSNER, 2015).

2.5 RNA não-codificante

RNAs não-codificantes (ncRNA) são RNAs que são transcritos, mas não conseguem ser traduzidos em proteínas, ainda que tenham suas funções biológicas, como alterações da cromatina, regulação pós-transcricional, organização nuclear, tradução e outros processos de desenvolvimento (CORREIA; CORREIA, 2007).

Os ncRNAs podem ter origens de diversas regiões do genoma, podendo ser de regiões não codificadoras, como os introns, ou codificadoras, como os exons. Assim, os ncRNAs podem ser classificados através do seu formato, tamanho ou função da molécula. Por exemplo, os RNAs longos, que são um tipo de ncRNA, são compostos de no mínimo 200 nucleotídeos de tamanho, já os *smalls* RNAs, que também são classificados como ncRNA, possuem um tamanho de no máximo 200 nucleotídeos (FONSECA, 2018).

2.6 RNA longo não-codificante

LncRNA é um tipo de ncRNA, que vem se destacando muito nos últimos anos. Eles possuem sequências de no mínimo 200 nucleotídeos, e apresentam uma grande importância nos genomas de eucariotos, pois eles estão ligados com a regulação da expressão gênica transcricional e pós-transcricional (NEGRI et al., 2018).

Apesar dessa importância, poucos lncRNAs tem as funções validadas em plantas, pois se trata de um procedimento muito difícil, sendo considerado um dos maiores desafios dos próximos 20 anos na pesquisa de RNAs (NEGRI et al., 2018).

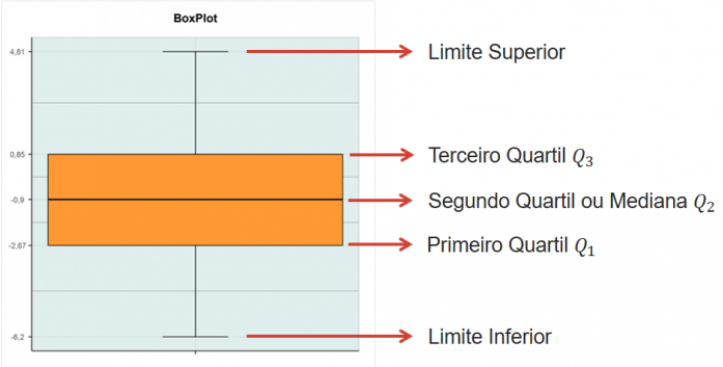
2.7 Heurísticas de Nielsen

As heurísticas de Nielsen são 10 normas para padronizar desenvolvimentos de interface, de forma a melhorar a usabilidade do *software*, sem precisar de um manual de instrução. Como por a heurística de visibilidade de status do sistema, que o usuário está sempre sendo informado das ações que está realizando. Ou então a heurística de prevenção de erros, quando o usuário é perguntado se deseja mesmo realizar uma ação (MACEDO, 2017).

2.8 Boxplot

A utilização do *boxplot* se deve à sua fácil visualização, uma vez que ela nos permite perceber a dispersão e assimetria dos dados. Ele é separado em quartis, onde o quartil 1 é o valor responsável por 25% das amostras, o quartil 3 é o valor responsável por 75% das amostras e o quartil 2 é a mediana, como pode ser observado na Figura 1 (ACTION,).

Figura 1 – Exemplo de boxplot



3 METODOLOGIA

3.1 Linguagem de programação

Com o objetivo de desenvolver uma ferramenta de análise exploratória de dados, com uma interface gráfica, foi desenvolvido um código em Python que irá realizar todos os cálculos necessários para a análise, e também a criação da interface. A utilização dessa linguagem se deve à sua sintaxe concisa e fácil, a caracterizando como uma linguagem mais produtiva. Além disso, é possível utilizar bibliotecas e *frameworks* de terceiros, que facilita a usabilidade da mesma (BORGES, 2014).

3.2 Bibliotecas

Uma das bibliotecas que será utilizada, é a Biopython. Ela foi desenvolvida focada em aplicações de bioinformática, onde facilmente ajudará na abertura e lida de arquivos biológicos, e outras características necessárias (COCK et al., 2009). Ela apresenta módulos separados para análise de proteínas, genética de população, alinhamento de sequências, *machine learning*, visualização de dados biológicos e filogenia (MARIANO, 2016).

Também será utilizado as bibliotecas Numpy (OLIPHANT, 2006) e Pandas (TEAM, 2020), que serão responsáveis por facilitar a utilização de possíveis vetores e matrizes, uma vez que elas oferecem estrutura e operações para manipular as séries e tabelas.

Outra biblioteca, que será muito importante, é a Matplotlib (HUNTER, 2007), que foi desenvolvida para a criação de visualização de dados em geral, e é baseada no MatLab (THE MATHWORKS, INC., 2017).

Além disso, para realizar a criação da interface gráfica, será utilizada o pacote Tkinter, que é a interface padrão do Python e está disponível na maioria das plataformas. Assim, ela se torna perfeita para o uso, uma vez que é nativa da linguagem e é muito fácil de ser utilizada, sendo necessário apenas realizar a importação da mesma (ROSSUM; DRAKE, 2009).

3.3 *Características (Features)*

3.3.1 *Width*

A *feature* de *width* irá calcular o comprimento de cada sequência de cada arquivo inserido pelo usuário, e em seguida será criado um gráfico de *boxplot* com o resultado, para cada arquivo inserido inicialmente. Essa *feature* pode ser considerada importante, uma vez que ela apresenta a variação dos comprimentos das sequências, e ficando fácil para entender se as sequencias são curtas ou longas.

3.3.2 *GC Content*

Na *feature GC Content*, será calculado o valor do GC content (1), e criado o gráfico de *boxplots* para cada arquivo. O valor do GC Content é importante, pois está relacionado com as características biológicas da organização do genoma, como densidade do gene, taxa de mutação, e nível e especificidade do tecido da transcrição (KOROL, 2013).

$$GcContent = \frac{G + C}{G + C + A + T} \tag{1}$$

3.3.3 *GC Ratio*

Já na *feature GC Ratio*, será calculado o valor do GC ratio(2), e também será criado o gráfico de *boxplots* para cada arquivo. Ele é utilizado para retratar o potencial de estabilidade térmica da molécula, onde quanto menor o seu valor, mais instável a molécula é (FONSECA, 2018).

$$GcRatio = \frac{G}{C} \tag{2}$$

3.3.4 Dinucleotídeo e Trinucleotídeo

Nas *features* Dinucleotídeo e Trinucleotídeo serão calculados as quantidades de dinucleotídeos e trinucleotídeos de cada arquivo, e em seguida serão criados gráficos de barras individuais para cada arquivo. Dentre as aplicações, podem ser citados o alinhamento de sequências, montagem do genoma, estimativa do tamanho do genoma, e identificação de repetição (BRIHADISWARAN, 2020).

3.3.5 Autocorrelação

A autocorrelação descreve o nível de correlação entre dois dinucleotídeos ou trinucleotídeos em termos de sua estrutura específica ou propriedade físico-química. Esse nível pode ser utilizado para prever conteúdo de hélices de proteínas, e tipos de proteína transmembrana (DONG ZHI-JIANG YAO, 2016). Eles podem ser divididos em quatro tipos, que são DAC (*Dinucleotide-based Auto Covariance*), DCC (*Dinucleotide-based Cross Covariance*), TAC (*Trinucleotide-based Auto Covariance*) e TCC (*Trinucleotide-based Cross Covariance*), onde os dois de autocorrelação são para uma mesma propriedade, e os de cross covariação são para duas propriedades físico-químicas diferentes.

Assim, o usuário poderá escolher a propriedade e a distância entre os dinucleotídeos ou trinucleotídeos, resultando na criação de um arquivo com os resultados para cada sequência. Os valores físico-químicos foram retirados da ferramenta iLearn(CHEN et al., 2019), onde foi criada uma tabela com seis propriedades físico-químicas para cada dinucleotídeos, e uma outra com duas propriedades físico-química para cada trinucleotídeos. E essas tabelas foram retiradas da ferramenta Bio-Seq Analysis 2.0(LIU; GAO; ZHANG, 2019).

3.3.5.1 DAC (*Dinucleotide-based Auto Covariance*)

Ele mede a correlação por toda a sequência, de um mesmo índice físico-químico para dois dinucleotídeos, separado por uma distância chamada de LAG(CHEN et al., 2019). E ele é calculado pela equação DAC(3).

$$DAC = \sum_{j=1}^{L-LAG-1} ((P_u(R_jR_{j+1}) - \overline{P_u})((P_u(R_{j+LAG}R_{j+1+LAG}) - \overline{P_u})/(L - LAG - 1)) \quad (3)$$

Onde *L* é o comprimento da sequência, *u* é o índice físico-químico, e $P_u(R_jR_{j+1})$ é o valor do índice físico-químico para o dinucleotídeo, e a média dos índices físico-químicos ao longo da sequência é determinado por $\overline{P_u}$ (4).

$$\overline{P_u} = \sum_{k=1}^{L-1}((P_u(R_kR_{k+1})/(L-1)) \quad (4)$$

3.3.5.2 DCC (*Dinucleotide-based Cross Covariance*)

Ele mede a correlação por toda a sequência, de dois índices físico-químicos para dois dinucleotídeos, separado por uma distância chamada de LAG(CHEN et al., 2019). E ele é calculado pela equação DCC(5).

$$DCC = \sum_{j=1}^{L-LAG-1} ((P_{u_1}(R_jR_{j+1}) - \overline{P_{u_1}})((P_{u_2}(R_{j+LAG}R_{j+1+LAG}) - \overline{P_{u_2}})/(L - LAG - 1)) \quad (5)$$

Onde *L* é o comprimento da sequência, *u* é o índice físico-químico, e $P_u(R_jR_{j+1})$ é o valor do índice físico-químico para o dinucleotídeo, e a média dos índices físico-químicos ao longo da sequência é determinado por $\overline{P_u}$ (6).

$$\overline{P_{u_x}} = \sum_{k=1}^{L-1}((P_{u_x}(R_kR_{k+1})/(L-1)) \quad (6)$$

3.3.5.3 TAC (*Trinucleotide-based Auto Covariance*)

Ele mede a correlação por toda a sequência, de um mesmo índice físico-químico para dois trinucleotídeos, separado por uma distância chamada de LAG(CHEN et al., 2019). E ele é calculado pela equação TAC(7).

$$TAC = \sum_{j=1}^{L-LAG-2}((P_u(R_jR_{j+1}R_{j+2}) - \overline{P_u})((P_u(R_{j+LAG}R_{j+1+LAG}R_{j+2+LAG}) - \overline{P_u})/(L - LAG - 2)) \quad (7)$$

Onde *L* é o comprimento da sequência, *u* é o índice físico-químico, e $P_u(R_jR_{j+1}R_{j+2})$ é o valor do índice físico-químico para o trinucleotídeo, e a média dos índices físico-químicos ao longo da sequência é determinado por $\overline{P_u}$ (8).

$$\overline{P_u} = \sum_{k=1}^{L-2}((P_u(R_kR_{k+1}R_{k+2})/(L-2)) \tag{8}$$

3.3.5.4 TCC (*Trinucleotide-based Cross Covariance*)

Ele mede a correlação por toda a sequência, de dois índices físico-químicos para dois dinucleotídeos, separados por uma distância chamada de LAG(CHEN et al., 2019). E ele é calculado pela equação TCC(9).

$$TCC = \sum_{j=1}^{L-LAG-2}((P_{u_1}(R_jR_{j+1}R_{j+2}) - \overline{P_{u_1}})((P_{u_2}(R_{j+LAG}R_{j+1+LAG}R_{j+2+LAG}) - \overline{P_{u_2}})/(L-LAG-2)) \tag{9}$$

Onde *L* é o comprimento da sequência, *u* é o índice físico-químico, e $P_u(R_jR_{j+1}R_{j+2})$ é o valor do índice físico-químico para o trinucleotídeo, e a média dos índices físico-químicos ao longo da sequência é determinado por $\overline{P_{u_x}}$ (10).

$$\overline{P_{u_x}} = \sum_{k=1}^{L-2}((P_{u_x}(R_kR_{k+1}R_{k+2})/(L-2)) \tag{10}$$

3.3.6 Exportar *features*

Por fim, os botões *Save in CSV* e *Save in ARFF* irão criar arquivos de textos com as *features* de comprimento, *GC Content*, e *GC Ratio* para cada arquivo, dessa forma, além do usuário possuir os gráficos, também possuirá os resultados em forma de texto. O formato ARFF também será disponibilizado pois é muito importante em *software* de aprendizado de máquina como o Weka (HALL et al., 2009).

3.4 Validação da ferramenta

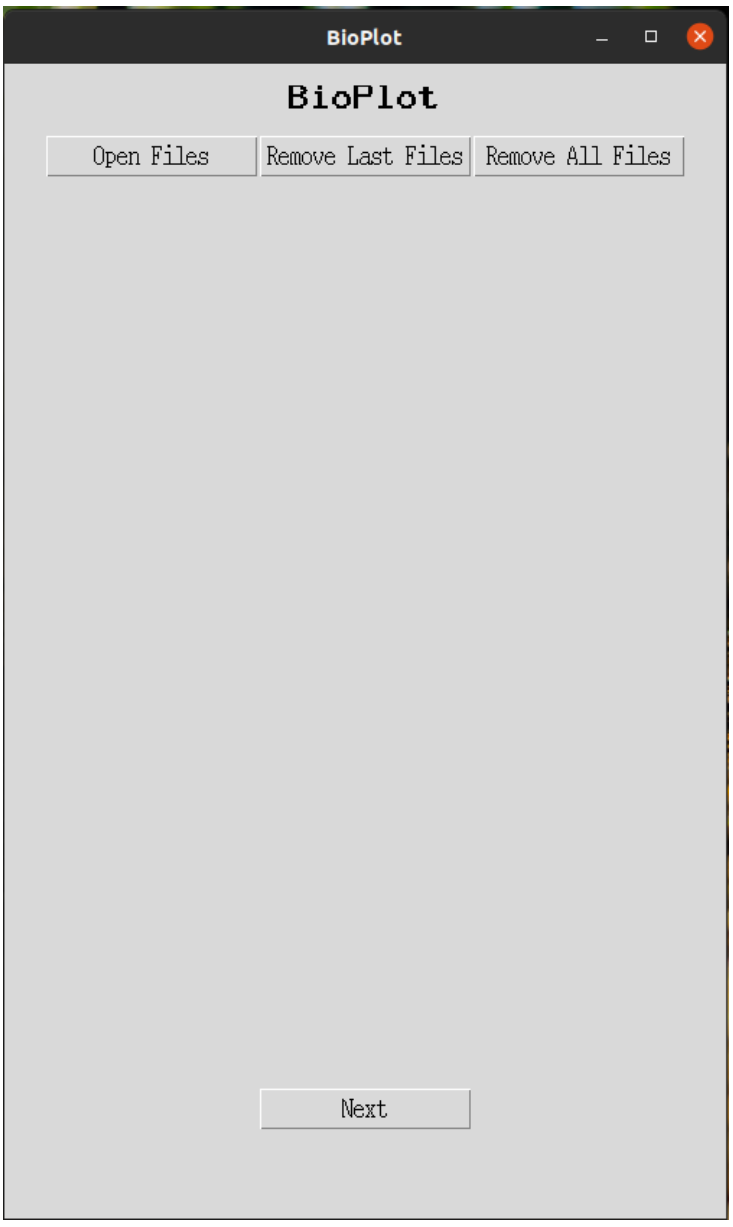
Para a realização da validação da ferramenta, serão utilizado sequências públicas de Longos RNAs não-codificantes (lncRNA), que estão anotadas no genoma das plantas *Arabidopsis thaliana, Glycine max* e *Zea mays*, que tem sido aplicados frequentemente em pesquisas científicas. Serão extraídos os arquivos no formato fasta dos bancos de dados (BD) CANTATAdb 2.0, PLncDB e GreeNC, totalizando assim nove genomas.

4 APRESENTAÇÃO E DISCUSSÃO DOS RESULTADOS

4.1 Interface

Com o intuito de criar uma interface limpa e intuitiva, foi desenvolvida uma página inicial para o usuário adicionar os arquivos que deseja utilizar, conforme a Figura 2. Assim, ao usuário clicar no botão para adicionar arquivos, irá aparecer uma caixa para ele escolher o arquivo que deseja utilizar, como pode ser observado na Figura 9 do Apêndice A, e em seguida, irá aparecer na interface o arquivo adicionado, conforme a Figura 10 do Apêndice A, podendo ser adicionado até dez arquivos, para que os gráficos finais fiquem indubitáveis.

Figura 2 – Página Inicial

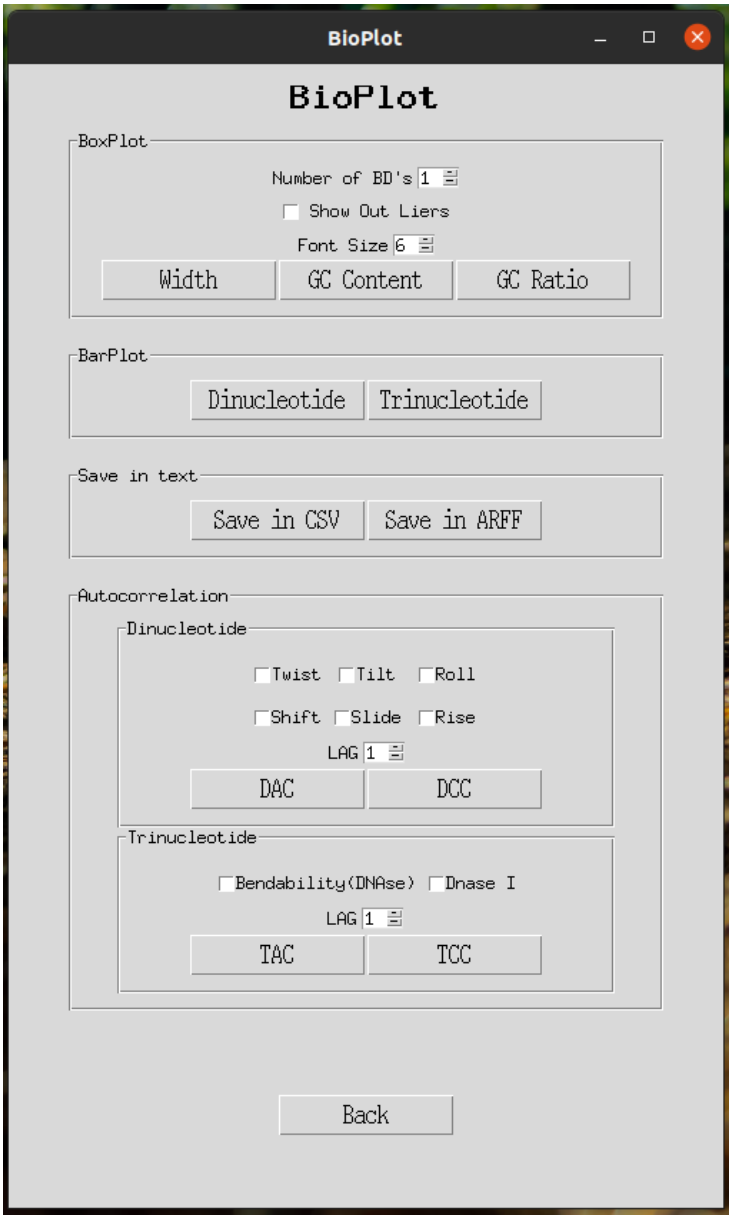


Assim, de forma a respeitar as heurísticas de Nielsen (MACEDO, 2017), toda vez que

o usuário adicionar ou remover um arquivo, uma mensagem na parte inferior da interface irá informar das ações. Além disso, sempre que o usuário clicar para remover um ou todos os arquivos, uma caixa de confirmação irá aparecer, para evitar erros acidentais.

Após o usuário adicionar os arquivos desejados, ele deverá pressionar o botão *next* que o levará para a tela de características, como mostra a Figura 3. Essa interface está dividida em 4 partes, onde na primeira é para a criação de *boxplots*, com as funções de Comprimento da sequência, *GC Content* e *GC Ratio*. A segunda parte é para a criação de gráfico de barras, contendo as características de dinucleotídeo e trinucleotídeo. A terceira parte é para a exportação dos resultados, onde o usuário poderá exportar no formato *CSV* ou *ARFF*. E por fim, a quarta parte é para o cálculo de autocorrelação, podendo ser determinado o DAC, DCC, TAC e TCC.

Figura 3 – Página de Características



Além disso, na parte de *Boxplot* é possível alterar a quantidade de banco de dados,

mudando as cores para cada *boxplot* no gráfico, também é possível mostrar ou remover os *outliers*, e aumentar o tamanho da fonte do nomes dos *boxplots*, para não ter perigo de sobrepor os nomes no final. E na parte de *Autocorrelation*, possui as caixas para a escolha da propriedade físico-química e o tamanho do LAG.

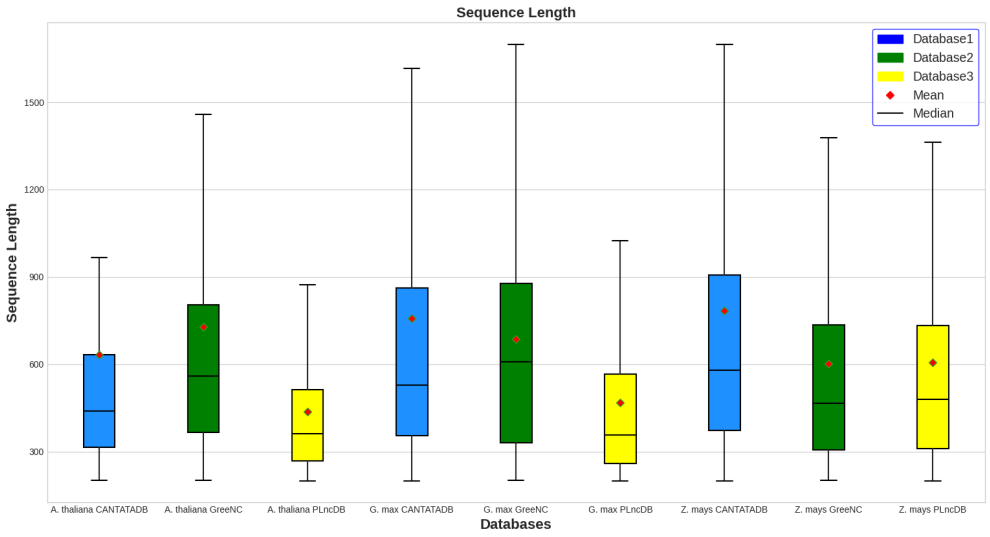
Por fim, na parte inferior da interface, irá aparecer mensagens de acordo com a *feature* escolhida pelo usuário, informando quando finalizado ou ocorrer algum erro. Além disso, possui o botão *back*, que permite o usuário voltar para a tela inicial e adicionar ou remover arquivos.

4.2 Validação

Primeiramente, foi realizada uma filtragem nas sequências de todos os arquivos, onde foram removidos todas as sequências que continham bases "N", pois se tratam de nucleotídeos que não foram possíveis de definir sua estrutura exatamente. Essa filtragem é importante para realizar os cálculos das autocorrelações, uma vez que não existe um valor físico-químico para as bases "N".

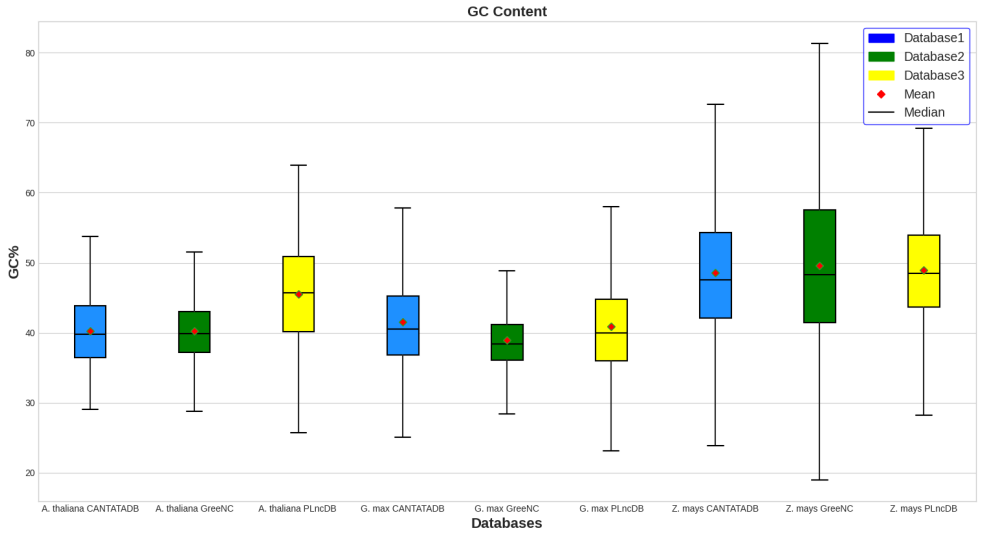
Após serem adicionados os arquivos com as sequências, igual a Figura 10, foi iniciado primeiramente a criação dos *boxplots*, onde definiu-se as configurações para três bancos de dados, sem mostrar os *outliers* e fonte da legenda com tamanho 14. Assim, o primeiro gráfico obtido foi para o comprimento das sequências, como pode ser visto na Figura 4. Em seguida, foram utilizadas as mesmas configurações para os gráficos de *GC Content* e *GC Ratio*, como pode ser observado na Figura 5 e Figura 6, respectivamente.

Figura 4 – Gráfico de comprimento das sequências



No gráfico do comprimento das sequências é possível observar que as sequências tem comprimento mínimo de 200 nucleotídeos, comprovando assim que são lncRNAs. Além disso, é

Figura 5 – Gráfico do GC Content



possível observar várias diferenças entre cada banco de dados, como a média e mediana dos comprimentos. Isso se deve por exemplo ao banco GreeNC possuir sequências muito maiores de *A. thaliana* que os outros bancos, ou por o banco PLncDB apresentar sequências menores.

Já no gráfico do *GC Content* é possível observar uma grande variação na porcentagem na espécie *Zea mays* do banco GreeNC, pois algumas sequências apresentam uma quantidade maior de guanina e citosina do que outras. Mas já a espécie *Glycine max* do mesmo banco apresenta pouca variação em relação aos outros bancos.

Por fim, no gráfico de *GC Ratio* apresenta em sua maioria a mesma quantidade de guanina e citosina, porém em algumas sequências a quantidade de guanina é muito maior que a de citosina, como no caso da espécie *A. thaliana* do banco PLncDB que a quantidade é quase quatro vezes maior, indicando que esse arquivo possui sequências com grande estabilidade térmica entre as moléculas.

Em seguida, foram criados os gráficos de barras de dinucleotídeo e trinucleotídeo para cada arquivo, resultando assim em nove gráficos para dinucleotídeo, que podem ser observadas a partir da Figura 11 até a Figura 19, e nove para trinucleotídeo, que podem ser observadas a partir da Figura 20 até a Figura 28, do Apêndice A. Assim é possível observar as maiores quantidades em cada arquivo. E quando comparado com o *boxplot* do ratio, mais especificamente com a espécie *A. thaliana* do banco de dados PLncDB, a quantidade de bases de dinucleotídeo e trinucleotídeo que contêm bases de citosina são menores que as restantes, comprovando assim a discrepância na Figura 6.

Além disso, com os resultados de dinucleotídeos da espécie *A. thaliana*, foi criado outro gráfico para comparar os três bancos, como pode ser visto na Figura 7. Assim, é possível observar uma grande discrepância entre os bancos, e isso está diretamente ligado aos

Figura 6 – Gráfico do GC Ratio

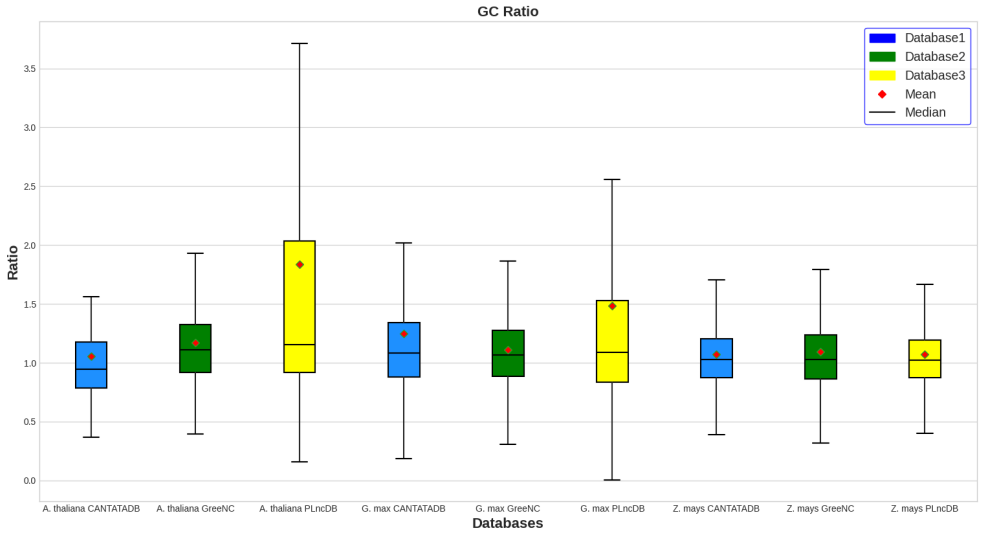
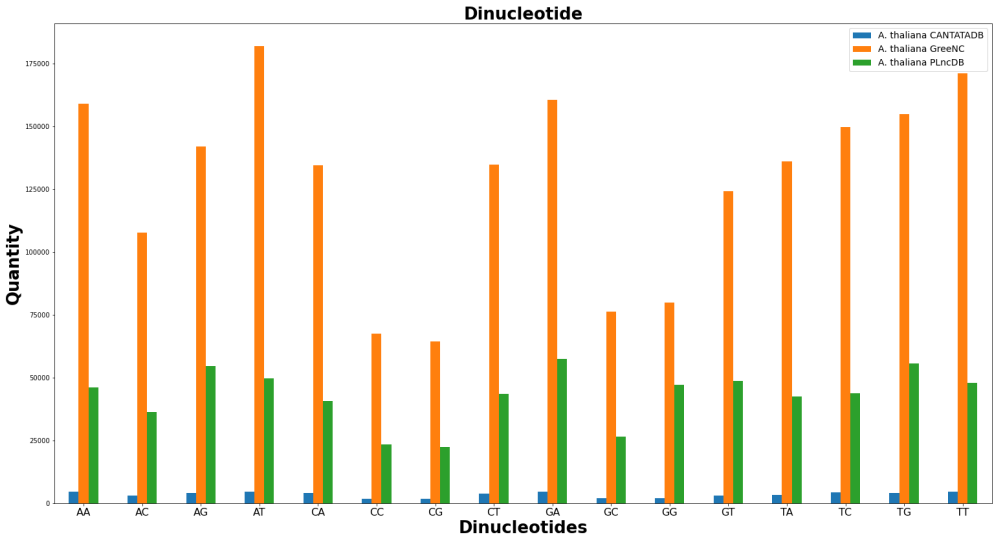
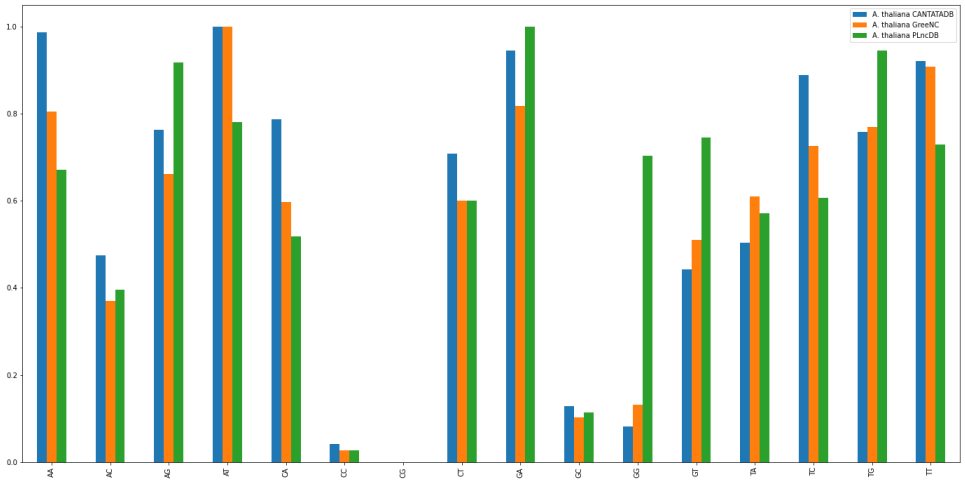


Figura 7 – Gráfico de Dinucleotídeo



comprimentos das sequências, uma vez que sequências maiores apresentam uma quantidade maior de dinucleotídeos. Mas como o gráfico apresenta barras com tamanhos muitos diferentes para cada banco de dados, realizei uma normalização de Mínimo-Máximo para cada *dataset*, e em seguida, foi criado o gráfico com esses valores normalizados, resultando na Figura 8. Dessa forma, podemos comparar a proporção de cada dinucleotídeo, para cada banco de dados da *A. thaliana*. E é possível perceber que a maioria dos dinucleotídeos apresentam a mesma proporção, tirando as barras de "GG", que apresenta uma quantidade muito maior no banco

Figura 8 – Gráfico de Dinucleotídeo Normalizado



PLncDB.

Posteriormente, foi realizada a solicitação de salvar o arquivo no formato *CSV*, para assim gerar nove arquivos com as características propostas. Dessa forma, uma tabela contendo o ID de cada sequência e acompanhado do valores do comprimento, *GC Content* e *GC Ratio* da mesma. Por estar trabalhando com *Dataframes*, o tempo para a conclusão aumenta de acordo com a quantidade de arquivos e de sequências, mas é informado na parte inferior da interface assim que concluído. O arquivo final é salvo com o mesmo nome do arquivo original e no formato desejado, uma vez que realiza os mesmos procedimentos para salvar em *ARFF*.

No primeiro teste de autocorrelação, foi definido *Twist* como a propriedade físico-química deseja, e o tamanho do LAG como 1. Dessa forma, foi calculado o *DAC* das sequências, e como saída foi gerado um arquivo no formato *CSV*, contendo as colunas com o ID da sequência e os valores do *DAC* de cada sequência.

Para o cálculo do *DCC*, definiu-se como propriedade físico-química os valores de *Twist* e *Tilt*, mantendo também o LAG de valor 1. Dessa forma, como saída foi gerado um arquivo no formato *CSV*, também contendo os ID's das sequências, mas dessa vez com mais duas colunas. A primeira com os valores de *Twist* para *Tilt*, e a segunda com os valores de *Tilt* para *Twist*.

Por fim, foi realizado o mesmo procedimento para o *TAC*, utilizando a propriedade físico-química *Bendability(DNAse)*, e da mesma forma que o *DAC*, foi gerado um arquivo de saída no formato *CSV*. Além disso, o cálculo do *TCC* ocorreu da mesma maneira que o *DCC*, mas nesse caso foram utilizado os padrões físico-químicos *Bendability(DNAse)* e *DNAse I*. Resultando também no arquivo de saída com as colunas do ID da sequência, *TCC* de *Bendability(DNAse)* para *DNAse I*, e *TCC* de *DNAse I* para *Bendability(DNAse)*.

Para garantir a confiabilidade dos resultados, foram utilizadas as ferramentas *iLearn*

e *BioSeq-Analysis* para comparar os meus resultados com o dessas ferramentas, utilizando os mesmos arquivos de entradas. E como esperado, os resultados foram aproximadamente os mesmos, tendo uma maior diferença no tempo de execução, uma vez que essas ferramentas utilizam procedimentos de aprendizado de máquina.

Da mesma forma que ocorreu para salvar as características em *CSV* ou *ARFF*, para calcular os valores de *DAC, DCC, TAC* e *TCC*, foram utilizadas listas e *dataframes*, causando uma lentidão ao ser solicitada a realização das tarefas. Assim, quanto mais arquivos solicitados e quanto maior as quantidade de sequências dos arquivos, mais demorado será esse processo.

5 CONSIDERAÇÕES FINAIS

Com o objetivo de desenvolver uma ferramenta para análise exploratória de dados, a linguagem de alto nível Python permitiu criar uma interface simples para usuários. Com gráficos coloridos e editáveis, um rápido entendimento dos resultados foi disponibilizado.

Além disso, com os resultados obtidos, o usuário pode utilizá-los de forma a serem salvos no formato *CSV*, para ser utilizado em outras ferramentas da análise em gráficos, ou até mesmo em *ARFF*, que posteriormente pode ser utilizado no software Weka para aprendizado de máquina.

E também, para garantir a confiabilidade dos resultados, foram realizados testes com ferramentas parecidas, e obtive resultados muito próximos entre as ferramentas, diferenciando apenas no tempo de execução da ferramenta.

Dessa forma, a ferramenta apresenta uma ótima usabilidade, permitindo o software ter uma boa vida útil, e por ser uma ferramenta amigável, pode ser facilmente utilizada por usuários leigos na computação. Principalmente para biólogos que não têm muita familiaridade com softwares de bioinformática, que através da interface torna sua interação mais intuitiva.

Referências

ACTION, P. 3.1-boxplot. Disponível em: <http://www.portalaction.com.br/estatistica-basica/31-boxplot>. Citado na página 4.

BORGES, L. E. **Python para desenvolvedores: aborda Python 3.3**. [S.l.]: Novatec Editora, 2014. Citado na página 5.

BRIHADISWARAN, G. Bioinformatics 1: K-mer counting. 2020. Disponível em: <https://medium.com/swlh/bioinformatics-1-k-mer-counting-8c1283a07e29>. Acesso em: 29 de abril de 2021. Citado na página 6.

CHEN, Z. et al. iLearn: an integrated platform and meta-learner for feature engineering, machine-learning analysis and modeling of DNA, RNA and protein sequence data. **Briefings in Bioinformatics**, v. 21, n. 3, p. 1047–1057, 04 2019. Citado 4 vezes nas páginas 2, 6, 7 e 8.

COCK, P. J. A. et al. Biopython: freely available Python tools for computational molecular biology and bioinformatics. **Bioinformatics**, v. 25, n. 11, p. 1422–1423, 03 2009. ISSN 1367-4803. Disponível em: <https://doi.org/10.1093/bioinformatics/btp163>. Citado na página 5.

CORREIA, J. D.; CORREIA, A. D. Funcionalidades dos rna não codificantes (ncrna) e pequenos rna reguladores, nos mamíferos. **REDVET. Revista Electrónica de Veterinaria**, Veterinaria Organización, v. 8, n. 10, p. 1–22, 2007. Citado na página 3.

DONG ZHI-JIANG YAO, M. W. e. a. J. Biotriangle: a web-accessible platform for generating various molecular representations for chemicals, proteins, dnas/rnas and their interactions. 2016. Citado na página 6.

FONSECA, B. H. R. d. Modelagem, integração e análise exploratória de dados públicos de mirtrons. 2018. Disponível em: <http://repositorio.utfpr.edu.br/jspui/handle/1/4507>. Citado 2 vezes nas páginas 3 e 6.

GREGO, M. Conteúdo digital dobra a cada dois anos no mundo. 2014. Disponível em: <https://exame.com/tecnologia/conteudo-digital-dobra-a-cada-dois-anos-no-mundo/>. Citado na página 1.

HALL, M. et al. The WEKA data mining software: an update. **SIGKDD Explorations**, v. 11, n. 1, p. 10–18, 2009. Citado na página 8.

HUNTER, J. D. Matplotlib: A 2d graphics environment. **Computing in Science & Engineering**, IEEE COMPUTER SOC, v. 9, n. 3, p. 90–95, 2007. Citado na página 5.

JERóNIMO, S. M. d. Q. P. C. Análise visual de dados: uma ferramenta também para startups! 2016. Disponível em: <https://jornaleconomico.sapo.pt/noticias/analise-visual-de-dados-uma-ferramenta-tambem-para-startups-2923>. Citado na página 1.

KOROL, A. B. Recombination. In: LEVIN, S. A. (Ed.). **Encyclopedia of Biodiversity (Second Edition)**. Second edition. Waltham: Academic Press, 2013. p. 353–369. ISBN 978-0-12-384720-1. Disponível em: <https://www.sciencedirect.com/science/article/pii/B9780123847195001209>. Citado na página 6.

LIU, B.; GAO, X.; ZHANG, H. BioSeq-Analysis2.0: an updated platform for analyzing DNA, RNA and protein sequences at sequence level and residue level based on machine learning approaches. **Nucleic Acids Research**, v. 47, n. 20, p. e127–e127, 09 2019. Disponível em: <https://doi.org/10.1093/nar/gkz740>. Citado 2 vezes nas páginas 2 e 6.

MACEDO, G. M. 10 heurísticas de nielsen para o design de interface. 2017. Acesso em: 29.03.2020. Disponível em: <https://brasil.uxdesign.cc/10-heur%C3%ADsticas-de-nielsen-para-o-design-de-interface-58d782821840>. Citado 2 vezes nas páginas 4 e 9.

MARIANO, E. A. **Introdução À Programação Para Bioinformática Com Biopython**. [S.l.]: Laboratório de Bioinformática e Sistemas Departamento de Ciência da Computação Universidade Federal de Minas Gerais, 2016. Citado na página 5.

MEDRI, D. W. análise exploratória de dados. 2011. Disponível em: <http://www.uel.br/pos/estatisticaeducacao/textos_didaticos/especializacao_estatistica.pdf>. Citado na página 3.

NEGRI, T. d. C. et al. Pattern recognition analysis on long noncoding RNAs: a tool for prediction in plants. **Briefings in Bioinformatics**, v. 20, n. 2, p. 682–689, 04 2018. ISSN 1477-4054. Disponível em: <https://doi.org/10.1093/bib/bby034>. Citado na página 3.

NIST/SEMATECH. **e-Handbook of Statistical Methods**. [s.n.], 2003. Disponível em: <https://doi.org/10.18434/M32189>. Citado na página 3.

OLIPHANT, T. E. **A guide to NumPy**. [S.l.]: Trelgol Publishing USA, 2006. v. 1. Citado na página 5.

PEVSNER, J. **Bioinformatics and functional genomics**. [S.l.]: John Wiley & Sons, 2015. Citado na página 3.

ROSSUM, G. V.; DRAKE, F. L. **Python 3 Reference Manual**. Scotts Valley, CA: CreateSpace, 2009. ISBN 1441412697. Citado na página 5.

TEAM, T. pandas development. **pandas-dev/pandas: Pandas**. Zenodo, 2020. Disponível em: <https://doi.org/10.5281/zenodo.3509134>. Citado na página 5.

TEO, Y. Y. Exploratory data analysis in large-scale genetic studies. **Biostatistics**, v. 11, n. 1, p. 70–81, 10 2009. ISSN 1465-4644. Disponível em: <https://doi.org/10.1093/biostatistics/kxp038>. Citado na página 2.

THE MATHWORKS, INC. **MATLAB version 9.3.0.713579 (R2017b)**. Natick, Massachusetts, 2017. Citado na página 5.

Apêndices

APÊNDICE A – Gráficos

Figura 9 – Caixa de seleção dos arquivos

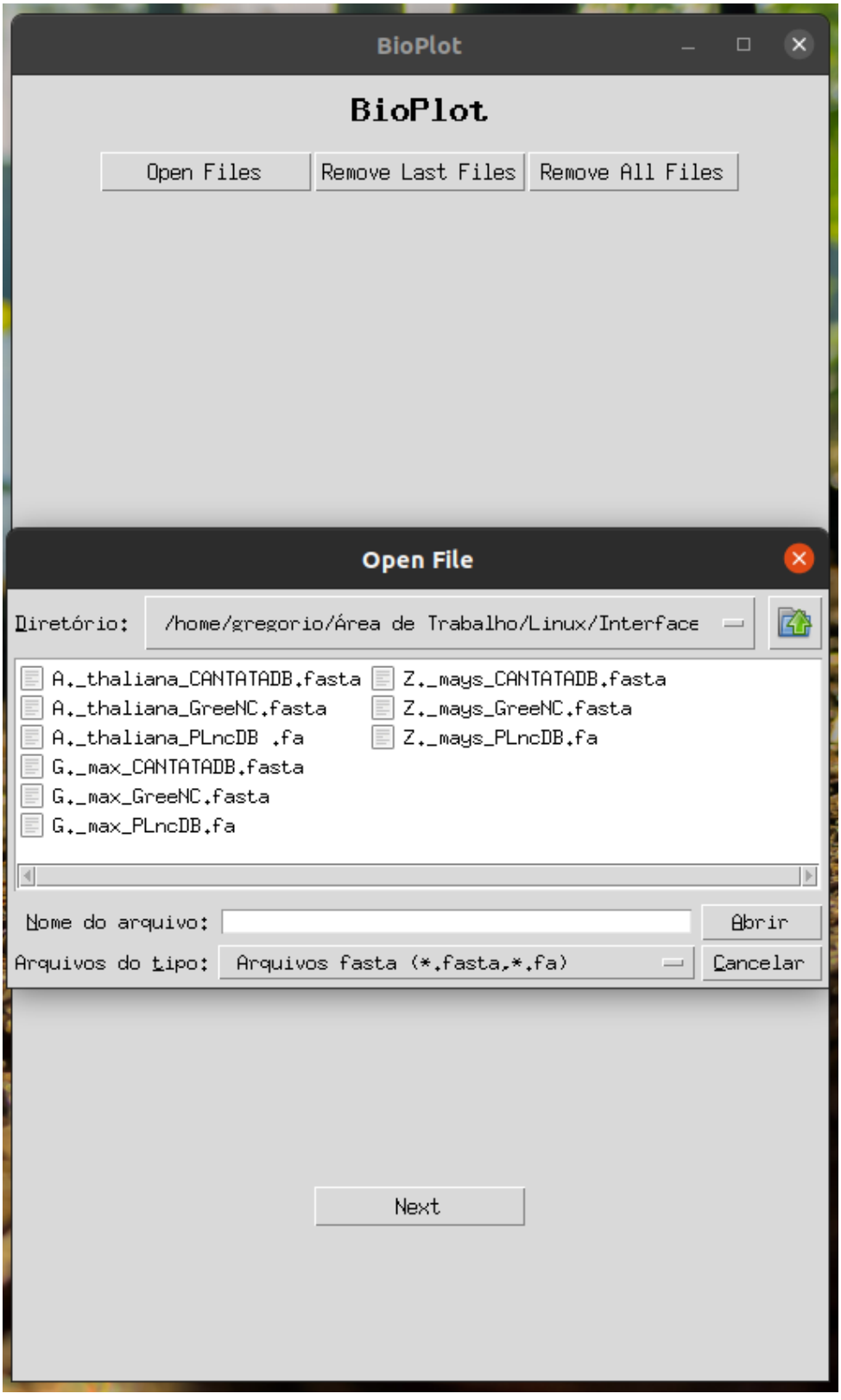


Figura 10 – Tela inicial com arquivos

Figura 11 – Gráfico do Dinucleotídeo da *A. thaliana* do CANTATADB

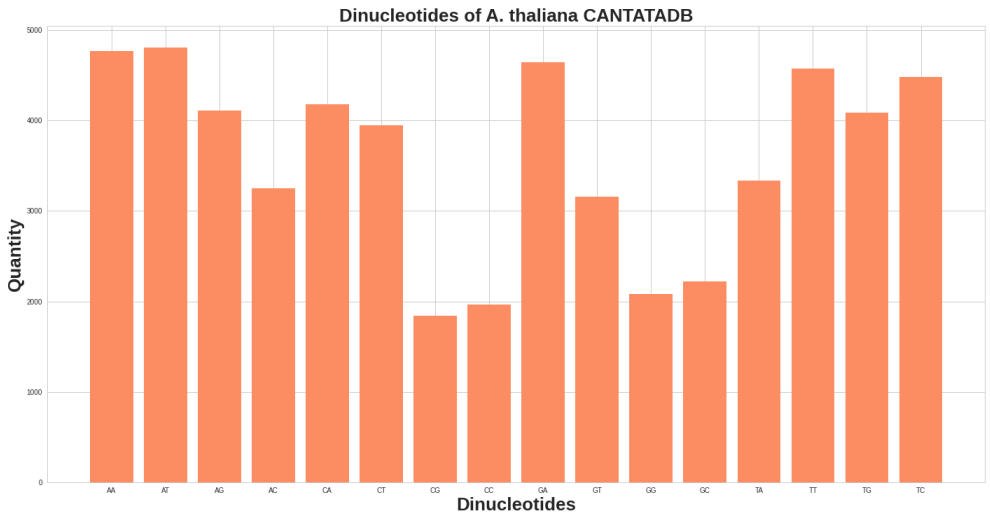


Figura 12 – Gráfico do Dinucleotídeo da *A. thaliana* do GreeNC

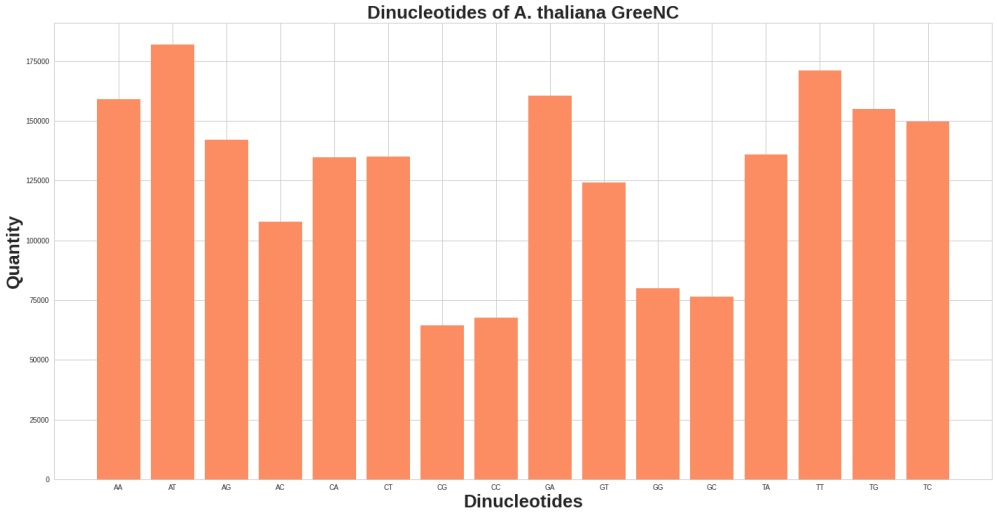


Figura 13 – Gráfico do Dinucleotídeo da *A. thaliana* do PLncDB

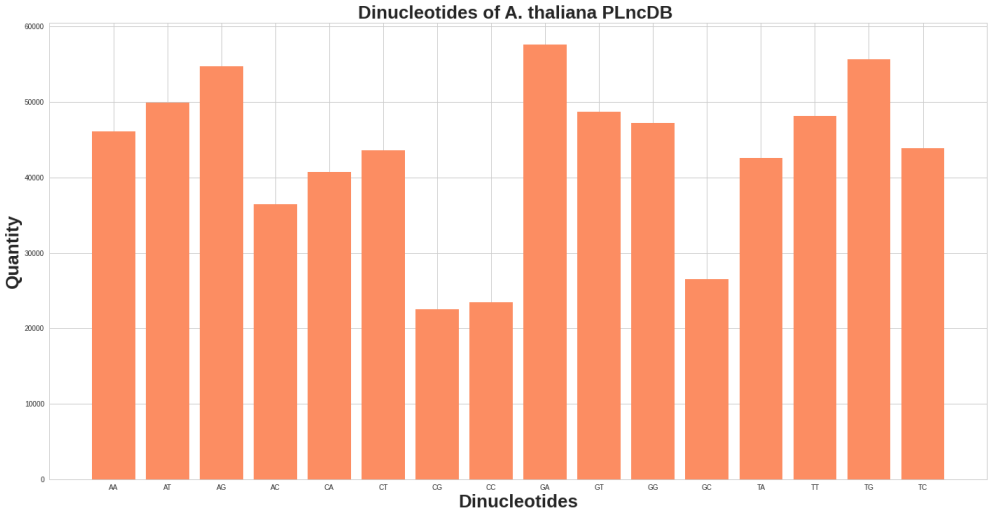


Figura 14 – Gráfico do Dinucleotídeo da *G. max* do CANTATADB

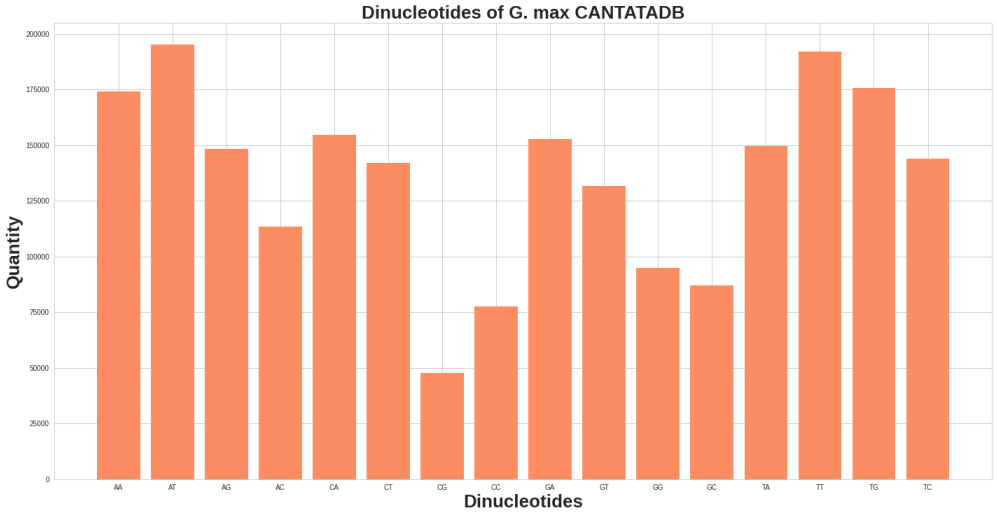


Figura 15 – Gráfico do Dinucleotídeo da *G. max* do GreeNC

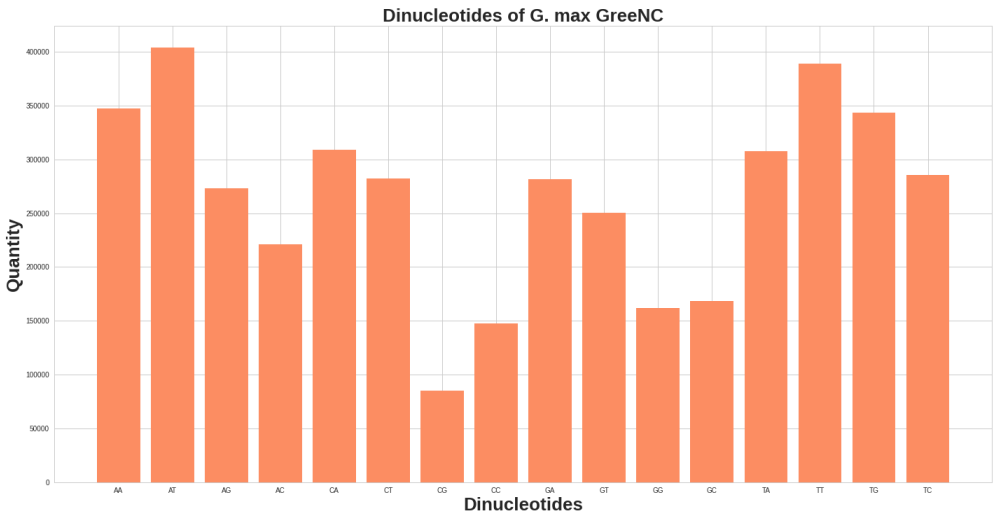


Figura 16 – Gráfico do Dinucleotídeo da *G. max* do PLncDB

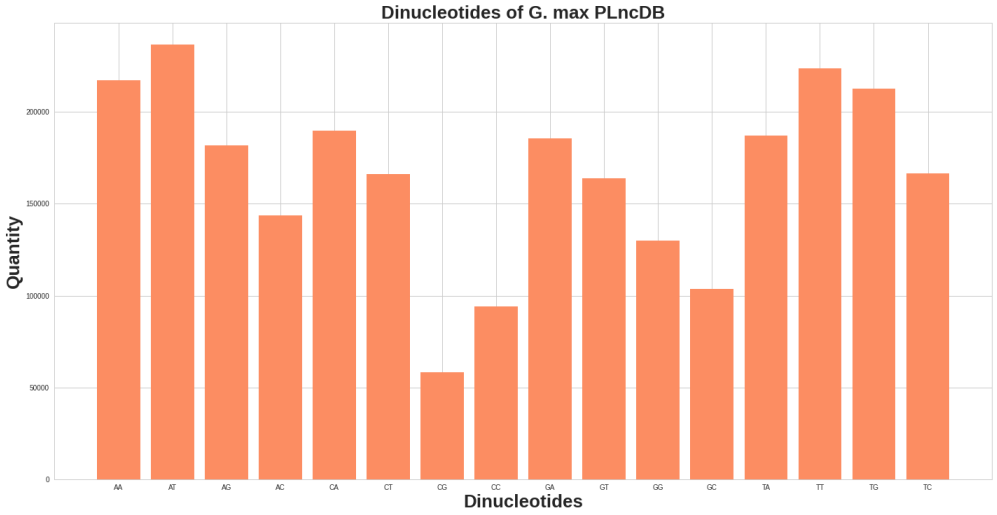


Figura 17 – Gráfico do Dinucleotídeo da *Z. mays* do CANTATADB

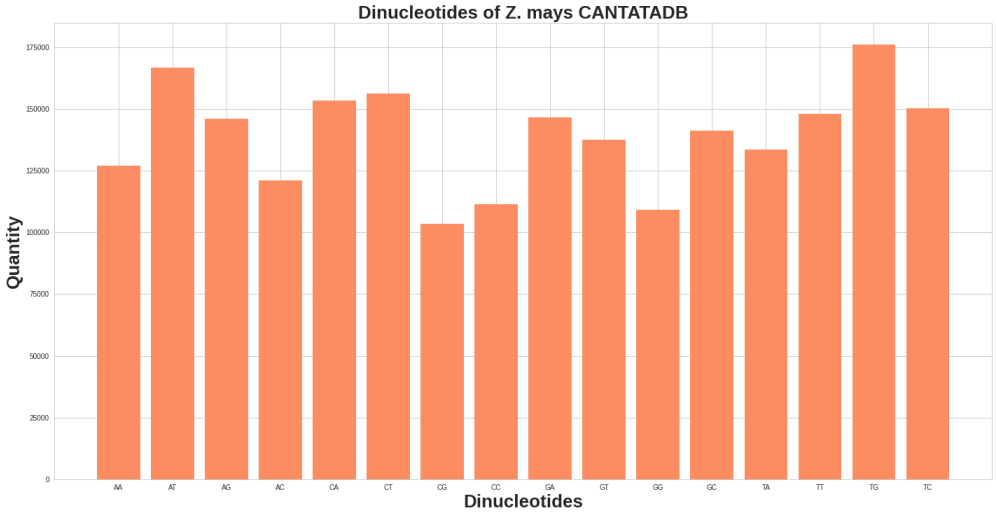


Figura 18 – Gráfico do Dinucleotídeo da *Z. mays* do GreeNC

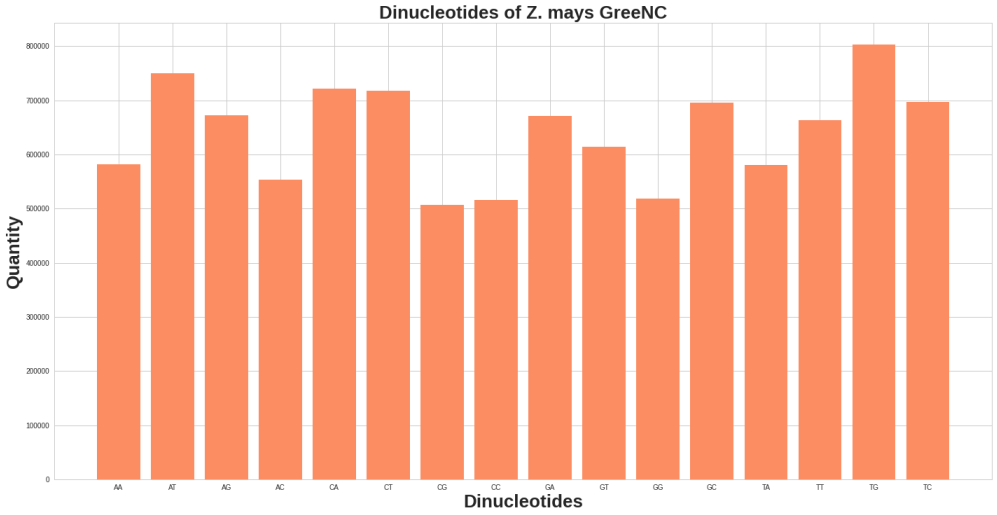


Figura 19 – Gráfico do Dinucleotídeo da *Z. mays* do PLncDB

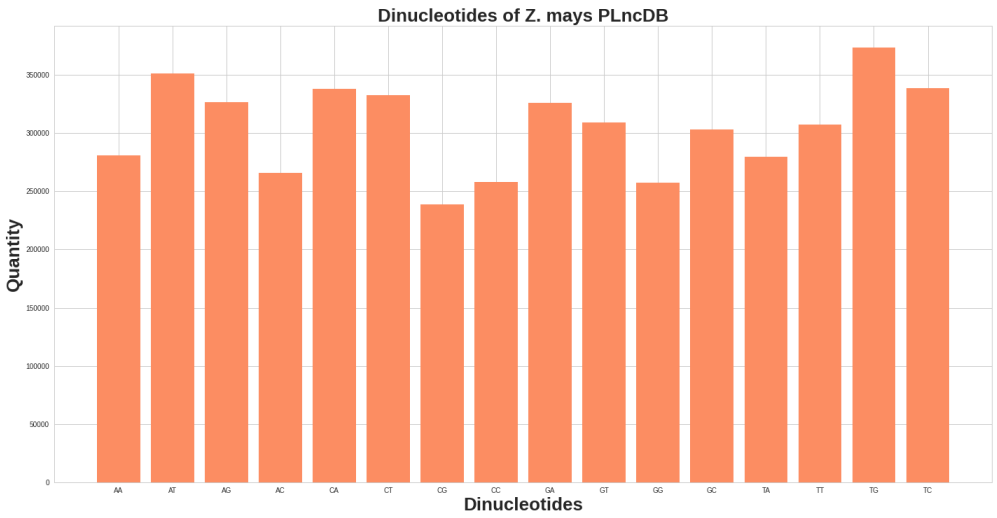


Figura 20 – Gráfico do Trinucleotídeo da *A. thaliana* do CANTATADB

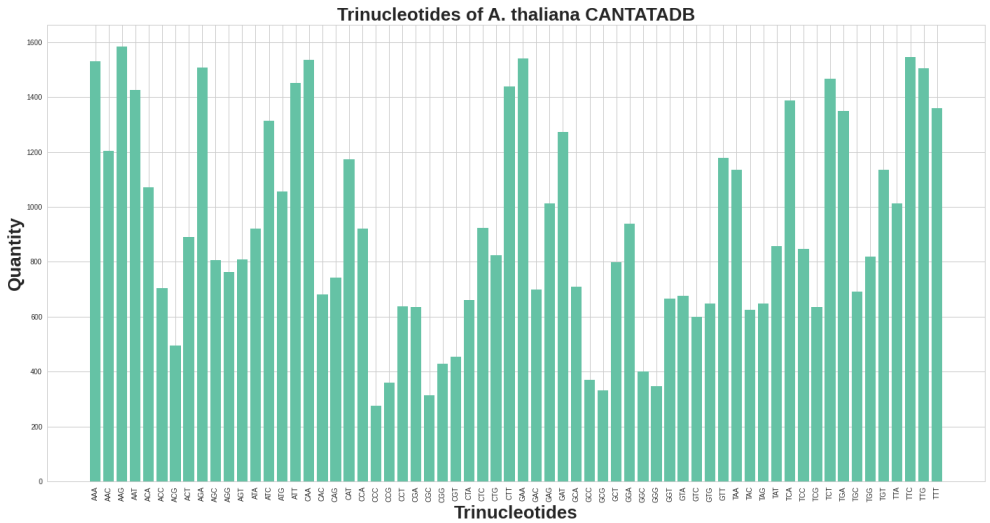


Figura 21 – Gráfico do Trinucleotídeo da *A. thaliana* do GreeNC

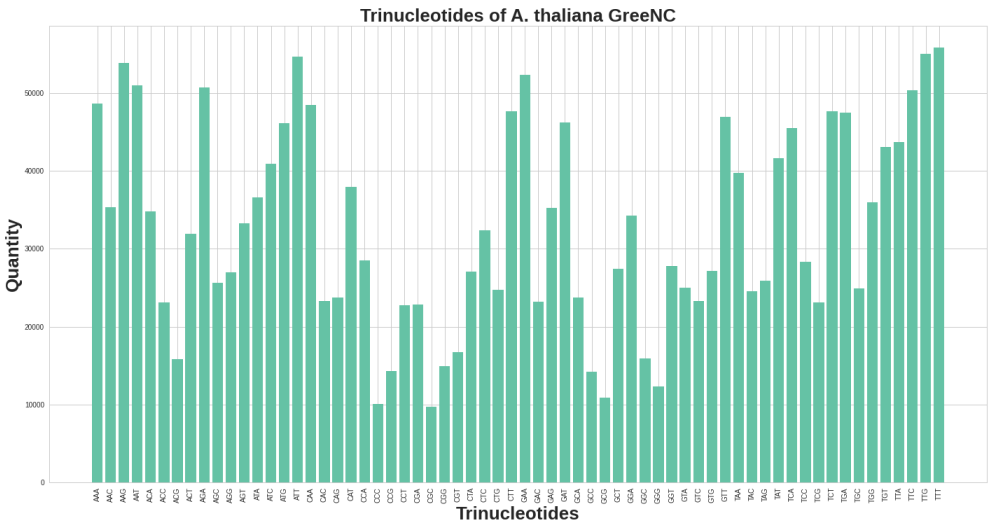


Figura 22 – Gráfico do Trinucleotídeo da *A. thaliana* do PLncDB

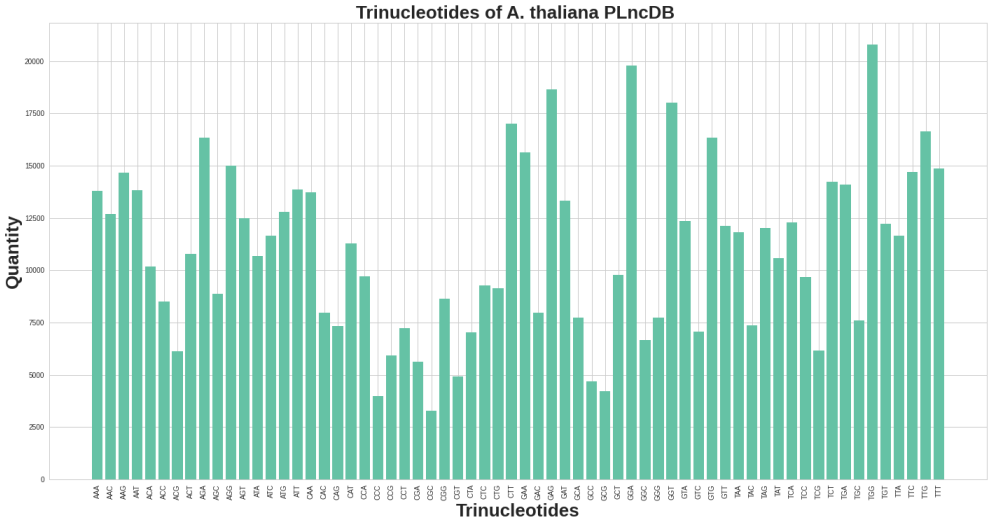


Figura 23 – Gráfico do Trinucleotídeo da *G. max* do CANTATADB

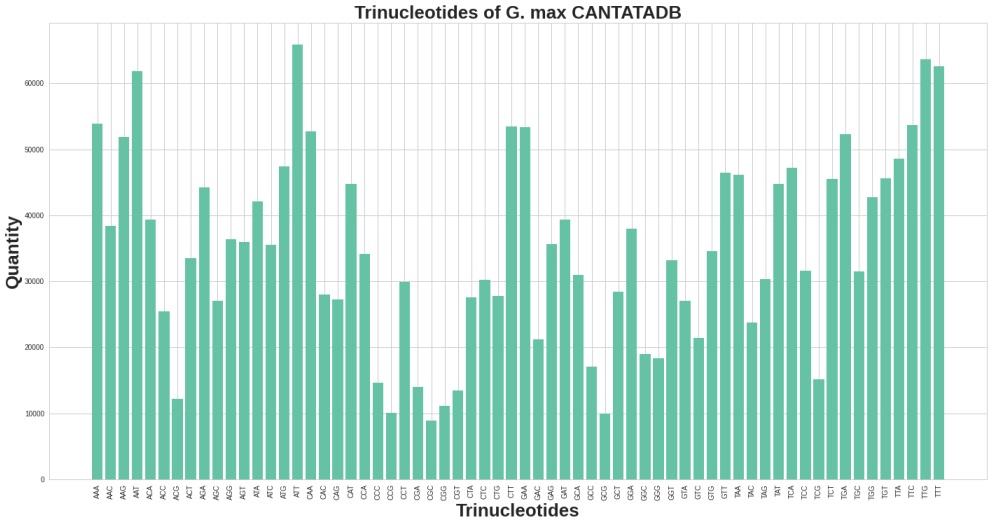


Figura 24 – Gráfico do Trinucleotídeo da *G. max* do GreeNC

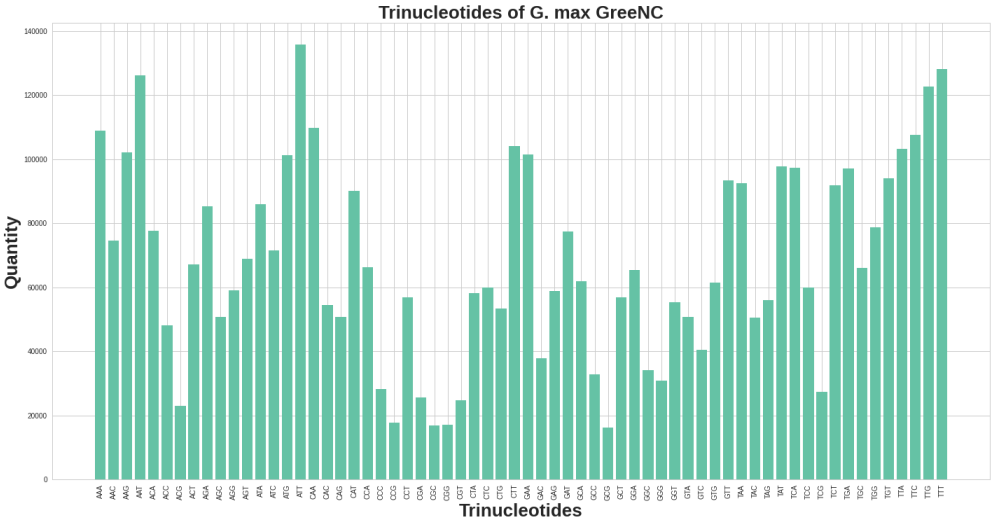


Figura 25 – Gráfico do Trinucleotídeo da *G. max* do PLncDB

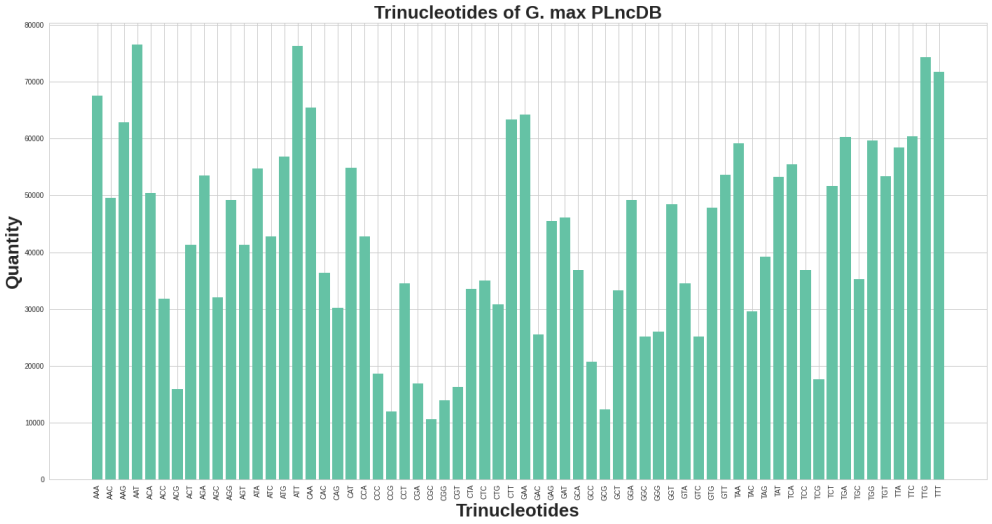


Figura 26 – Gráfico do Trinucleotídeo da *Z. mays* do CANTATADB

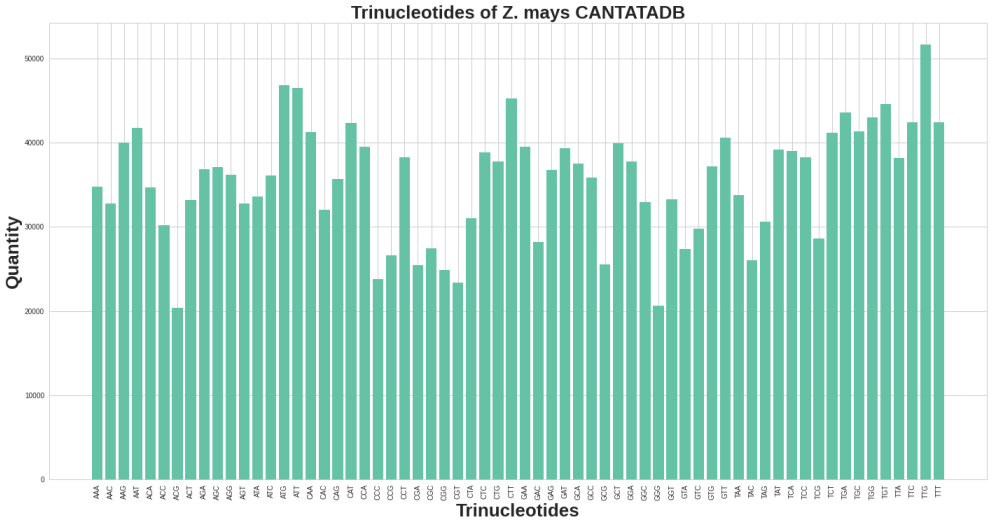


Figura 27 – Gráfico do Trinucleotídeo da *Z. mays* do GreeNC

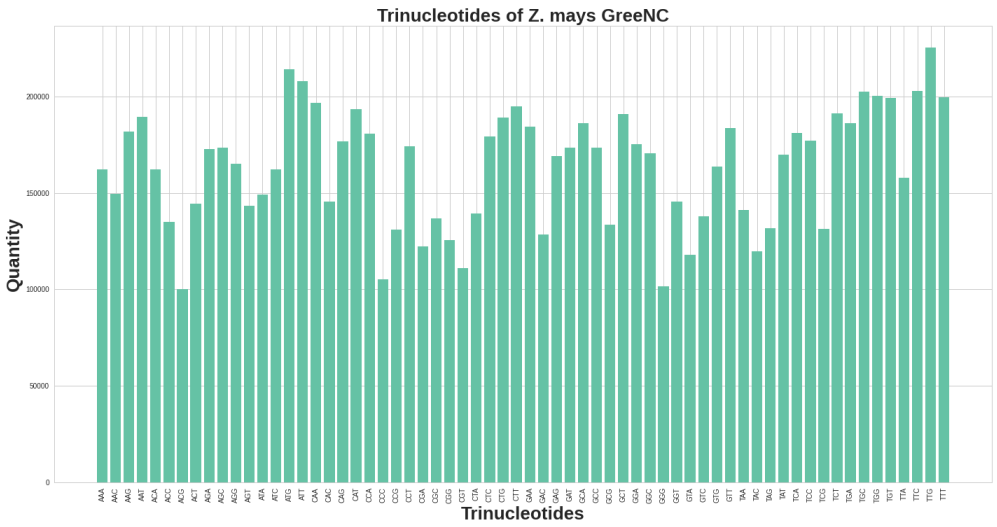


Figura 28 – Gráfico do Trinucleotídeo da *Z. mays* do PLncDB