PhageSNP: Prediction of the effect of hypermutable phage proteins along *Staphylococcus* genus

Alexandre Esperança

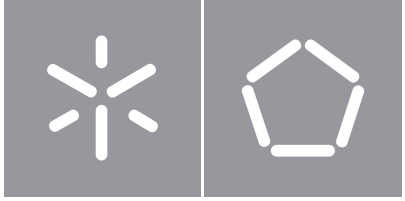
UMinho | 2023

University of Minho
School of Engineering

Alexandre Miguel Magalhães Esperança

PhageSNP: Prediction of the effect of hypermutable phage proteins along *Staphylococcus* genus

october 2023



University of Minho
School of Engineering

Alexandre Miguel Magalhães Esperança

PhageSNP: Prediction of the effect of hypermutable phage proteins along *Staphylococcus* genus

Masters Dissertation
Master's Degree in Bioinformatics

Dissertation supervised by
Luís Daniel Rodrigues de Melo
Óscar Manuel Lima Dias

october 2023

Copyright and Terms of Use for Third Party Work

This dissertation reports on academic work that can be used by third parties as long as the internationally accepted standards and good practices are respected concerning copyright and related rights.

This work can thereafter be used under the terms established in the license below.

Readers needing authorization conditions not provided for in the indicated licensing should contact the author through the RepositóriUM of the University of Minho.

License granted to users of this work:



CC BY-SA

https://creativecommons.org/licenses/by-sa/4.0/

Acknowledgements

I start by thanking my advisors, Professor Luís Melo and Professor Óscar Dias, for the support provided over the months for the project development. I appreciate all the efforts performed to surpass the numerous setbacks encountered during the paper development and the patience with my work evolution.

To my family and partner, I extend my deepest gratitude for your constant support during my most challenging times. Despite the stressful season, I always had my back on your aid, which made it possible to manage work and study life.

I will keep remembering the great moments with my classmates during the years of studying. I appreciate all the support in the various subjects that we faced and hope that I have provided you with equal consideration.

Thank you to my coworkers, who always understood my situation and helped me as much as possible. In particular, I thank Joel Carneiro for constantly providing me with external insight into the ongoing work and opinions on writing improvement.

Lastly, I would like to thank my family once more for allowing me to pursue my objectives and maintain my years of study despite the various issues. I will never be grateful enough!



This work was funded by national funds through the FCT - Portuguese Foundation for Science and Technology in the framework of the project DeathTrigger (project reference PTDC/BIAMIC/2312/2020)

Statement of Integrity

I hereby declare having conducted this academic work with integrity.

I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

University of Minho, Braga, october 2023

Alexandre Miguel Magalhães Esperança

Abstract

Bacteriophage therapy is an emergent alternative to treat highly resistant bacteria. It consists of applying bacteriophages to eliminate specific pathogenic bacteria in a patient. It portrays several advantages, but the co-evolution between virus and host regularly prevents the infection from the infective agent. Hence, it is imperative to investigate the proteins that drive recognition, specifically the Receptor Binding Proteins (RBPs). These proteins suffer numerous single-nucleotide polymorphisms (SNPs) and other mutations to accompany the adaptation rate of the bacteria phage receptors. Considering this, the PhageSNP tool aims to identify the mutations that promote this behavior. It is an annotation tool employing a Machine Learning mechanism to isolate all the RBP sequences within an unfamiliar genome. It also offers an in-depth analysis of all potential SNPs within each extracted sequence, assessing their predicted impact. We evaluate its RBP prediction performance on 100 random *Staphylococcus* bacteriophage genomes and its innovative SNP annotation feature as possible. In summary, PhageSNP provides a groundbreaking approach for the host range expansion analysis, exhibiting great RBP annotation results and SNPs sustained by robust theoretical research.

Keywords Bacteriophages, Receptor Binding Proteins, Single-Nucleotide Polymorphisms, Machine Learning, Substitution Matrices

Resumo

Terapia fágica é uma alternativa emergente para tratar bactérias altamente resistentes. Consiste na aplicação de bacteriófagos para eliminar bactérias patogénicas específicas num paciente. Possui diversas vantagens, mas a coevolução entre vírus e hospedeiro impede regularmente a infeção pelo agente infecioso. Portanto, é imperativo investigar as proteínas que controlam o reconhecimento, especificamente as Proteínas de Ligação ao Recetor (RBPs). Estas proteínas sofrem numerosos polimorfismos de nucleótido único (SNPs) e outras mutações para acompanhar o rácio de adaptação dos recetores bacterianos. Tendo isto em consideração, a ferramenta PhageSNP pretende identificar as mutações que promovem este comportamento. PhageSNP é uma ferramenta de anotação que emprega um mecanismo de Aprendizagem de Máquina para isolar todas as sequências de RBPs num genoma desconhecido. Também fornece uma análise aprofundada de todos os potenciais SNPs de cada sequência extraída, avaliando o seu impacto previsto. Nós avaliamos o desempenho na previsão de RBPs em 100 genomas aleatórios de bacteriófagos de *Staphylococcus* e o recurso de anotação de SNPs o quanto possível. Em suma, PhageSNP fornece uma abordagem inovadora para a análise da expansão do reconhecimento de hospedeiros, exibindo bons resultados na anotação de RBPs e SNPs sustentados por pesquisa teórica robusta.

Palavras-chave Bacteriófagos, Proteínas de Ligação ao Recetor, Polimorfismos de Nucleótido Único, Aprendizagem de Máquina, Matrizes de Substituição

Contents

I Introductory material 1

1 Introduction 2

1.1 Motivation . . . 2

1.2 Objectives . . . 3

2 State of the Art 5

2.1 Bacteria and antimicrobial resistance . . . 5

2.2 *Staphylococcus* spp. . . . 6

2.2.1 *Staphylococcus aureus* virulence . . . 6

2.3 Bacteriophages therapy as an antibiotic alternative . . . 7

2.4 Bacteriophages . . . 8

2.4.1 Bacteriophage morphologies and characteristics . . . 8

2.4.2 Bacteriophage life cycles . . . 10

2.5 Host recognition . . . 11

2.5.1 Receptor Binding Proteins . . . 13

2.5.2 Tail Fiber Proteins . . . 13

2.5.3 Tail spike Proteins . . . 14

2.6 Single Nucleotide Polymorphisms . . . 14

2.7 Bioinformatics Brief History . . . 15

2.8 Clustering and Alignment . . . 16

2.9 Machine Learning . . . 17

2.9.1 Feature Extraction . . . 18

2.9.2 Feature Selection . . . 19

2.9.3 Linear Models . . . 20

2.9.4 K-Nearest Neighbors . . . 21
2.9.5 Support Vector Machines . . . 21
2.9.6 Decision Tree . . . 23
2.9.7 Ensemble Models . . . 24
2.9.8 Deep Learning . . . 25
2.10 Substitutions Matrices . . . 28
2.11 Galaxy . . . 30

II Work Plan **31**

3 Methodology **32**
3.1 Dataset construction . . . 32
3.1.1 Positive Samples Extraction . . . 33
3.1.2 Negative Samples Extraction . . . 35
3.1.3 Sequences Analysis . . . 36
3.2 Receptor Binding Proteins Prediction . . . 37
3.2.1 Feature Extraction . . . 37
3.2.2 Model Development . . . 38
3.2.3 Model Validation . . . 39
3.2.4 Model Development Overview . . . 40
3.3 Single Nucleotide Polymorphisms Annotation . . . 41
3.3.1 Dataset Clustering . . . 42
3.3.2 Cluster Alignment . . . 42
3.3.3 Polymorphisms Extraction and Scoring . . . 43
3.3.4 Polymorphisms Analysis Overview . . . 46
3.4 PhageSNP Tool . . . 46
3.4.1 Usage Guidelines . . . 46
3.4.2 Model Application . . . 47
3.4.3 Tool Overview . . . 48
3.4.4 Galaxy Publishing . . . 48

4 Results and Discussion **50**
4.1 Constructed Dataset . . . 50

4.2 Evaluation of RBP Prediction Model 51
4.2.1 Dataset Labeling 51
4.2.2 K-mer Values Analysis 52
4.2.3 Final Model Results 53
4.2.4 Model Performance Assessment 56
4.3 SNP Scoring 59
4.4 Galaxy Application 61

5 Conclusions and future work **64**
5.1 Conclusions 64
5.2 Prospect for future work 64

List of Figures

1 Structure of the different phage divisions. a) Myovirus-like structure has a long tail, denoting a sheath component that confers the contractile feature to it; b) Shipovirus-like resembles myoviral morphology in the long tail characteristic, yet, the absence of sheath confers a flexible configuration; c) Podovirus-like morphology differs by its short and flexible tail. Image adapted from: 10.3390/v9060126 (1). 9

2 Schematic representation of the phage life cycles. (A) Phage adsorption to the host cell, consisting of the binding between RBPs and proteins in the bacteria surface; (B) Genetic material injection by the phage into the bacteria facilitated through the tail; (C1) Lytic pathway of the phage life cycle, comprises the replication of the injected DNA and host metabolism control take-over; (C2) Lysogenic pathway of the phage life cycle, whereas the phage DNA is incorporated into the host DNA or persists in the cytoplasm as a plasmid; (D1) Synthesis of the components, such as nucleic acids and proteins, essential to assemble the phage and posterior cell lysis by the action of enzymes like endolysins. (D2) Host cell replication with prophage transmission. Spontaneous induction may lead to prophage gene activation and cell lysis by produced phages. 10

3 Schematic illustration of myovirus-like phage adsorption to the host. Image adapted from: 10.1038/s41579-018-0070-8 (2). 13

4 Theoric representation of the k-mer method applied on a DNA sequence. The indicated "k" value is 3, extracting overlapping subsequences with the mentioned size. These comprehend the final vector relative to the sequence. Hereon, different techniques may apply for feature selection and application in ML model training. Image adapted from (3). 19

5 The different optimal hyperplanes encountered in the high-dimensional feature space according to the several kernel values applied. It differentiates the data point classes and attributes the prediction to unseen data of the located region. Image adapted from (4). 23

6 Structure overview of a Convolutional Neural Network, highlighting the role of each hidden layer leading up to the final prediction made by the output or fully connected layer. The hidden layers comprehend the convolutional and pooling layers, which are responsible for extracting feature maps from the data and reducing the dimensions of these, respectively. Image adapted from (5). 27

7 Image describing the general pipeline of the dataset construction. We begin with GenBank accession ID identification corresponding to the "Receptor Binding Proteins" or associated keywords. This extraction occurs in the most known protein databases, such as NCBI and UniProt. We retrieve the corresponding protein and nucleotide sequences from the previous accession IDs. Hereafter, several sequence analysis steps occur to make the final dataset as robust and complete as possible. 32

8 Schematic representation of the feature extraction process. (1) K-mer parsing of the DNA sequence, resulting in hexamers subsequences; (2) Aggregation of all hexamers into a single string; (3) Obtaining the matrix token count by measuring k-mer frequencies with the aid of the CountVectorizer class. 38

9 Model development overview resuming the main steps performed to achieve the optimal choice for RBP prediction. 1) Comprehend the procedures taken to collect all the RBP sequences existent in *Staphylococcus* phages, executing distinct approaches for positive and negative sample extraction; 2) Method for feature extraction where we employ the k-mer counting strategy; 3) Possible models assessment through training and testing of multiple supervised ML approaches, complemented with hyperparameter tuning; 4) Final consideration of the testing results to choose the optimal model for our objective; 5) Best model employed to predict RBP sequences. 41

10 Overview of the cluster selection method implemented. We compare the inputted sequence by the user with all the cluster representatives previously defined through the CD-HIT tool. We execute this comparison through Needleman-Wunsch or global alignment, which provides an alignment score proportional to the sequence similarity. Thus, the highest score represents the best cluster to integrate the sequence. 43

11 Overview of the SNP scoring strategy implemented. Firstly, the cluster alignment is analyzed to extract all the possible candidate SNPs. The Alternative Allele Frequency equation indicates whether a site *b* is a candidate by the frequency value exceeding the defined threshold. Hereon, we uncover the altered amino acid resulting from the mutation and cross-reference it among all the substitution matrices considered in the paper. Each matrix has a weight value according to its relevance, and its appliance with the average calculation results in the final score that reveals the theoretical SNP impact value. 45

12 Overview of the executed steps to analyze the SNPs within the RBPs. 1) Retrieval of the best cluster to integrate the RBP sequence; 2) Multiple alignment process of the several sequences within the cluster and the initial sequence; 3) Mutation identification through the analysis of the produced alignment and verification if it represents a candidate SNP; 4) SNP scoring through the value consideration of multiple substitution matrices; 5) Relevant SNP filtering according to the defined impact value thresholds. 46

13 Tool overview resuming the principal steps executed to generate the output file of SNP scoring of each RBP sequence. 1) Given the input sequence or genome by the user, the Pyrodigal package returns the ORFs within the genome if applied; 2) Our chosen ML model predicts and filters the existent RBPs; 3) SNP identification according to the default threshold value or the inputted by the user, aided by clustering and alignment methods. Lastly, SNP scoring according to multiple substitution matrices; 4) Final pretty result, detailing the RBP nucleotide sequence, SNPs found, and respective impact score. 48

14 ToolShed preview of the PhageSNP interface, detailing the input parameters and the package requirements for normal execution. Moreover, it provides a simple interpretation and information about the output files. Page available through the URL: `https://toolshed.g2.bx.psu.edu/view/phage_snp/phage_snp/465bfb24dcc5` . . 49

15 Labeling class values. Due to the low occurrence frequency of the "RBP" classification, we opted not to include it to prevent the trained model from underfitting. As a result, only the remaining three classifications are taken into consideration. 52

16 Boxplots that visually represent the impact scores of SNPs referenced in prior studies. Boxplot A represents the currently analyzed SNPs from phages that specifically infect *Staphylococcus spp.*, whereas B includes boxplot A plus other polymorphisms from phages with different host recognition. Relative to A, we observe a distribution of the scores between 1 and -4, with interval expansion when considering other species hosts. Additionally, we highlight in red the SNPs that the PhageSNP tool successfully identified as candidate polymorphisms. We computed all the obtained scores with the SNP scoring method implemented within PhageSNP. 61

17 Visual representation of the PhageSNP tool on the Galaxy platform. We present a detailed description of its functioning, documenting package requirements and the input/output data. As input, it is instructed an unchanged FASTA with the sequence, indication of genome analysis, and the threshold value to consider. Page available in the URL: `https://galaxy.bio.di.uminho.pt/root?tool_id=toolshed.g2.bx.psu.edu/repos/phage_snp/phage_snp/PhageSNP/0.1.0` 62

18 An illustrative example of an output file with the genome analysis considered. It includes detailed information about the classification, location, and corresponding ORF sequence of the anticipated RBPs in the analyzed genome. Furthermore, for each predicted SNP, we document the nucleotide and amino acid changes, its position in the sequence, and the impact score on the resulting protein. 63

List of Tables

1 For each CD and Pfam candidate considered for RBP sequence mining, we display its comprehensive information, including the analyzed accession IDs and corresponding candidate descriptions. These accession ID values correspond to the date of 25/06/2023. 34

2 Methods employed for positive sample sequences extraction, serving to maximize the sub-dataset size. We highlight the number of retrieved accession IDs, resources applied, and pertinent observations. The adopted strategies are (1) Entrez search mechanism using the nucleotide database to extract genome accession identifiers associated with the term "staphylococcus phage"; (2) Restrictive query search in the UniProt web platform and output JSON algorithm analysis; (3) Search sequences included in the most recurrent CDs and Pfams within the dataset and associated to RBPs; (4) Query advanced search in NCBI protein database, restraining the records to *Staphylococcus* genus and protein description including one of the RBPs keywords. 35

3 Each ML technique tested in the model development, denoting the default and tuning required hyperparameters. The table contains Supervised ML models, including several NN strategies. Notably, the CNN model has the highest hyperparameter variation as the model is theoretically the most suitable DL model for our paper's objective. 39

4 The weights employed in each variation of BLOSUM and PAM matrices, with greater weights assigned to the ones representing higher sequence relationships. BLOSUM proximity relevance is proportional to its representative number, whereas PAM has the opposite behavior. Therefore, the most significant matrices are BLOSUM90 and PAM100. 44

5 The weights implemented for the PFASUM and VTML matrix variations. Like BLOSUM and PAM, the matrices for closer homology have higher weights to exert more influence in the final matrix value calculation. Thus, PFASUM60 and VTML20 are the most significant matrices. 44

6 Table representation of each method sequence results. Finally, the dataset encompassed 2718 samples, further reduced to 2000 by the validation dataset approaches. The 718 removed samples correspond to accession IDs retrieved by the fourth process, in which we only identified the amino acid sequences and failed to extract the nucleotide sequence through the complex emplaced strategies. 51

7 F1 scores comparison for each trained model and dataset with variation in the k-mer size. Dataset A corresponds to the one encompassing 2,000 negative samples, whereas dataset B represents the dataset with 4,000 "Not RBPs". We observe distinct results according to the model used, displaying diverse relations with the length variation. Ultimately, k-mer size of 5 portrayed the best F1 scores, promoting its choice as the parameter value for the final model development. 53

8 A summary of each model metrics score, displaying the optimal hyperparameters when applied. Dataset A corresponds to the one encompassing 2,000 negative samples, whereas dataset B represents the dataset with 4,000 "Not RBPs". The SVC model exhibits the best score on dataset A with 0.963, establishing as the best approach for our tool RBP prediction feature. Relative to all the models, the linear ones produced the best results, whereas KNN was the worst. Finally, RF and DT yielded satisfactory results, with RF standing out. 55

9 A summary of each model metrics score, displaying the optimal hyperparameters when applied. Dataset A corresponds to the one encompassing 2,000 negative samples, whereas dataset B represents the dataset with 4,000 "Not RBPs". These DL approaches are taken posteriorly to the conclusion of the best ML model for the PhageSNP tool incorporation. Therefore, they are potential methods to improve the PhageSNP's ability to predict RBP sequences in the future. Specifically, the CNN model exhibits the best score of 0.968 with dataset A, establishing the best approach to integrate the tool RBP prediction feature. 56

10 Table summary of the statistics observed in the validation process of the model developed to predict RBPs in Staphylococcus phages. We analyzed 12,011 sequences, with a majority belonging to the category of "No Classification" or ORFs apart from RBPs. This class demonstrates high efficiency, proving the model works well in discerning RBPs of other sequences. Relative to the tail spikes, they appear in much lower frequency with only three occurrences and accurately labeled in the validation process. Tail fibers are the most prominent among the RBPs and display difficulty distinguishing them from close-related tail sequences of other functions. However, the correctly predicted tail fibers exceed more than double the incorrectly labeled, being the misclassified composed of inconclusive sequence functions and similar ORFs to RBPs. 58

11 Confusion matrix enumerating all the outcome values obtained from the actual and predicted classifications. The outcome classification denotes the correct/wrong indication of the presence/absence of the studying characteristic. Therefore, we may use them to analyze the model's performance on multiple standards by metrics calculation. The accuracy metric can be misleading since it is susceptible to unbalanced data. This case study exemplifies the presence of unbalanced data, characterized by the infrequent appearance of RBP sequences within a genome. Thus, the remaining metrics are noteworthy because they provide better insight into the model's behavior. F1-score and MCC (Matthews correlation coefficient) are especially relevant since they embody the most effective and robust metric methods. With respective scores of 0.78 and 0.79, our model demonstrates the potential for significant improvement. 59

Acronyms

DNA Deoxyribonucleic Acid

dsDNA double-stranded Deoxyribonucleic Acid

RNA Ribonucleic Acid

UV Ultraviolet

NCBI National Center for Biotechnology Information

EMBL European Molecular Biology Laboratory

EMBL-EBI EMBL's European Bioinformatics Institute

ORF Open Reading Frame

JSON JavaScript Object Notation

MUSCLE MUltiple Sequence Comparison by Log- Expectation

MAFFT Multiple Alignment using Fast Fourier Transform

MCC Matthews correlation coefficient

Part I

Introductory material

Chapter 1

Introduction

1.1 Motivation

Bacteria play a fundamental role in ecosystem and are essential for the normal functioning of the human body. The human microbiome comprises numerous commensal bacterial species with diverse functions, including food digestion aid and providing body defense. However, pathogenic bacteria can also cause significant health problems and even death. Bacterial infections' constant occurrence, a persistent issue throughout history, has led to the research and development of various reduction strategies. Adequate hygiene and cleanliness practices are crucial in reducing the spread of infections. With this, antibacterial methods, such as antibiotics and medicinal plants, are commonly used to combat persistent and infectious pathogens. When done in restricted quantities, the administration of antibiotics can be effective, but overuse and misuse can lead to the emergence of antibiotic-resistant bacteria strains (6). This phenomenon of bacteria resistance growth is due to its high replication rate and various factors that induce genetic variation (7). Consequently, large outbreaks of bacterial infections in a specific population can lead to the emergence of community-associated strains, which can become major public health threats, particularly in healthcare settings. In light of this, alternative approaches as phage therapy, have emerged as promising solutions.

Bacteriophages, often referred to as phages, are viruses that infect and thrive inside bacteria cells. These unique organisms infect and use bacteria as their host cells, replicating until their lysis. They are among the most abundant and variable entities on Earth, such that distantly related phages often have little to no genome similarity (8). Moreover, co-evolution between phages and bacteria is an ongoing and massive process driven by selective pressures acting on both organisms (9). The field of phage therapy, the application of phages to treat bacterial infections, is gaining momentum as an alternative to traditional antibiotics due to the development of antibiotic-resistant bacteria. Accordingly, investigations carefully se-

lect and manipulate the phages to ensure patient safety and treatment effectiveness. The most important characteristics to evaluate for this application are the phage's fitness and interactions with its host bacteria. Consequently, analyzing the phage genome and identifying factors that influence variability, such as SNPs, recombination events, and horizontal gene transfer, are crucial for the successful implementation of phage therapy.

Phage therapy is an alternative approach that could be a game changer in treating bacterial infections, mostly when antibiotics and other antibacterial methods are no longer effective. Genetic analysis, as SNPs analysis, of crucial binding and infection proteins promotes the enhancement of phage characteristics, mainly of host range recognition. This approach is critical in scenarios, where traditional methods have failed, providing a new avenue for treatment. However, phage-restrictive recognition of hosts limits their application as certain bacteria strains evolve to modify their surface receptors. Ergo, specific bacterial infections become untreatable due to a lack of effective phages.

Regarding SNP evaluation, Machine Learning (ML) models hold significance for predicting advantageous SNPs for enhancing phage characteristics (10). The assessment of proteins that recognize the host bacteria is the center of attention for improvement, as polymorphisms promote the host range expansion (11). Herewith, the advanced predictive ML models guide the phage genome manipulation sites. Thus, gaps in phage therapy are filled by using modified phages specific to the target bacteria, improving its elimination.

1.2 Objectives

This project aims to shed light on the effect of various SNPs on the phage features, such as their interactions with the host and their scope of infection. Thus, ML models can assess potential mutations and their corresponding phenotypes. Accordingly, gathering the genomes of staphylococcal phages from various databases is necessary. Afterward, the alignment and clustering of protein groups simplify SNP visualization along the sequences. For this matter, using several bioinformatics tools and automating processes through algorithms becomes crucial for dataset elaboration. Finally, the appraisal of the most appropriate ML model to apply and its training with the dataset will provide a means of predicting the effect of SNPs on phage proteins.

The overarching goal of this study is to create an ML model capable of predicting the effects of SNPs on phage proteins. For this purpose, several previous steps require detailed attention, such as:

- Dataset construction: For the training of a ML model, the creation of a large and complete dataset is critical.
- RBP annotation: RBP sequence extraction in a genome is fundamental to isolate the relevant nucleotide sequences for further analysis.
- Machine Learning model: Data feature extraction and selection paired with ML model examination. To develop the most effective predictive algorithm, multiple models must undergo evaluation.
- Analysis of predicted proteins: Determining potential SNPs and assessing their influence on the host range recognition of RBPs.

Chapter 2

State of the Art

2.1 Bacteria and antimicrobial resistance

Antibiotics were introduced in human medicine at the beginning of the 20^{th} century. The salvasarn, an medicament commercialized in 1910 to cure syphilis, is the first efficient way to combat human infections, specifically caused by *Tryopanosoma brucei*. However, the discovery of penicillin, distinctly referred to as benzylpenicillin, in 1928 is considered the pivotal moment in bacterial infection treatment (12). The "golden" accident by Dr. Alexander Fleming allowed the uncovering of penicillin and, herewith, the evolution of the β-lactam antibiotics. Hereon, several new members of these antimicrobial groups were developed to broaden their range of activity and counteract the emergence of resistance (13).

As the use of antibiotics in medicine increased, the occurrence of antibiotic-resistant bacteria strains proceeded. The upsurge of antibiotic-resistant bacteria is a serious cause for concern, as previously controlled bacterial infections are returning as significant public health threats. The bacteria great source of genetic diversification such as mutations or horizontal gene transfer made it possible to develop numerous mechanisms to render antibiotics useless (14), leading to a pressing need for the development of new and potent alternatives to diminish them. For instance, β-lactam antibiotics act upon the bacterial cell wall, binding to penicillin-binding proteins and interrupting the cell wall formation. Mutations in these receptors obstruct the antibiotics' means of action, making them the primary cause of β-lactam resistance (13). Alternatively, β-lactamases are immensely challenging enzymes that deactivate β-lactam antibiotics, serving as another source of resistance development (15). Sadly, the prevalence of resistant strains is a critical apprehension in bacteria groups such as *Staphylococcus* spp., which can deftly spread throughout the population (16).

2.2 *Staphylococcus* spp.

Staphylococci organisms are Gram-positive bacteria widely known for causing various infections, such as respiratory, skin, soft tissue, bloodstream, and digestive disorders (17; 18; 19). These rapidly spread among humans and animals, impacting animal husbandry and society's economy (20). The *Staphylococcus* genus encompasses species such as *S. aureus*, *S. epidermidis*, *S. lugdunensis*, and others. They spread through direct and indirect contact between individuals, being present freely in the environment. A great source of staphylococci infections is food contamination. Multidrug-resistant *Staphylococcus* spp. have been encountered in fruits and vegetables with elevated resistance rates to certain β-lactam antibiotics (20). Besides, these produce dangerous proteases that weaken the host immune system and host tissue degradation, often resulting in skin edemas (21). Among the species within the genus *Staphylococcus*, *S. aureus* is particularly notable for its antibiotic resistance and virulence (22).

2.2.1 *Staphylococcus aureus* virulence

Staphylococcus aureus, a member of the *Staphylococcaceae* family, has been identified as a dangerous and persistent pathogen in humans for decades. Reports from the late 90s highlight the emergence of multidrug-resistant strains of *S. aureus* (23), which continue to pose a global health threat today. This Gram-positive bacterium is harmless to approximately 30% of healthy individuals. However, it is responsible for causing a wide range of human diseases as skin infections, abscesses, and necrotizing pneumonia (22). The most common form of colonization is in the nasal passages as a commensal organism, existing in the microbiome of 20% of the population (24). Moreover, it is also known to colonize the human skin and gastrointestinal tract (25). Studies have shown a change in gene expression when *Staphylococcus sp.* transitions from a commensal state to a pathogenic state, with different expression patterns observed among different strains (26).

Before the discovery of penicillin, *S. aureus* had a mortality rate of nearly 80%. As one significant outbreak in medicine, methicillin is a small-spectrum β-lactam antibiotic belonging to the penicillin group. It became the primary way to combat the bacteria, dramatically decreasing its pathogenicity. However, constant and uncontrollable usage led to the emergence of a resistant strain, the Methicillin-Resistant *Staphylococcus aureus* (MRSA).

In recent years, new strains of MRSA known as Community-Associated MRSA (CA-MRSA) have become an epidemic in many parts of the world. CDC (Centers for Disease Control) and WHO (World Health Organization) even considered it an urgent threat to public health (25; 27). These strains contain unique

genetic traits that improve bacteria infection rate, such as a mobile genetic element known as staphylococcal cassette chromosome *mec* (SCC*mec*) and the Panton-Valentine leukocidin (PVL) cytotoxin (28; 29). PVL is a toxin that can destroy human macrophages, leukocytes, and monocytes. Besides, it triggers an inflammatory immune response and tissue damage (8; 30). Recent research has shown that the genes responsible for PVL production, *lukS-PV* and *lukF-PV* (30), are present in both MRSA and MSSA (31), although they are more common in MRSA (32). However, the role of PVL in the spread and severity of CA-MRSA is not entirely clear. Currently, diverse antibiotic combinations are frequent to treat different MRSA groups and differ according to the type of infection (19). Yet, the indicated antibiotic guidelines lack efficiency and warrant a thorough examination for any potential side effects. As a result, finding alternative treatments for them becomes increasingly imperative.

2.3 Bacteriophages therapy as an antibiotic alternative

Phage therapy has emerged as a promising alternative to traditional antibiotics treatment of several infectious bacterial strains (11). These can effectively reduce deeply the population of a bacteria (8), but they can also transfer genes to the host bacteria, including genes that confer antibiotic resistance. Consequently, phages used in therapy should not contain genes that could be horizontally transferred to the host genome (11), such as virulence or drug resistance factors. Multiple phage may be administered together or in sequence for more effective treatment, carrying only lytic cycle genes to accomplish the goal of eliminating the bacteria without transferring any harmful genes (11).

In recent years, phage cocktail has been much more resorted to treating severe bacterial infections. Several benefits are noted, such as broader strain coverage and less resistance development from the bacteria. Accordingly, mutant phages extremely specific for species or strain exhibit worse results in eliminating a bacterium than its application grouped in a cocktail (11). As mentioned, co-evolution between bacteria and phage is persistent and constant. Thus, as with antibiotics, bacteria can evolve to become resistant to phages, rendering phage therapy ineffective. Therefore, bacteria treatment is preferable with a phage cocktail, since it is less conducive to developing resistance (33).

Concerning *S. aureus*, several phages have been identified as effective in its elimination. Phages 812h1 and ϕMR11 are excellent examples of MRSA treatment success (11; 34). Phage 812h1 is a mutant from wild-type 812 that suffered countless single-nucleotide polymorphisms. An investigation ascertained its efficiency and verified that over 90% of 186 different MRSA strains were susceptible to it. On the other hand, phage ϕMR11 exhibited outstanding results against certain reduced MRSA strains in mice therapy.

Therewithal, both phages are absent of any gene for antibiotic resistance or virulence.

However, many different bacteria families threaten human health, requiring extensive distinct phage analysis due to their high specific host recognition. *Bacillus cereus*, a Gram-positive bacterium known for causing food poisoning and other infections (35), has already been fought with antibiotics aided by phages' action. Bc431v3 phage is a strong lytic phage with great host range recognition between the *Bacillus* species, particularly *Bacillus cereus* (36), having great potential for vast therapy application. Contrarily, phage PBC1 has an extremely narrow host range to certain strains of *Bacillus cereus*. Regardless, it has an essential role in a phage cocktail of killing a distinctive bacteria strain not aimed by other phages (37).

2.4 Bacteriophages

Phages are the most abundant entities (38) in the biosphere and specifically use bacteria as their hosts. They have a high degree of genetic diversity and use various mechanisms to promote variability and share reduced genomic similarity with different host genera (8). Phages classification occurs in several different ways, including based on their shape, host range, and nucleic acid type. Shape classification is a standard method of categorizing and is divided into four main categories: tailed, polyhedral, filamentous, and pleomorphic. (39). Tailed phages are the most common type and belong to the *Caudoviricetes* class. They can divide into three different structures: myovirus-like, siphovirus-like, and podovirus-like. Each arrangement has a distinct tail structure: myoviruses have long, contractile tails; siphoviruses have long, flexible tails; and podoviruses have short, flexible tails (40). Additionally, all tailed phages have double-stranded DNA (dsDNA) and icosahedral capsids.

2.4.1 Bacteriophage morphologies and characteristics

The phage families are constantly updating, requiring regular verification of genomic and morphology differences that promote the appearance of new families and the disposal of previous ones. For this reason, we are considering the previously mentioned divisions to ascertain the morphological differences between the phages. These correspond respectively to the discontinued *Myoviridae*, *Siphoviridae*, and *Podoviridae* families (41).

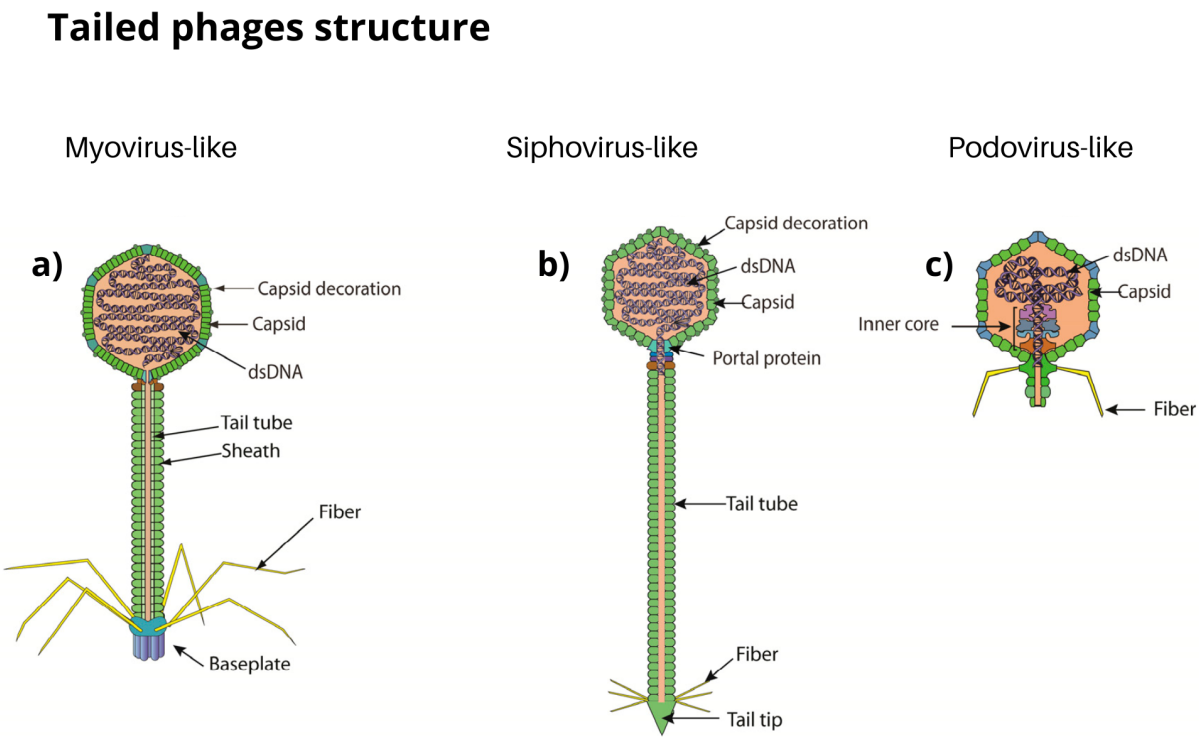


Figure 1: Structure of the different phage divisions. a) Myovirus-like structure has a long tail, denoting a sheath component that confers the contractile feature to it; b) Shipovirus-like resembles myoviral morphology in the long tail characteristic, yet, the absence of sheath confers a flexible configuration; c) Podovirus-like morphology differs by its short and flexible tail. Image adapted from: 10.3390/v9060126 (1).

The tail is the primary source of differentiation between the morphologies presented. It is formed by a joint of filamentous proteins, acting as a linkage between the dsDNA-containing capsid and the distal end of the tail (42). Furthermore, the distal termination exhibits RBP responsible for host cell recognition and adsorption (43). Relatively to myovirus-like structure, the genome tends to be larger than other phages (44). It differs by containing a large and contractile tail with a sheath around its tube, serving as an interface for head attachment (45). In addition, similar to the siphovirus-like structure, they generally contain baseplate and tail tip components. Tail tip complexes (TTC) are frequently referenced as baseplates and consist of protein structures that aid in the host cell adsorption and genome injection, besides functioning as attachment platforms for the RBPs (46). Hereupon, siphoviral morphology lacks the sheath feature resulting in its flexible tail characteristic. Lastly, the podovirus-like structure stands out for its short or absence of a tail (47).

2.4.2 Bacteriophage life cycles

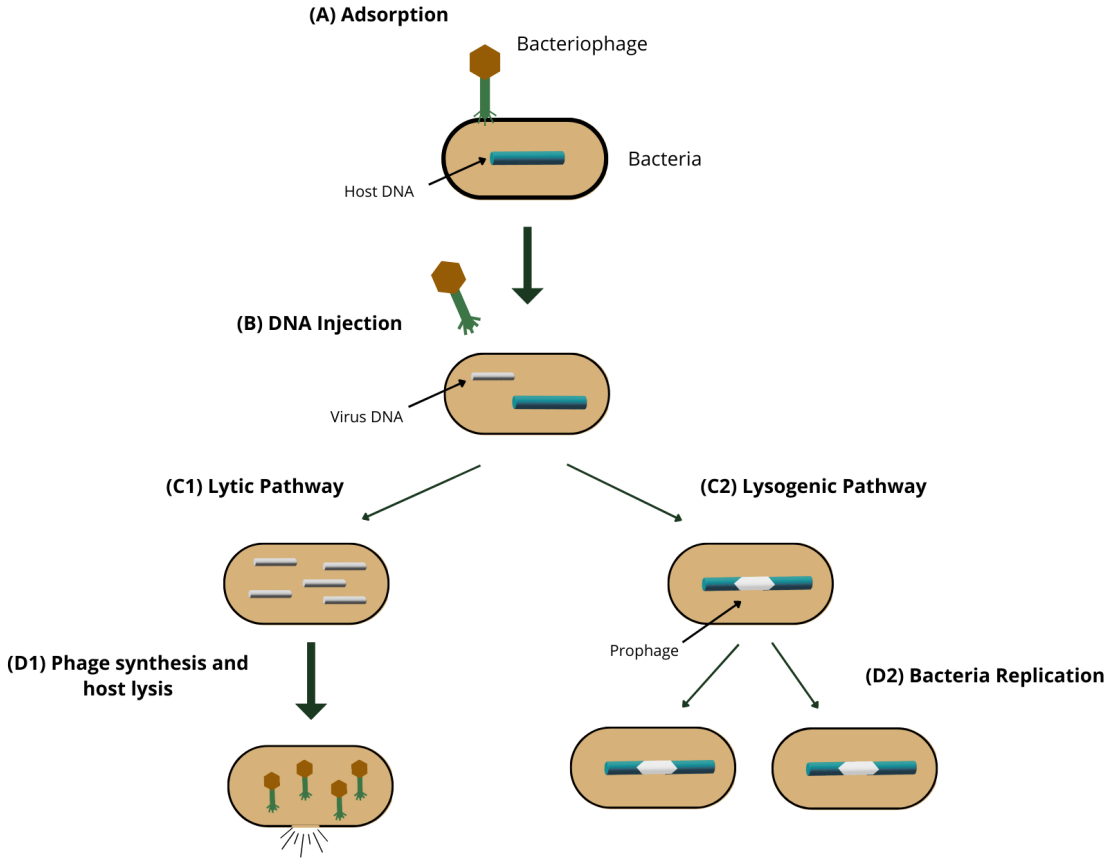


Figure 2: Schematic representation of the phage life cycles. (A) Phage adsorption to the host cell, consisting of the binding between RBPs and proteins in the bacteria surface; (B) Genetic material injection by the phage into the bacteria facilitated through the tail; (C1) Lytic pathway of the phage life cycle, comprises the replication of the injected DNA and host metabolism control take-over; (C2) Lysogenic pathway of the phage life cycle, whereas the phage DNA is incorporated into the host DNA or persists in the cytoplasm as a plasmid; (D1) Synthesis of the components, such as nucleic acids and proteins, essential to assemble the phage and posterior cell lysis by the action of enzymes like endolysins. (D2) Host cell replication with prophage transmission. Spontaneous induction may lead to prophage gene activation and cell lysis by produced phages.

The phage life cycle consists of phage adsorption to the host cell, followed by DNA injection, phage replication, and cell lysis. However, its behavior in the bacteria differs between the life cycle pathways, lytic and lysogenic.

Phage adsorption to the bacteria cell is highly dependent on the nature of the bacteria and its typical receptors on the wall surface. It is due to the morphological differences between Gram-negative and

Gram-positive bacteria. Respectively, they are distinguished for owning lipopolysaccharides and teichoic acids generally acting as wall receptors. The adsorption phase defines the phage-host specificity since successful genome injection is critical to continue phage virulence and survival (48). Adsorption is divided into two types, reversible and irreversible, whereas the difference remains in the attachment persistence after contact, being the second permanently bound to the surface. In Gram-positive bacteria attachment, tetrapeptide presence enables an irreversible coupling to the teichoic acids receptors (48). Besides, surrounding environment may also interfere with the adsorption type since no disturbance increases the frequency of irreversible attachments (49).

Afterward, it proceeds the host cell penetration and further phage DNA injection. Several enzymes, such as lysozymes located at the phage tail, hydrolyze linkages of the cell wall, degrading it and aiding in the cell envelope penetration (50). Hereon, DNA injection is favored and enters the bacteria cytoplasm. Hence, the life cycle differentiates into two main pathways: lytic and lysogenic. The lytic pathway comprehends the phages that commit to the phage genome replication and phage assembly through host metabolism takeover and final host cell lysis. Lysogenic phages remain in a dormant state that integrates the host genome and may become lytic by an induction event. It consists of the destruction or deactivation of the phage repressor due to various stress factors, ranging from UV light exposure or the presence in the environment of phage-inducing factors (51). Additionally, phages may exhibit a behavior known as pseudolysogeny, in which they exist as a plasmid and multiply within the cell.

Concerning phage therapy application, phages with a lysogenic life cycle are the least appealing ones, since they allow the bacteria to maintain their standard replication and require a stimulus to destroy it. On the contrary, lytic phages rapidly promote the destruction of the host bacteria, therefore being the best alternative for rapid pathogen elimination.

2.5 Host recognition

Phage therapy has as its main hardship the recognition of the pathogen infecting the patient due to the high specificity of RBPs and bacteria surface receptors. Therefore, host recognition is the primary research topic in phage therapy applications, being required enhanced methods for RPBs structure and sequence analysis, as well as rapid phage engineering (52). Concerning bacteria receptors, they differ according to the type of bacteria to recognize, Gram-negative or Gram-positive. The first contains an outer membrane and a thinner peptidoglycan layer. Meanwhile, the second holds a thicker peptidoglycan layer but no outer layer. These are susceptible to different antimicrobials due to their morphologic discrepancies and

antimicrobials' action methods (53).

Gram-positive bacteria's thick peptidoglycan layer confers the unique characteristic of being stained as violet when applied with crystal violet and safranin. Peptidoglycan is a heteropolymer made up of sugars, such as *N*-acetylglucosamine (NAG) and *N*-acetylmuramic acid (NAM), and is the main component of the bacteria cell wall. On its surface, its encountered receptors such as proteins, teichoic acids (TA), and lipoteichoic acids (LTA). Protein receptors have several functions as structuring and transporting (48), but the teichoic acids are the distinctive component of this type of bacteria. TA are water-soluble polymers assembled of repeating units of glycerol or ribitol, connected by phosphodiester bonds (48; 54). LTA has a similar composition and transposes the peptidoglycan layer, anchoring to the phospholipid bilayer via a lipid anchor (54). Whereas TAs covalently link to the peptidoglycan layer. Hence, these serve as a meaningful connection for phages to adsorb to the bacteria (55).

Contrarily, Gram-negative bacteria are stained red due to their thinner peptidoglycan layer. Noticeably, it carries an increased number of integral proteins, conferring a higher permeability characteristic than the Gram-positive bacteria (48). As aforementioned, they distinguish for the presence of an outer membrane absent in their bacteria counterparts, followed by the thin peptidoglycan cell wall and an inner membrane. Here it is denoted the presence of lipopolysaccharides (LPS) that divide into three segments: O-antigen, core, and lipid A (56). LPS is a polymer comprised of monosaccharides and fatty acids, divided into two types: smooth, which contains the previously mentioned structure, and rough, lacking the O-antigen component. Due to the presence and high variability of the O-antigen, the smooth LPS type has the narrowest host range specificity among the two types. Herewith, phage adsorption to rough LPS type eases by the conservative structure of the core segment (48; 55). Interestingly, phages may also use bacteria pili, flagella, and capsular polysaccharides as binding receptors (48; 57).

In this investigation, we will emphasize Gram-positive bacteria due to our primary focus on the *Staphylococcus* genus.

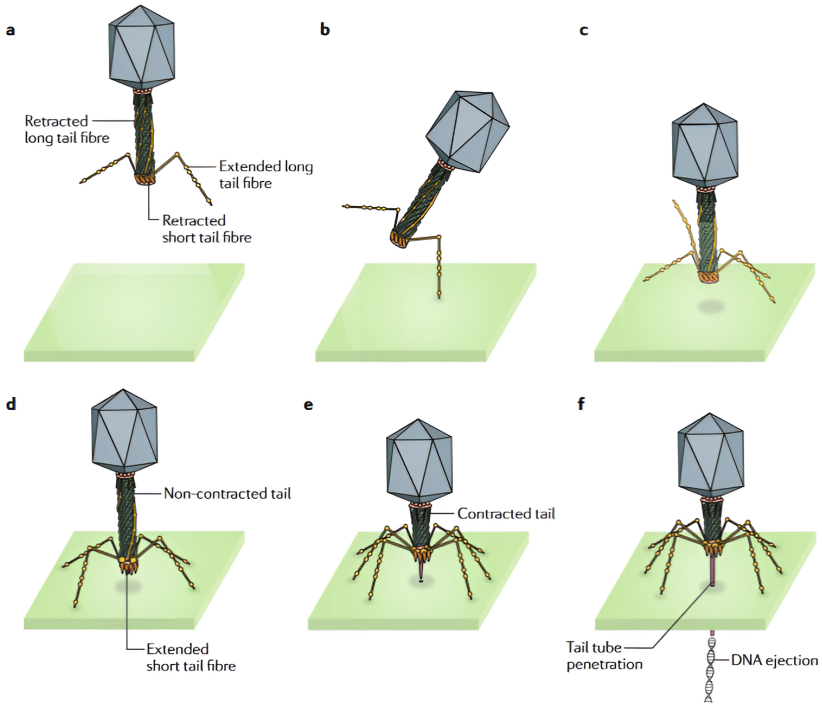


Figure 3: Schematic illustration of myovirus-like phage adsorption to the host. Image adapted from: 10.1038/s41579-018-0070-8 (2).

2.5.1 Receptor Binding Proteins

As aforementioned, the tail component of the *Caudoviricetes* phages is responsible for host recognition, exhibiting RBPs at its tip (43). These interact with specific ligands on the bacteria's surface, enabling adsorption and determining the host range of the phage (58). Adaptation strategies of phages to improve or alter their host range focus on RBPs, including diversification of hypervariable regions or exchange of alleles and domains (52). Therefore, the amino acid and domain analysis can often identify the phage's host range (58). Lastly, RBPs can be classified as tail fibers or tail spikes proteins.

2.5.2 Tail Fiber Proteins

Tail fiber proteins are a diverse group of proteins that play a crucial role in the host recognition process of phages. These proteins are located at the tip of the phages tail and bind to the previously mentioned receptors on the bacterial surface, determinants in its host range of infection (59). Tail fiber proteins are prone to mutations that can improve the infection characteristics of the phage, and upon contact with the host cell surface, these undergo significant conformational changes (60). However, these proteins require the assistance of chaperones, known as tail fiber assembly (Tfa) proteins, to properly fold and assemble.

Reports point out that Tfa proteins may also influence the host range through specific binding to phage cell surface components such as membrane proteins or wall carbohydrates, in addition to their role in tail fiber binding (61). Together with tail fiber proteins, Tfa proteins present a new area of research for phage-host interactions investigation.

Tail fibers can be described as long tail fibers (LTF) or short tail fibers (STF), being both attached to the baseplate when existent (62). When present simultaneously, phage exhibits the most effective adsorption and infection of bacteria. As the name indicates, these differ in length, being in T4 phage described with 145 and 40 nanometers, respectively (63). LTFs make the first contact with the host surface and search for an ideal location for phage adsorption. Most phages display a unique LTF protruding from the end of the tail, being the first to bind to the receptors (64). Hereon, the STFs bind to their receptors to induce irreversible attachment and subsequently orient the genetic material ejection (2). These are better characterized, at a structural and functional level, in phages that infect Gram-negative bacteria. However, less is known about the phage and Gram-positive bacteria interaction (65).

2.5.3 Tail spike Proteins

Tail spike proteins have a complex constitution featuring globular structures and a central domain with an enzymatic activity of binding and degradation of the host receptors. They are located in the baseplates and proceed upon the tail fibers' optimal attachment (66). Hereon, these proteins act upon the TA and LPS receptors on the respective type of bacteria, occasionally portraying hydrolyzing activity (67). However, their affinity to receptor proteins varies from the phage and bacteria in question. Reports suggest diverse specificity of tail spikes to different receptors, such as phage SPP1 tail spike to the membrane protein YueB (68) or tail spike activity specific to O-antigens of distinct Gram-negative bacteria species (58). Thus, this distinctive execution raises an example of phage host range adaptation. In addition to the anchorage function, region conservation analysis implied a position in signal transmission for the DNA ejection on the tail spike encounter with the bacterial surface (69). Regardless, tail spike proteins' role in DNA ejection remains too deeply uncovered.

2.6 Single Nucleotide Polymorphisms

SNPs and other genetic mutations are one of the sources of diversification that leads to the co-evolution between phages and host bacteria. Relatively to therapeutic application, SNPs analysis is systematic to uncover variations that contribute to the most efficient treatment. For this, genomic manipulation

techniques, such as CRISPR-Cas9, urge great interest. Pivoting *Staphylococcus aureus*, RBPs manipulated phages have already shown efficiency in dissemination among several *S. aureus* in vitro cultures and in vivo infections. As anticipated, RBPs revealed wider host range recognition and, thus, improved viral infection (70).

Any surroundings, such as the available hosts or other selective pressures, induce particular phage polymorphisms, as illustrated by the analysis of *Staphylococcus* phages' hypervariable loci (71). For instance, wild-type phage 812 develops several mutants through numerous SNPs. Of these, mutant 812h1 is the most remarkable for its efficiency in MRSA elimination, expounding the broader recognition of different MRSA groups compared to the other mutants. In comparison to wild-type 812, the mutant 812h1 displays 15 significant protein polymorphisms and, during its dissemination within the host bacteria, 255 SNPs (11). Therefore, multiple SNPs were responsible for the mutant efficiency, even though a single-point mutation may exploit new host specificities. Specifically, a simple replacement of a polar positively charged amino acid for a non-charged amino acid unveiled different host recognition on *Pseudomonas aeruginosa* infectious phages (72).

2.7 Bioinformatics Brief History

Bioinformatics is a continuously emerging field that utilizes informatics tools and mathematics to analyze biological data for meaningful information extraction, boosting the efficiency of several coupled domains. Its importance increased with the expansion of biological data, as such massive information requires automatic tools for proteomics or genomic data interpretation. This domain was introduced in the late 50s when the protein insulin was first sequenced (73) through a technique posteriorly implemented in a bioinformatics software pioneer, COMPROTEIN (74). Margaret Dayhoff, associated with Robert S. Ledley, created it in 1962 and later developed the currently known one-letter amino acid code nomenclature. These and several other achievements granted her the title of the bioinformatics field parent (75).

One of the first ML introductions in bioinformatics was the Basic Local Alignment Search Tool (BLAST) creation. Developed in 1990 by Stephen Altschul, it applies heuristic algorithms for sequence alignment and similarity calculation (76). Simultaneously, the Human Genome Project (HGP) debuts the scene with its completion in 2003, relying on the sequencing of DNA fragments and subsequent assembly (77). DNA sequencing methods were unsuitable for large-scale projects within the development of HGP, thence the extended period to accomplish it. However, it unveiled a new era for genome analysis with the introduction of Next-Generation DNA Sequencing (NGS) shortly after (78). The previous methods are time-consuming

and expensive, whereas NGS undertakes these issues on genome sequencing (79). Consequently, massive data from NGS projects arose, urging new DNA analysis approaches.

Given the multifaceted nature of this vast information, bioinformatics tools and ML methods have emerged as effective means to apply them to diverse use cases. Noteworthy, ML is an incredible tool to employ in different specific environments with equal beginning conditions, being useful for research patient-oriented (80) or genome analysis (81).

2.8 Clustering and Alignment

The alignment and clustering are indispensable bioinformatics processes that portray an essential role in biological data analysis and contribute significantly to our understanding of biological systems. These are regularly coupled in bioinformatics tools, although they exercise vital functions separately. Alignment is an effective method for analyzing sequence homology and identifying regions of high conservation (82). Furthermore, its execution is possible with multiple sequences, promoting the comprehension of the evolutionary profile within the related entries. On the other hand, clustering is critical in several tools to reduce the data complexity and ease the comprehensive process in, for instance, ML methods (83).

Sequence alignment is available through multiple bioinformatics tools and is the basis for several renowned database construction (82). The best-known alignment tool is indisputably the previously mentioned BLAST tool (76). Given a DNA or protein sequence, it returns the most similar homologous within the NCBI database with detailed matching values. However, multiple tools display advantages, suiting better on specific cases. A prime example is the Clustal Omega tool, which represents the primary resource for multiple sequence alignment (84). It is not suited for pairwise alignment but handles an enormous number of sequences within a satisfactory time. Besides, it is fundamentally a command-line program and is currently available online through the EMBL-EBI platform (85). Clustal Omega executes by selecting a subset of the sequence to operate as a guide tree. Then, it applies a Hidden Markov Model profile-profile technique that refines the alignment by comparison of sequence pairs.

Relative to clustering, its application occurs regularly in complex bioinformatics tools, mainly in strategies for multiple sequence alignment. However, some tools allow this approach execution individually, such as Cluster Database at High Identity (CD-HIT). CD-HIT is a command-line tool developed in 2001, distinguished by its efficiency and speed in clustering a large dataset (86). Initially, it had a restricted application to proteins (86), but posterior updates have broadened its scope to include nucleotide sequences. The 2006 update introduced the *cd-hit-est* program for DNA and RNA sequence clustering, besides the

tool performance improvement (87). CD-HIT uses an incremental algorithm with user-defined parameters, such as the similarity threshold. It designates the longest sequence as a cluster representative and associates the ones that exceed the similarity threshold with that representative. This procedure repeats until all samples are within a cluster. One of the tool's strengths is the ongoing maintenance from the developers, who continually improve the tool's performance (88).

2.9 Machine Learning

In our current investigation, similar to previous studies that have utilized biological data, we face the challenge of analyzing and extracting valuable information from a large amount of data. ML techniques, particularly those incorporating Deep Learning, effectively address this issue by allowing the identification of patterns from data and solving complex tasks (89). ML has seen widespread adoption in various fields, including medicine (90), drug discovery (91), and microbiology (89). It is commonly applied using four learning methods: supervised, unsupervised, semi-supervised, and reinforcement learning (92). Supervised learning utilizes labeled data, such as positive and negative inputs (93), to make predictive outcomes based on the inputted data. Contrarily, unsupervised learning works with unlabeled data, recognizing patterns deliberated by the algorithm. Semi-supervised learning combines elements of both supervised and unsupervised approaches and is suitable for both labeled and unlabeled datasets. Finally, reinforcement learning involves self-learning through trial and error rather than relying solely on data (92).

The application of ML models in the genomics field has led to the emergence of innovative approaches for discovering drug targets, predicting pandemics, and identifying vaccine candidates (89). Previous studies have demonstrated the potential of these models to uncover phenotypic consequences of genomic mutations (10) and to identify critical protein domains that aid in the classification of pathogens in patient blood samples (94). Additionally, this field has shown promising results in the study of phages (70). As a result, new opportunities for gaining a deeper understanding of various organisms have emerged, and phages are particularly promising for scientific discovery and wide-scale applications.

ML has various possible applications regarding phages. It has shown promise in identifying significant SNPs for host range expansion, specifying interactions, and improving phage efficiency (95). Besides, the field of gene annotation has revolutionized with the development of ML, improving the prediction of ORFs for function-specific proteins (96). These are time-saving processes in biological sequence data classification (97), simplifying researchers' work in genome analysis. They regularly require previous extensive sequence categorization, as various techniques are suitable to manage large data volumes and outcome

pattern prediction. Notably, the most efficient ML models involve supervised learning that, as mentioned before, retains patterns from provided input and output data, which is the focus of this investigation.

2.9.1 Feature Extraction

A fundamental step in any ML model construction involves extracting features from the studying dataset. The designated "features" differ depending on the dataset and problem to overtake but systematically represent data attributes that aid in pattern identification and final prediction by the model (98). In bioinformatics, the input data is regularly genomic or protein content, urging for methods that retrieve sequence-associated features. The 2021 publication from RP Bonidia and others (99) offers a great insight into multiple approaches for feature extraction of genomic sequences for classification purposes. It analyzes mathematical, biological, and hybrid techniques, with the mathematical ones standing out for their superior overall performance. In particular, the combined k-mer and entropy strategies demonstrate high effectiveness, albeit at the expense of increased computational demands.

The k-mer strategy is a powerful method for feature extraction that considers the overlapping substrings of a sequence with a length of "k". Researchers have widely utilized this technique, dating back to 1995 with the development of the TIGR Assembler tool (100). It played a crucial role in simplifying the assembly of large sequencing projects by determining the most frequent oligonucleotide repetitions. K-mer has a simple procedure for genomic data, demonstrating high efficiency in protein sequence prediction, as related in the HBP_NB model for hormone binding protein identification (101). The performance is contingent upon the "k" value since its optimal can fluctuate depending on the specific use case. However, its application must be cautious, as the computational demands increase exponentially with the "k" value (102). Moreover, models that utilize this approach demonstrate exceptional precision in classification tasks, such as the Kameris tool used for categorizing HIV-1 sequences (103), and in determining phylogenetic proximity, as exemplified by the Skmer tool (104).

Currently, feature extraction of genomic content, such as DNA or amino acid sequences, is optimized by specific bioinformatics tools. They utilize functions to derive numerical variables from the unprocessed data, ergo considering them mathematical tools. Some widely known tools for this purpose comprehend the MathFeature (105) and Biopython (106) packages. MathFeature is highlighted as an exceptionally comprehensive package in this field, providing 20 practical methods from the literature for extracting descriptors. Conversely, Biopython includes the module SeqUtils, which contains functions for feature extraction relative to the sequence content, such as nucleotide base content calculation or sequence molecular weight. Furthermore, recently developed tools show promising results in developing new meth-

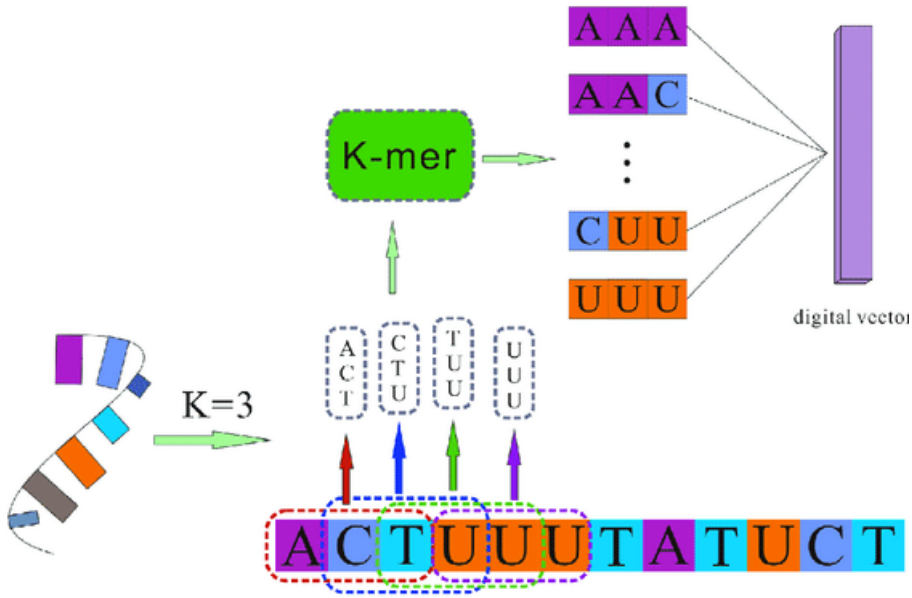


Figure 4: Theoric representation of the k-mer method applied on a DNA sequence. The indicated "k" value is 3, extracting overlapping subsequences with the mentioned size. These comprehend the final vector relative to the sequence. Hereon, different techniques may apply for feature selection and application in ML model training. Image adapted from (3).

ods for an increased number of features extracted. A prime example is the FEGS tool, which transforms an amino acid sequence into a vector comprising 578 numerical descriptors, demonstrating impressive performance (107).

2.9.2 Feature Selection

As suggested by its name, feature selection is a beneficial process that eliminates irrelevant features extracted from the dataset, possibly increasing the model's accuracy. It is fundamental for any use case, as the appropriate feature selection method varies with the dataset and study issue (108). In addition to enhancing accuracy, reducing the amount of input data for model training also decreases the computational and time efforts. A relevant review published in 2019 by K. Tadist (109) and others comprehends the feature selection methods in 6 different groups: filters, wrapper, embedded, hybrid, ensemble, and integrative.

Filter and wrapper methods are the simplest to comprehend, as the first determines the features based on selection criteria and the second through model training and evaluation of feature subsets. Interestingly, ensemble strategies take a similar approach to the wrapper method in evaluating feature subsets. Instead of choosing the best-performing subset, they attribute a performing weight and combine the outputs to

the final prediction. The ML-specialized Python package, scikit-learn (4), provides multiple approaches to apply the mentioned strategies through the *feature_selection* class. Its processes range from simple filter approaches given specific criteria to sequential removal or weighting of independent features.

Conversely, the following methods regularly exhibit a higher level of complexity and urge for tailored strategies relative to the use case. The embedded methods operate on the training process, incorporating a penalty value to reduce feature complexity and enhance efficiency. Relative to integrative techniques, they aim for high feature diversity to present the model with different perceptions of the descriptors variation. Finally, hybrid methods integrate multiple feature selection methods to potentialize this procedure.

2.9.3 Linear Models

As the name suggests, linear models work around the assumption that the proposed data has a linear behavior, adjusting the weights in this accordance. These are the simplest and easiest models to interpret, making them more accessible for regularization and feature analysis. Additionally, linear models have lower computational demands than other ML approaches due to their straightforward relationships. However, it underperforms in more complicated issues, lacking adaptability in non-linear relationships. Its basic premises have some advantages, but its application is limited to real-world problems. Some of the most used linear models comprehend Linear Regression, Logistic Regression, and Ridge Regression.

Linear Regression

Linear Regression is known as the simplest of the ML models, as it describes the samples using a straight line (110). It intends to estimate the relationship between independent variables and the output or dependent variable. The nomenclature categorizes the number of independent variables involved and denominates as either a simple or multiple linear regression accordingly (111). Therefore, this model can assess the mutational factors required to establish a particular phenotype, such as cancer-driven mutations (112).

Logistic Regression

Logistic Regression (LR) differs from the previous in being a classification algorithm that computes the probability of a particular outcome, being more suitable for dichotomous problems. Furthermore, the sigmoid function produces the curve representing the dataset, converting the numeric features into a single probability value (110). Similarly, it may deliver binomial or multinomial outcomes, according to the problem at hand. This model is convenient in biology to evaluate whether a conjunction of mutations results in a particular characteristic or condition (113).

Ridge Regression

Ridge Regression is a method to undertake regression problems with multicollinear parameters (variables highly correlated) that render unfeasible single-parameter estimation. Accordingly, it performs a regularization to prevent model overfitting and provide adequate estimators (114). It outperforms other linear models for multiple SNP analysis while maintaining low computational costs (115).

2.9.4 K-Nearest Neighbors

K-nearest neighbors (KNN) is a non-parametric algorithm that bases its output on the assumption that samples with similar characteristics are likely close to each other or belong to the same classification. It is one of the oldest and simplest classification methods, assigning the class result according to the most common along its "k" nearest neighbors (116). Likewise, a similar strategy is engaged for regression problems, where the average of the "k" nearest neighbors determines the predicted numeric value (116). The "k" value employed defines the model's efficiency, whereas its performance may drastically vary according to the complexity of the problem. Lower "k" values suffer from outliers' impact and capture more local patterns, possibly leading the model to overfit. Conversely, higher values consider more neighbors, performing a more general prediction, which may neglect specific trends and potentially lead to underfitting.

Despite its wide application, classification issues in a biological context are regularly more suitable to undertake with the KNN model. Cancer is an overwhelming matter in current medicine, and this method has shown potential in identifying its type based on the underlying nucleotide variations (117). Early diagnosis significantly enhances the cancer treatment success rate, denoting the importance of accurate classification models. Additionally, it is appropriate for feature selection with positive results when linked to Random Forests (118) or single-handed (119).

2.9.5 Support Vector Machines

Like KNN, the support vector machine model is a robust algorithm applied in classification and regression tasks, with its primary strength lying in classification problems. Its strategy involves finding the optimal hyperplane in a high-dimensional feature space that separates the samples into different classes (120). This approach allows the algorithm to be resilient against outliers, as it disregards peculiar data points when uncovering the best hyperplane. This powerful model has numerous applications in linear, nonlinear, and feature selection tasks (121). In the bioinformatics field, the implementations remain endless, showcasing superior performance compared to any ML model and matching artificial neural networks (122).

Given the performance of support vector machine models, they are often the strategy of choice for bioinformatics tool construction when facing classification tasks. Their accuracy remains very satisfactory even in the category prediction of highly mutable genomic sequences (123). Furthermore, ORF annotation holds significant importance in today's bioinformatics requirements, and support vector classifiers (SVC) provide a solid basis for gene annotation tools of atypical sequences (124). In the context of bacterial treatment, these algorithms improve phage selection by identifying crucial viral elements through prior gene identification (125). As a result, they contribute to enhanced bacterial eradication.

The support vector machine model offers several parameters to tune its performance according to the specific problem domains. Nevertheless, the "kernel" parameter has special consideration as it defines the mathematical function used to map the data points in the high-dimensional space (120). It allows the model to adapt a linear or nonlinear behavior to uncover the optimal hyperplane that separates the data classes, as illustrated in figure 5. In detail, the "kernel" parameter may hold linear, polynomial, gaussian radial basis function (RBF), and sigmoid values.

Linear Kernel

The only kernel to assume a linear behavior, where it computes the product between the two vectors. It is the simplest and most commonly used kernel, well-suited for linear data or relationships. It places low computational demands but has limited applications, as it is unsuitable for nonlinear datasets and noisy data points.

$$K_{(x,y)} = x * y \tag{2.1}$$

Polynomial Kernel

The polynomial kernel is a practical approach to analyze nonlinear data points as commits on a feature space based on the polynomial values of the input samples. Here, we can define a constant positive value, illustrated as *r*, and control the degree of the polynomial, *d*.

$$K_{(x,y)} = (x * y + r)^d \tag{2.2}$$

RBF Kernel

The RBF kernel measurements consist of the similarity between two points according to their distance in the dimensional space. The sole parameter control present is the gamma value (γ), which must be positive, managing the width of the kernel.

$$K_{(x,y)} = exp(-\gamma||x - y||^2) \tag{2.3}$$

Sigmoid Kernel

The sigmoid kernel is the least used of the possible entries and stems from theoretical applications in neural networks. Similar to the previous, it has parameters such as *r* and γ to define the constant value and width of the kernel, respectively.

$$K_{(x,y)} = tanh(\gamma(x * y) + r) \tag{2.4}$$

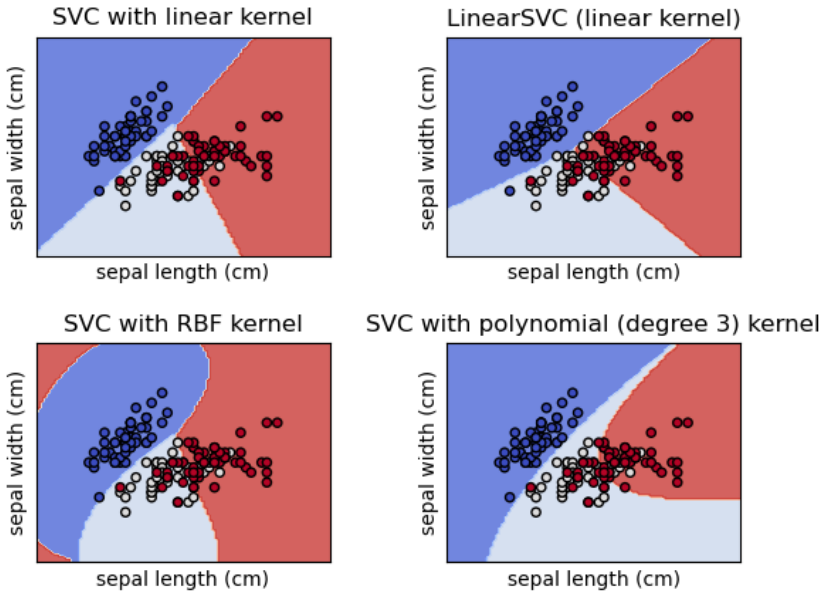


Figure 5: The different optimal hyperplanes encountered in the high-dimensional feature space according to the several kernel values applied. It differentiates the data point classes and attributes the prediction to unseen data of the located region. Image adapted from (4).

2.9.6 Decision Tree

The decision tree (DT) model is a non-parametric algorithm used for classification and regression problems and is extremely popular due to its flexibility and uncomplicated interpretation (93). It consists of a tree-like structure, which simplifies visualization, consisting of ramifications and node decisions. These nodes represent output prediction based on single data features. Firstly implemented in an algorithm in 1980 (126), its flexibility derives from the capability of handling categorical and numerical data, besides generating multi-output predictions. Additionally, its computation demands vary deeply with the depth of the tree, denoting the importance of previous feature selection to diminish extra algorithm efforts. This

method is well-known due to its application in ensemble models, specifically Random Forests, which comprise multiple DTs, predicting the final output based on most individual forecasts.

Relative to phage study, DTs play an essential role in several tools of phage analysis. When interpreting an unseen individual, the characteristics of viral behavior are fundamental to understanding phage utility in bacterial therapy. For instance, this procedure requires phages with a lytic cycle, recognition of the particular infectious bacteria, and efficient host eradication. In this regard, specialized tools based on the Random Forest model for predicting the life cycles of phages have been developed with exceptional efficiency (127). Additionally, it is the foundation of various bioinformatics tools that annotate virion proteins (128) and uncovered phage sequences within metagenomic data (129).

2.9.7 Ensemble Models

Model selection involves using ensemble ML algorithms that integrate multiple models to improve the efficiency of the outcome. The final prediction is generated by combining the most frequent class or computing the average values for classification and regression problems. This ML approach is typical when facing biological data and provides significant results (130). Furthermore, they are less conducive to overfitting and reduce bias, leading to a more robust accuracy. Some frequently used ensemble methods are Random Forests (RF) (131) and Gradient Boosting Machines (132). As previously mentioned, RF consists of a collection of DTs that combine each tree prediction to deliver the final decision. They are frequently necessary in biological data for several issues, such as gene selection (133) and protein interaction identification (134). Alternatively, Gradient Boosting Machines employ a sequence of models to enhance their predictions over time. It stands out for handling missing data and being robust to outliers. In addition, they are commonly appropriate in protein research (135) and mutation detection (136).

Gradient Boosting Machines exhibit a tree-like behavior, similar to RF, replicating the model's sequencing. While it's highly favored, its tree may contain various machine learning models beyond Decision Trees. It employs a boosting approach, meaning that the training of each model occurs based on the prediction error of the previous (137). In this regard, the user may define the desired loss-error function, promoting a high flexibility of the model for any particular task. However, Gradient Boosting Machines are widely unfavoured when compared to RF. They are prone to overfitting when outliers are within the dataset, are less comprehensive than the counterpart, and require more computational and time efforts (138).

2.9.8 Deep Learning

Deep Learning (DL) is a subfield of ML that focuses on algorithms modeled after the brain's structure and function, known as artificial neural networks. Their multiple-layer architecture and associated weights are ideal for handling large and complex datasets. The performance and designation of the known neural networks (NN) vary based on the design of the layers and the activation functions used. Ergo, each NN is suitable according to the issue at hand. For instance, deep convolutional NN (139) and Long Short-Term Memory (140) are suitable models for SNP impact prediction on proteins (141; 142). DL models are adequate for non-linear dataset relationships, as in the case of protein sequence analysis, and may incorporate backpropagation for error reduction. The integration of these features results in these algorithms frequently being the most effective ML models.

DL models undeniably offer the most potential due to their high customizability and intensive training. Nevertheless, these features are also their primary drawback, as they demand extensive computational resources and time, in addition to meticulous network layer construction. Therefore, its application needs careful consideration since several tasks are dispensable of this complexity level. Complex tasks such as speech recognition, image recognition, and natural language processing are current mechanisms that oblige DL usage (143).

Regarding bioinformatics, genomics analysis of complex interactions and mutation prediction has the best development with DL usage (81). The genomics field offers countless applications, with the development of numerous bioinformatics tools based on DL models. For instance, motif finding has value in studying cell molecular interactions, gene expression, primer design, and phylogenetic analysis (144). Thus, recently developed DL tools, such as MotifLen (145), demonstrate remarkable efficiency in optimizing motif detection. Additionally, popular tools like AlphaFold formulated a model capable of predicting the three-dimensional structure of proteins from their amino acid sequences (146).

Nevertheless, our investigation aims to evaluate variants' effects, and detailed tools, such as DeepVariant (147) and NanoCaller (148), perform this task proficiently. Notably, both built convolutional neural networks, being the decisive factor of their efficiency. Currently, the DL models field is highly extended, with some excelling in this field, such as:

Feedforward Neural Networks

Feedforward Neural Networks (FNN) are the simplest and the easiest to comprehend among the DL models. Dissimilar to others, it holds no cycles or loops in its network while maintaining the capacity to perform various complex tasks. The data flows from the input layer through the hidden layers before reaching the

output layer in a unidirectional, forward direction (149). A well-known tool that implements an FNN model is D-GEX, which predicts the expression of target genes (150).

Convolutional Neural Networks

Resembling FNN, the Convolutional Neural Network (CNN) model flows from the input layer to the output layer in a forward direction without any feedback. CNN contains multiple layers fundamental to its performance, such as convolution layers, pooling layers, and fully connected layers (151). The two first compose the hidden layers, whereas the last is the output layer. The convolution layers generate a prediction features map from the input data and are adept at recognizing local patterns. Multiple stacks of these layers prompt the continuous identification of more complex features. However, the size and complexity of the map grow immensely, making pooling layers essential in reducing their dimensionality by applying functions such as maximization filtering or average calculation. Lastly, fully connected layers perform the final task of the network through a weighting matrix and loss functions, having each neuron connected to all the previous and following neurons.

CNNs are the dominant DL model in genomics tasks (81), emphasizing its effectiveness by the noted DeepVariant and NanoCaller tools. Additionally, these may introduce a backpropagation method to reduce the prediction error in each layer. Here, the flow direction is opposite of the normal, propagating from the output to the input layer. At each layer, the relevant weights and biases are fine-tuned based on the gradient of the loss function (152). CNN overview is illustrated in figure 6 for better understanding.

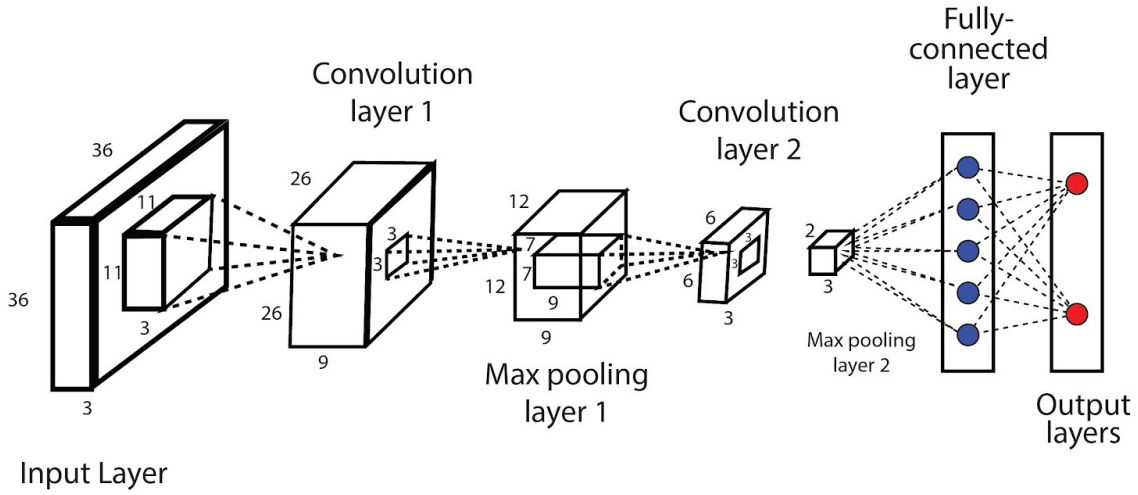


Figure 6: Structure overview of a Convolutional Neural Network, highlighting the role of each hidden layer leading up to the final prediction made by the output or fully connected layer. The hidden layers comprehend the convolutional and pooling layers, which are responsible for extracting feature maps from the data and reducing the dimensions of these, respectively. Image adapted from (5).

Recurrent neural networks

The Recurrent Neural Network (RNN) model stands out for possessing recurrent layers that enable the formation of cycles within the network. These cycles allow a connection between nodes from previous and subsequent steps, capturing patterns spread across the grid. RNNs are useful for sequential tasks that usually unveil patterns across the entirety of the dataset, being useful for speech recognition and natural language processing (NLP) (153). In genomics, RNNs are commonly integrated with other DL models, often in conjunction with CNNs. These comprise tools like ENNGene (154) that facilitate model building for genomics and DeepZ (155) for whole-genome annotation.

Long short-term memory networks

The Long Short-Term Memory (LSTM) model is an improved RNN model developed to overtake the RNNs' inability to capture long-term dependencies (156). It contains gates for the network flow, namely the input, forget, and output gates. They dynamically control the influence of which memory cell should receive and provoke by the input and output gates, respectively. On the other hand, the forget gate manages information retention or forgetfulness of the memory cell. Additionally, Bidirectional Long Short-Term Memory Networks (BLSTM) is a variation of LSTMs that combines a forward and backward flow direction to capture both past and future states (157). These networks hold significance in genomics since long-term sequence dependencies are often disregarded. Consequently, they play a crucial role in developing tools

for predicting protein subcellular localization (157) and sequence function (158), necessitating this broader perspective.

2.10 Substitutions Matrices

Substitution matrices serve as a method to evaluate the impact of a nucleotide or amino acid alteration, assigning a theoretical score supported by numerous criteria and extensive research. These scores are related to the mutation frequency and its subsequent effect (159). Higher scores represent more common alterations with minimal to no impact, while lower scores suggest rare mutations that could lead to a different phenotype. The substitution matrices are primarily suited for amino acid sequences given their complex 20-letter "alphabet" and high mutation influence. Conversely, nucleotide sequence alterations are deceitful due to their simple four-base structure and codon position dependence, inducing multiple synonym mutations. However, these are highly correlated as a single SNP may yield a significant amino acid modification in the protein sequence.

The substitution matrices have several applications in biological research, such as sequence alignment (160), homology detection (161), and database sequence search (162). Sequence alignment is the primary application for these matrices, with an essential role in various renowned tools. One such is MUSCLE, a multiple sequence alignment tool published in 2004 (163) that continues to influence contemporary research. It considers the 240 PAM VTML matrix for score calculation between protein sequences and recommends the usage of medium-size alignments (163). Similarly, the MAFFT tool created in 2002 (164) is a console-based program noted for its accuracy and versatility. It provides extensive customization options, including the choice of substitution matrix, gap penalty settings, and the maximum number of iterations. It has the BLOSUM62 as the default substitution matrix given its overall best performance (165).

As previously mentioned, substitution matrices are incredibly useful in categorizing the impact of a sequence point mutation, serving as an optimal assessment basis of sequence modification consequence. Packpred tool applies this approach very well, predicting point mutation consequences in proteins through a depth-dependent amino acid substitution matrix (166). The creation of this particular matrix has inspiration in the BLOSUM method, scaling its values according to the BLOSUM62 variation (166). Over the years, various matrices emerged, each developed on different criteria and characteristics. Some of these stand out, such as:

PAM matrix

The Point Accepted Mutations (PAM) are one of the first constructed matrices in bioinformatics, developed by Margaret Dayhoff in 1978 (167). Although its development was over four decades ago, its influence persists in protein similarity assessment, being a construction basis for other multiple substitution matrices. The PAM matrix scores derive from closely related protein sequences and quantify the evolutionary distance between two sequences. For instance, PAM100 indicates that in a set of 100 residues, each experienced one substitution, while in the PAM250 each had 2.5 substitutions. Thus, larger PAM values indicate more distantly related sequences.

BLOSUM matrix

The BLOcks SUbstitution Matrix (BLOSUM) is one of the most known substitution matrices oriented for protein sequences, created in 1992 by Steven and Jorja Henikoff (160). They constructed this method on the frequencies of amino acid substitutions obtained from highly conserved protein regions, designated as blocks. BLOSUM matrix possesses several variations stemming from different blocks' identical percentages. Specifically, the considerate BLOSUM62 matrix arises from the amino acid substitution frequencies within blocks that share 62% similarity. Therefore, the greater the value of the BLOSUM variation, the closer the evolutionary distance considered.

VTML matrix

The VTML matrix designation stems from the initials of its creators, Vingron and Mueller (VT), and the approach taken for its construction, maximum likelihood (ML) (168). Drawing initial matrix values from the Dayhoff model, the VTML method refines these values through iterative adjustments based on evolutionary distances and substitution rates. Additionally, it overtakes the PAM matrix's lack of data issue given the time of its development, using a more extensive and diverse dataset (169). The maximum likelihood is an alternative to the previously developed resolvent method and displays slightly better results but has superior computational demands (168). Therefore, it is most suitable for smaller data sets, exhibiting better performance in some analyses than other substitution matrices approaches (170).

PFASUM matrix

The PFASUM matrix arises from the data within the Pfam protein database (171). This database houses several protein families and holds diversified and curated multiple-sequence alignments, enhancing the quality of the matrix construction. Unlike other methods, the PFASUM matrix contemplates the entire

protein sequence and not solely the highly conserved locations (169). These characteristics confer an improved performance for distantly related sequence search, especially with the PFASUM60 variation (171).

GONNET matrix

The GONNET matrix, created in 1992 (172), takes the name of its creator, Gaston H. Gonnet, and stands out for its performance in distantly related sequences alignment (173). It derives from sequence alignments (173), similar to BLOSUM and PAM, and computes the matrix values by a complex logarithmic equation, considering amino acid alignment by chance. Then, normalize the final scores to 250 PAM units, conferring high affinity to distant alignments.

NUC44 matrix

The NUC44 matrix has a distinctive application to nucleotide sequences and is the default scoring foundation within NCBI's BLAST program (76). Additionally, it contemplates nucleotide ambiguous codes, known as IUPAC, that represent the uncertainty of the base position and give a designation to the known details, such as purines or pyrimidines.

2.11 Galaxy

Galaxy is an open-source platform that provides researchers and bioinformatics access to various computational biology tools essential for investigation (174). It provides a simple interface to input required files and parameters, enhancing the tool's processing through its potent environment. Besides bioinformatics tools, it comprehends datasets, workflows, and data analysis, benefiting the initial research process. Furthermore, it is an accessible method for bioinformaticians to share their work through a public platform, such as the ToolShed framework (175). It represents the initial step to include the developed tool in the Galaxy platform, as Galaxy administrators may access and install the numerous available instruments in their instances.

This tool emphasizes transparency in computational biology, possessing an active community for platform development to accompany the latest bioinformatics advancements. Its improvement stems from the continuous upload and feedback provided by the users.

Part II

Work Plan

Chapter 3

Methodology

3.1 Dataset construction

To achieve the research objectives, we need a comprehensive and extensive dataset of protein sequences. Our focus is on phages that target a specific bacterial genus, *Staphylococcus*, and the crucial proteins involved in its binding and recognition. As previously concluded, the RBPs are the fundamental focus of our sequence gathering. This assembly is designed to be automated and supported by algorithms and bioinformatics tools, including the Biopython library (106). Furthermore, our goal is to access multiple databases, including NCBI (176) and UniProt (177), to obtain the most extensive dataset of sequences possible. Using this approach, we aim to collect several sequences of RBPs with filtering and removal of unintended ones. The main algorithms developed for this section are described in the *data_mining.py* and *conserved_domains.py* files, where the sequence gathering comprises the positive samples sub-dataset. On the other hand, the negative samples sub-dataset collection, fundamental for the training of classification models, is performed in the *negative_samples.py* file.

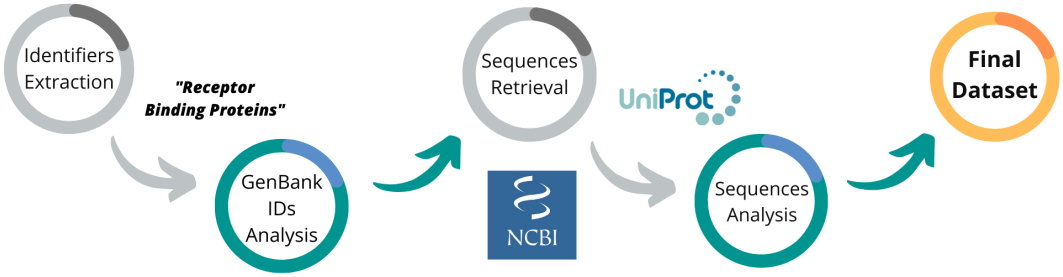


Figure 7: Image describing the general pipeline of the dataset construction. We begin with GenBank accession ID identification corresponding to the "Receptor Binding Proteins" or associated keywords. This extraction occurs in the most known protein databases, such as NCBI and UniProt. We retrieve the corresponding protein and nucleotide sequences from the previous accession IDs. Hereafter, several sequence analysis steps occur to make the final dataset as robust and complete as possible.

3.1.1 Positive Samples Extraction

We employed four approaches to extract positive sequence samples, utilizing the Entrez module from the Biopython library. Their development plans to expand the dataset significantly by extensive mining of public databases. These comprehend the following:

(1) Entrez Phage Genome Search

Firstly, a direct automated method with *Entrez.esearch()* function allows the acquirement of accession identifiers for genome sequences associated with the search term "staphylococcus phage". Despite the constraint of retrieving 10,000 records per query, parameters like *retstart* and other optional settings facilitate the expansion of the search list through iterative queries. It yielded a focused assessment of 1,127 analyzed genomes, encompassing approximately 73,112 protein entries.

(2) UniProt Mining

Subsequently, we extracted through the UniProt web platform 396 protein entries, constrained by the term searches referring to RBPs and organisms belonging to the *Staphylococcus* genus. Due to UniProt API's lack of resources and efficiency, the JSON details file from the online portal presented all the essential resources for each protein entry, demanding extensive algorithm development for the descriptions and sequence assessment.

(3) Conserved Domains and Protein Families

The subsequent strategies serve to address any potential gaps in sequence retrieval. Initially, our approach involved identifying the most frequently occurring conserved domains (CD) and protein families (Pfams) within our current dataset using the NCBI's Conserved Domain Database (CDD) (178). Through the CD-search tool (179), we retrieved 377 records from our dataset analysis, providing details such as score values, hit types, and entry descriptions. Despite the minute description, manual curation aided with algorithm filtering of the hit data was essential to eliminate undesirable records. This process led us to identify the cd13402, Pfam10651, and Pfam14928 identifiers as the most promising candidates. Hence, we obtained the incorporated accession identifiers from each candidate, documented in table 1. This prompted the development of the *conserved_domains.py* file for data extraction.

Table 1: For each CD and Pfam candidate considered for RBP sequence mining, we display its comprehensive information, including the analyzed accession IDs and corresponding candidate descriptions. These accession ID values correspond to the date of 25/06/2023.

Candidate	Accession IDs	Description
cd13402	329	Lytic transglycosylase-like domain of tail fiber-like proteins and similar domains
Pfam10651	3052	Purported to be a putative long-tail fiber protein domain.
Pfam14928	1	Short tail fiber protein receptor-binding domain

(4) NCBI Query Advanced Search

Lastly, we resorted to an advanced constrained query search on NCBI, resulting in 73,758 protein accession identifiers. Unlike the first method, we began with the protein database due to the significant scarcity of annotated nucleotide sequences. Moreover, these entries stem from a restrictive query focused on RBPs nomination and *Staphylococcus* genus, as outlined in table 2. Despite the enhanced potential for extraction, akin to the second method, several annotated proteins lack the necessary resources to obtain the corresponding DNA sequences, which are vital for constructing the dataset.

We summarize the detailed approaches in table 2, denoting the relevant parameters associated, such as the number of retrieved accession IDs, resources applied, and pertinent observations.

Table 2: Methods employed for positive sample sequences extraction, serving to maximize the sub-dataset size. We highlight the number of retrieved accession IDs, resources applied, and pertinent observations. The adopted strategies are (1) Entrez search mechanism using the nucleotide database to extract genome accession identifiers associated with the term "staphylococcus phage"; (2) Restrictive query search in the UniProt web platform and output JSON algorithm analysis; (3) Search sequences included in the most recurrent CDs and Pfams within the dataset and associated to RBPs; (4) Query advanced search in NCBI protein database, restraining the records to *Staphylococcus* genus and protein description including one of the RBPs keywords.

Method	**Accession IDs**	**Resources**	**Observations**
1	73,112	term="staphylococcus phage"	RBP keywords search with Entrez package search mechanism
2	396	*(organism_name:staphylococcus) AND (fiber OR spike OR "receptor binding")*	Uniprot query search and JSON file analysis
3	3,382	cd13402 Pfam10651 Pfam14928	CD and PFAMs sequence extraction and manual curation
4	73,758	*(txid1279[organism:exp]) AND (tail fiber OR tail spike OR receptor binding)*	NCBI's Protein database query construction
Total	150,648	-	-

3.1.2 Negative Samples Extraction

Negative samples are composed of sequences that do not encode RBPs. They are essential to classification models since they provide data examples of the negative class. Negative samples easily outnumber positive class occurrences, raising the demand for a balanced constructed dataset to avoid biased predictions. For this purpose, we develop two datasets differing in the number of randomly selected negative samples. The first dataset comprises 2,000 negative samples, while the second encompasses 4,000. This strategy evaluates the model's performance when the negative class distribution changes. The sequences were all

extracted from the *Staphylococcus* phages utilized in the positive samples sub-dataset. For a sequence to be considered a negative sample, it must fulfill specific parameters, such as:

1. It must be a Coding Sequence.
2. The sequence size must fall within the range of the minimum and maximum size of the positive samples.
3. Must not be a truncation of another sequence.
4. Does not encode an RBP protein.

3.1.3 Sequences Analysis

With the extraction from different sources, verification, and filtering becomes essential to remove truncation, repetitions, and sequences from unwanted genera. *Data_analysis.py* file construction serves this purpose. In addition to truncation and repetitions removal, sequences with 3'->5' orientation are identified, thereby obtaining their reverse complement.

The first analytic step implemented occurs during the creation of the dataset. Each candidate protein undergoes validation to ascertain whether it constitutes a duplication within the present sequence list or is encompassed within one of them. Nonetheless, we reapply this approach post-extraction to mitigate any potential sequence redundancy.

Initially, our data mining algorithm prioritizes the extraction of protein sequences, mainly because some nucleotide sequences lack proper annotation. Whenever available, we also store the corresponding DNA sequences of the proteins. More than 1,000 sequences missed the DNA annotation. Therefore, two different strategies for gene annotation recovery took place.

The first assumes that an unannotated sequence shares homology with a well-described sequence in a different organism. We then investigated the accession IDs of various organisms within the current dataset and successfully retrieved the DNA sequence of several missing genes. Despite this approach, there are instances where accessions with annotated genes may lack the complete genome sequence, making the extraction of the DNA correspondent unfeasible. In such cases, we resort to the second strategy. This strategy involves manipulating the accession ID by incrementing or removing the prefix "NZ_" depending on the current one. The "NZ_" prefix signifies the updated accession (176), and by modifying it, we aim to access the new annotation or, in the case of removing the prefix, the old annotation that may contain the complete genome sequence.

Thus, the implemented algorithms focus on the recovery of positive sample DNA sequences. Conversely, the negative samples did not require any analysis or recuperation.

3.2 Receptor Binding Proteins Prediction

The first step while analyzing a phage genome is the identification of relevant genes to the case study. In this case, genes that recognize the hosts (RBPs) are the search focus for the tool development. For this purpose, we require an ML model creation to enhance the tool performance. Similar mechanisms for gene search in a complete phage genome follow a classification model to identify whether a DNA sequence incorporates one class or another (96). Additionally, these models are useful for prophage uncovering within bacterial genomes, following a classification pattern with overlapping sequences. (180).

RBPs sequences vary immensely according to the strain, species, and genus that the phage recognizes (52). Their initial recognition in a not annotated genome can be challenging, suffering from a lack of online tools that overtake this issue. Therefore, our classification tool is specifically trained to identify these sequences in *Staphylococcus* phages, not ensuring the recognition of different genera due to the previously mentioned. The *rbp_identification_ml.ipynb* notebook describes the general model development for further details.

3.2.1 Feature Extraction

As previously noted, "K-mer counting" is one of the most well-known methods for feature extraction in DNA sequences. Its importance is evident in ML model development for genomic element classification, where specific tools like SPRISS (181) were conceived to improve efficiency and reduce processing time, given its high computational demands. Additionally, "k-mer counting" reveals superior effectiveness in model testing and validation compared to other feature extraction techniques (182). Therefore, it is responsible for the usefulness and superiority of some genomic classification tools (183).

"K-mers" are overlapping subsequences of a fixed length extracted from the original DNA sequence. The choice of the length value significantly affects the model's accuracy, and in this work, we assess a range of values spanning from 3 to 6. Unfortunately, higher values exploration was conditioned by the limited computational resources available. As the k-mer size increases, the quantity of subsequences expands exponentially, leading to heightened demands on processing capabilities and memory (184). From here, the Scikit-learn python package (4) contains diverse classes that greatly facilitate feature extraction and model training. Specifically, CountVectorizer class performs the "counting" phase of the method,

converting the collection of k-mers into a matrix of token counts.

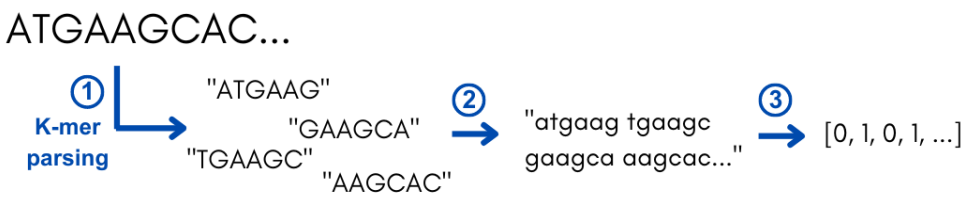


Figure 8: Schematic representation of the feature extraction process. (1) K-mer parsing of the DNA sequence, resulting in hexamers subsequences; (2) Aggregation of all hexamers into a single string; (3) Obtaining the matrix token count by measuring k-mer frequencies with the aid of the CountVectorizer class.

3.2.2 Model Development

For the RBPs classification model, we trained several supervised models on the previously constructed datasets. The selected models for fine-tuning include LR, KNN, DT, SVC, and RF Classifier. The constructed SupervisedModelOptimization class permits an efficient model fitting, testing, and relevant score metrics display. We divided the respective dataset with a 30% proportion for testing purposes, leaving the rest for model training. Furthermore, each model has predefined hyperparameters that can significantly impact its accuracy, as specified in table 3.

For model tuning, we employed the sci-kit's RandomizedSearchCV method, according to the "f1" estimator score metric. We kept the maximum hyperparameter iterations at the default value of 10, set cross-fold validation to 5, and defined the random state argument as 42.

Additionally, we constructed DL models to assess their future potential in predicting RBP sequences. This process occurs after the results validation processes, as we did not consider these models for the subsequent best ML model selection. They mainly represent future efforts for improved performance on RBP sequence prediction.

Therefore, we maintained the NN construction primarily with low layer complexity since DL models require high computational efforts. The employed DL models include FNN, CNN, and RNN, developed with sklearn and TensorFlow (185) libraries. We preserved the dataset division at 30%, tuning multiple parameters suitable for the objective, as noted in table 3.

We took testing procedures to evaluate the performance of BLSTM and LSTM models. However, their computational requirements exceed our production environment's capacity. Consequently, we will not incorporate these NNs into our testing and subsequent analysis.

The FNN method construction occurs with the MLPClassifier function available by the sklearn package. We maintained all the default parameters suggested, tuning the most relevant. On the other hand, the remaining NNs followed the same structure of encompassing their unique hidden layers complemented with one Dense layer. Nonetheless, several parameters retained a default setting, including an epoch size of 10, the "adam" optimizer, the "categorical_crossentropy" loss function, and the "accuracy" metric. Ideally, the default metric should follow the F1-score as the previous models, but we preferred to maintain the recommended metric by Keras documentation (186). In addition, the standard Dense layer found in all NNs, except for FNN, maintained a kernel value of 3, corresponding to the total number of potential classifications, and utilized the "softmax" activation function.

Table 3: Each ML technique tested in the model development, denoting the default and tuning required hyperparameters. The table contains Supervised ML models, including several NN strategies. Notably, the CNN model has the highest hyperparameter variation as the model is theoretically the most suitable DL model for our paper's objective.

Model	**Default Parameters**	**HyperParameters Tuning**
LR	{ solver="lbfgs", max_iter=400 }	-
KNN	-	{ n_neighbors: [5, 10, 20], weights: ["uniform", "distance"], p: [1, 2] }
DT	-	-
SVC	-	{ kernel: ["linear", "poly", "rbf", "sigmoid"] }
FNN	{ max_iter=1000, random_state=42 }	{ hidden_layer_sizes: [100, 200, 300] }
CNN	-	{ hidden_layers: [32, 64], kernel_size: [2, 3, 5], activation_function: ["relu", "linear"], batch_size: [32, 64] }
RNN	{ activation_function="relu", batch_size=32 }	{ hidden_layers: [32, 64] }

3.2.3 Model Validation

Despite the model metric scores, a verification in random sequences is the proper validation approach to assess its accuracy. To fulfill our objective, we randomly selected 100 genomes from NCBI that corresponded to *Staphylococcus* phages. We achieve this by utilizing the Entrez library to search for the term

"Staphylococcus phage" and retrieve 2,000 associated genome sequences. Furthermore, we use the Pyrodigal package (187) to predict the ORFs within each genome. It is an open Python module bonded to Prodigal (188) that offers an intuitive approach to obtaining all the ORF locations and strands, making gene sequence extraction simple. Due to the Pyrodigal package restriction of a minimum 20kbs genome length for proper ORF extraction training, urges the filtering requirement of all the entries below the length threshold.

Hereon, we applied the chosen model on each ORF and corroborated all the gene annotations in the GenBank accession with the model-predicted RBPs of the phage genome. Subsequently, we evaluated all the mislabeled sequences and the reasons behind them. Finally, we perform manual curation in inconclusive labeling with the Protein BLAST instrument available through NCBI (76). The *predict_phage_rbps.ipynb* notebook contains the validation described steps.

3.2.4 Model Development Overview

The model overview resumes the general strategy employed to achieve the best ML model for RBP prediction, as described in figure 9. Notably, we execute several steps to obtain a dataset of RBP sequences with a considerable size without any truncation or duplication. We used multiple approaches for the highest possible number of positive samples, enriching the dataset with an even greater quantity of negative samples. Posteriorly, k-mer counting is the chosen feature extraction method, denoting high performance when applied in classification issues of nucleotide sequences (182). It demonstrates varied performance with the "k" value chosen, urging for parameter analysis. Hereon, multiple supervised ML models are trained and evaluated, with hyperparameter tuning, to assess the best model for our objective.

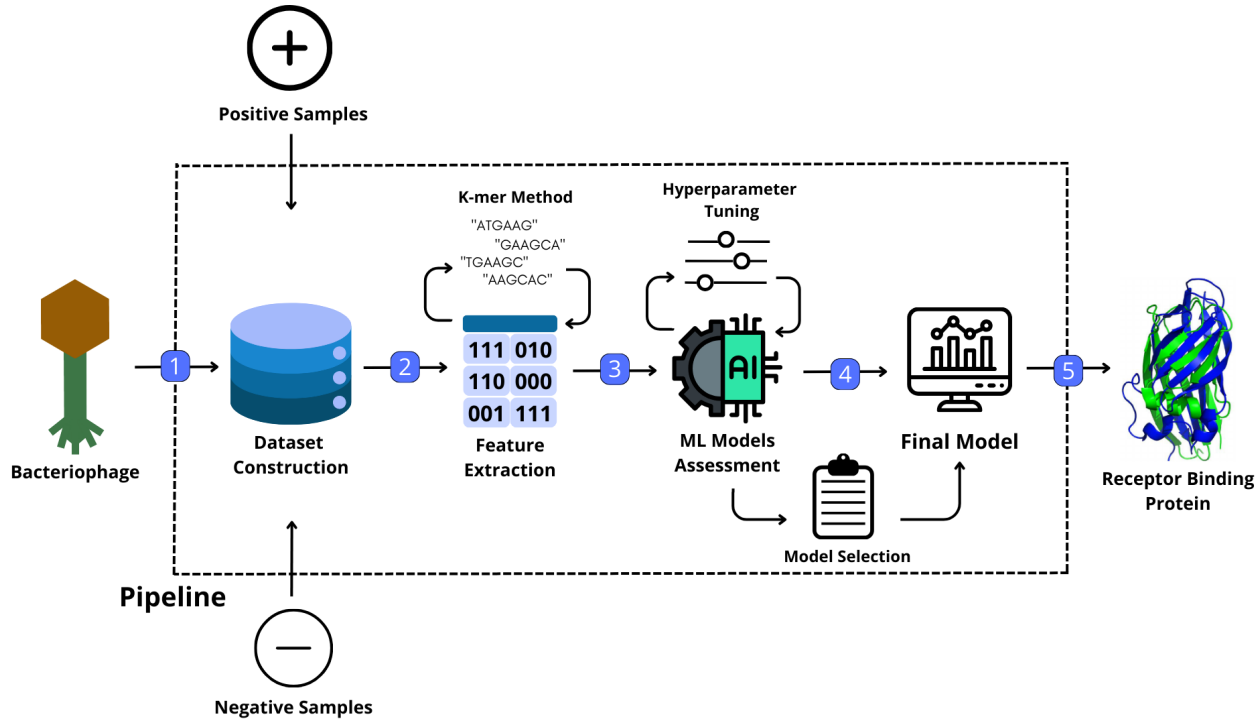


Figure 9: Model development overview resuming the main steps performed to achieve the optimal choice for RBP prediction. 1) Comprehend the procedures taken to collect all the RBP sequences existent in *Staphylococcus* phages, executing distinct approaches for positive and negative sample extraction; 2) Method for feature extraction where we employ the k-mer counting strategy; 3) Possible models assessment through training and testing of multiple supervised ML approaches, complemented with hyperparameter tuning; 4) Final consideration of the testing results to choose the optimal model for our objective; 5) Best model employed to predict RBP sequences.

3.3 Single Nucleotide Polymorphisms Annotation

SNP proper annotation is exceptionally challenging given the absence of annotated mutations specific to our use case in the literature or comprehensive datasets. Therefore, the lack of labeled data is a big hurdle for ML model training. Herewith, we remain with the alternative of analyzing each polymorphism by generic substitution impact. The best-known approaches for gene substitution influence land on developed substitution matrices, such as BLOSUM and PAM. These assess the significance of amino acid alterations by assigning a score according to the resulting and original amino acid (159).

With this, by assembling multiple well-known and established substitution matrices, we can assign for each SNP a theoretical impact. These combined scores arise from diverse characteristics examination within a sequence position alteration, allowing a speculative assessment of its consequence.

3.3.1 Dataset Clustering

Dataset clustering is essential to group, based on homology, the thousands of RBPs composing our positive samples. It promotes a straightforward observation of multiple alterations along the sequences. Hereon, we intend with this arrangement to ease the analysis of unknown sequences by integration within the most similar cluster.

The Cluster Database at High Identity with Tolerance (CD-HIT) is a commonly used tool for efficient clustering out of a sequence dataset. It allows the process to be performed according to a customizable alignment threshold, providing control over cluster specificity. For our research intentions, we maintained the boundary at 70% to gather the sequences belonging to the same homologous group, avoiding overly specific clusters from a high threshold and ambiguous clusters from a lower one. For the rest, we kept the default CD-HIT parameter values for the clustering processing.

3.3.2 Cluster Alignment

Proceeding the clustering phase, for the analysis of an unannotated sequence, we require the identification of the best cluster for its incorporation with a minimum of 4 in-depth. This threshold is fundamental to acquiring a minimum efficiency in SNP identification within the multiple sequence alignment. Lower values contribute to extracting several unwanted mutations, slowing the analysis process and filling the output with non-significant polymorphisms. The SNP identification is possible through the pairwise2 module featured in the biopython library, which comprehends the most known alignment procedures, namely the global alignment technique. This algorithm calculates the score between the input and each cluster representative, determining the most suitable set for the given sequence, as described in figure 10. The values applied for the alignment are 1 for identical characters, -1 for non-identical, and -1 for each gap.

Afterward, we require an approach to pinpoint the SNP positions within the sequence. To this end, Clustal Omega is an exceptional tool for multiple sequence alignment with a straightforward guideline (84). It fulfills our purpose of mutation identification since the conserved and polymorphic locations from the cluster become well-described. Currently, Clustal Omega only provides its implementation via command line code by accessing its source code. Thus, we apply it through subprocess execution that allows a command line snippet to be conducted from Python. As a result, it produces an alignment file easily interpreted for SNP examination.

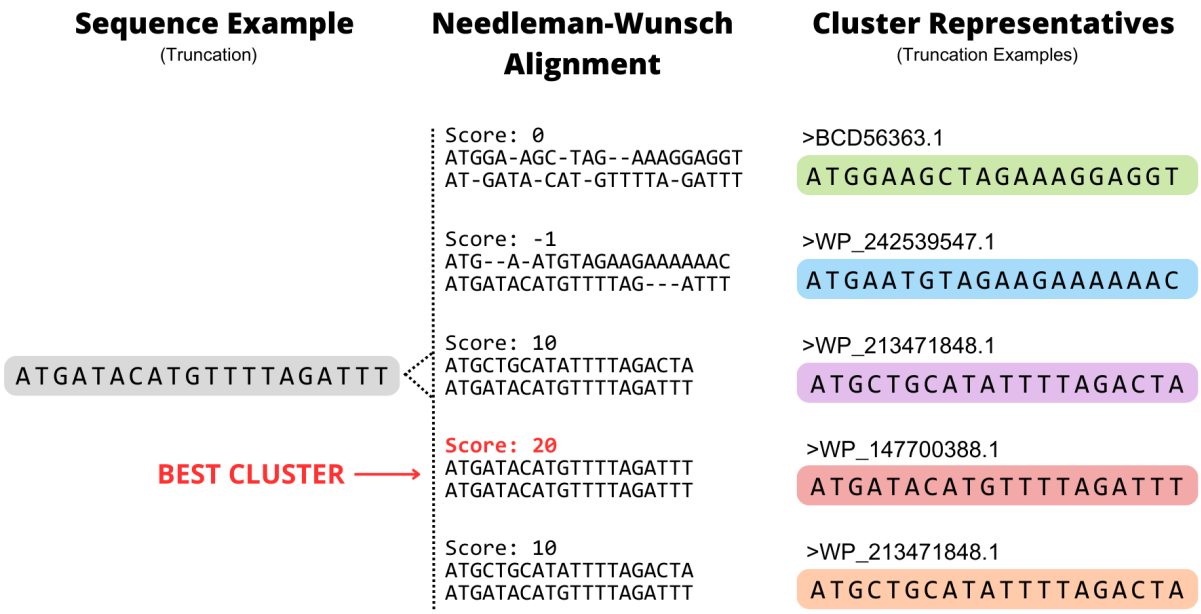


Figure 10: Overview of the cluster selection method implemented. We compare the inputted sequence by the user with all the cluster representatives previously defined through the CD-HIT tool. We execute this comparison through Needleman-Wunsch or global alignment, which provides an alignment score proportional to the sequence similarity. Thus, the highest score represents the best cluster to integrate the sequence.

3.3.3 Polymorphisms Extraction and Scoring

The SNP extraction and scoring phase comprises four fundamental stages: identification of candidate polymorphisms, calculation of each matrix score, computation of the final score, and threshold filtering.

Firstly, the selection of the candidate polymorphisms has inspiration from the work developed in the NanoCaller tool (148). Considering an alignment position, we evaluate the alternative allele frequency relative to the allele from the input sequence. Therefore, for each genomic site, we apply the following equation, supposing *B* is the reference allele, *b* is the genomic site, and *d* is the site read depth:

$$Frequency = \frac{max\{allele\ B\ at\ site\ b\}}{d} \tag{3.1}$$

In the NanoCaller work, the threshold value for the alternative allele frequency is 15%. However, we established it at 40% to decrease the occurrence of false positives, noting that the user may define its desired threshold through the available parameters.

Subsequently, each SNP candidate undergoes analysis of the SnpMatricesScores developed class. This step involves extracting the original and modified amino acids from the genomic polymorphism useful

for specific calculations. Herewith, we evaluate the alteration through several substitution matrices, such as BLOSUM, PAM, PFASUM, VTML, CONNET, and NUC44. The scores for each matrix are computed based on the nucleotide or amino acid mutation, as required, and then aggregated. BLOSUM, PAM, PFASUM, and VTML methods include multiple matrices, varying with the sequences' proximity (159). Therefore, we perform a weighted computation of some matrix alternative values, favoring the ones applied in highly related sequences, as described in tables 4 and 5. Relative to BLOSUM and PAM matrices, we chose five variants equally distributed to assess the different impacts of the amino acid alteration. Additionally, we implemented the three PFASUM best-performing matrices (171) and three VTML matrices distributed with good results (171) and most suitable for close related sequences (189). Given the study case, we intend to investigate a specific protein family, RBPs, within a distinct bacterium genus, *Staphylococcus* spp., so we expect an elevated genomic relationship.

Table 4: The weights employed in each variation of BLOSUM and PAM matrices, with greater weights assigned to the ones representing higher sequence relationships. BLOSUM proximity relevance is proportional to its representative number, whereas PAM has the opposite behavior. Therefore, the most significant matrices are BLOSUM90 and PAM100.

Matrices	**BLOSUM45 & PAM250**	**BLOSUM50 & PAM200**	**BLOSUM62 & PAM160**	**BLOSUM80 & PAM120**	**BLOSUM90 & PAM100**
Weights	5	7.5	12.5	25	50

Table 5: The weights implemented for the PFASUM and VTML matrix variations. Like BLOSUM and PAM, the matrices for closer homology have higher weights to exert more influence in the final matrix value calculation. Thus, PFASUM60 and VTML20 are the most significant matrices.

Matrices	**PFASUM31 & VTML160**	**PFASUM32 & VTML80**	**PFASUM60 & VTML20**
Weights	20	30	50

Finally, we conduct a similar approach to the multiple matrices score estimation processes for the final value assessment. Here, each matrix has a defined weight to bias the result for the ones with higher closeness. Specifically, the group of various matrices has the highest significance of 20% due

to the prior estimation tendency targeted to close homology. On the contrary, the GONNET matrix has the lowest weight of 5% since it normalizes upon the PAM250 matrix, which serves for a more divergent set of sequences (190). The NUC44 has the unique feature of using nucleotide alteration for the score calculation and an importance ratio of 15%, lower than the others due to its decreased relevance relative to amino acid analysis.

After obtaining the set of SNP scores, we exclude the values equal to or greater than 1 to filter the less relevant alterations and lower or equal to -4 to remove too impactful modifications in the resulting protein function. Typically, substitution matrices interpret a score of 0 as found by chance, with rareness increasing as the value decreases (191). Thus, a less frequent mutation indicates a higher impact on the resulting phenotype.

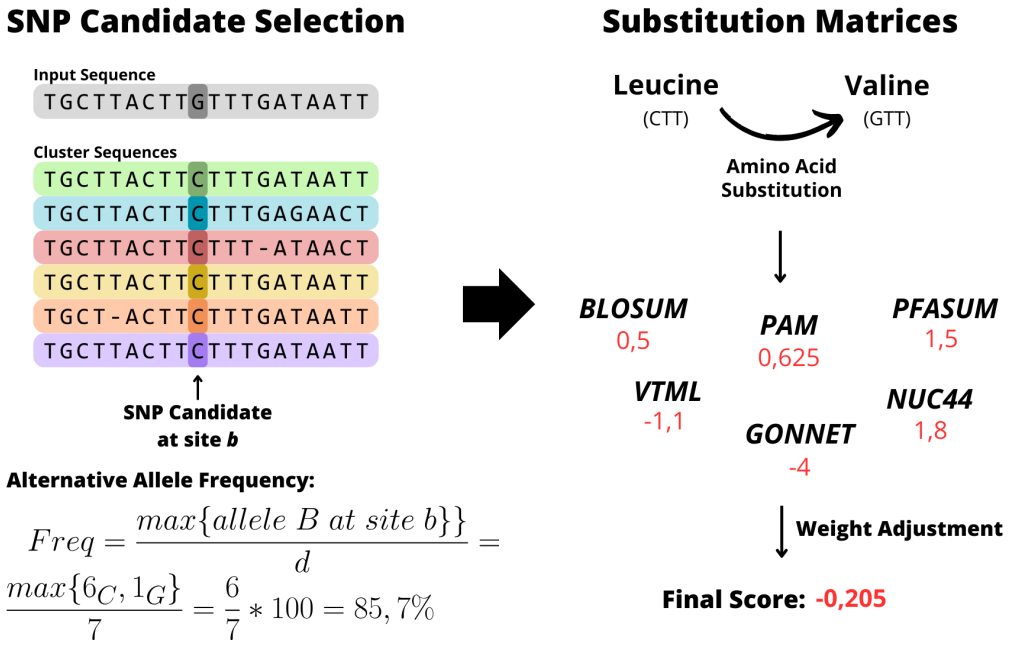
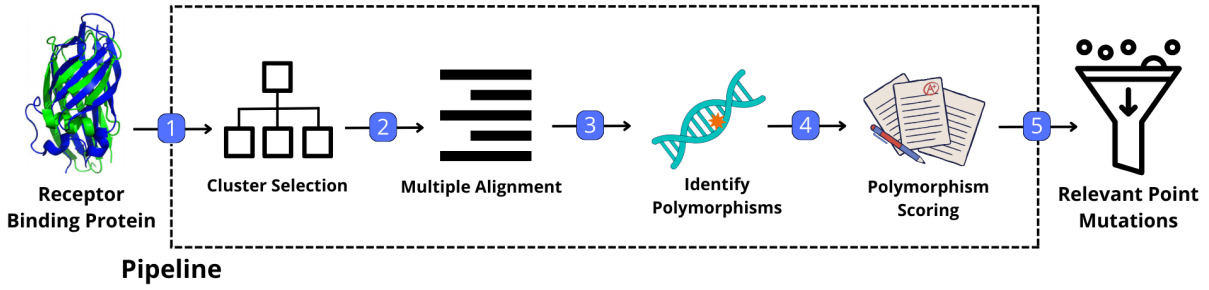


Figure 11: Overview of the SNP scoring strategy implemented. Firstly, the cluster alignment is analyzed to extract all the possible candidate SNPs. The Alternative Allele Frequency equation indicates whether a site *b* is a candidate by the frequency value exceeding the defined threshold. Hereon, we uncover the altered amino acid resulting from the mutation and cross-reference it among all the substitution matrices considered in the paper. Each matrix has a weight value according to its relevance, and its appliance with the average calculation results in the final score that reveals the theoretical SNP impact value.

3.3.4 Polymorphisms Analysis Overview

The SNP analysis overview, demonstrated in figure 12, shows the procedures developed to extract all the relevant SNPs within an RBP sequence. The study starts with the selection of the most appropriate cluster to assemble. We select it by the highest global alignment value obtained from the RBP sequence and cluster representatives' alignment. Then, we apply the Clustal Omega tool to perform a multiple alignment on the cluster and RBP sequences. Hereon, SNP identification becomes more straightforward, allowing the extraction of all the candidate polymorphisms according to the alternative allele frequency function value. Subsequently, we analyze each SNP with multiple substitution matrices that act upon the amino acid or nucleotide alteration. The final impact score derives from the conjunction of all the substitution matrix values, each adjusted to its corresponding weight. Lastly, we filter the non-relevant SNPs by assigning a value range threshold of -1 to 4.



wo

Figure 12: Overview of the executed steps to analyze the SNPs within the RBPs. 1) Retrieval of the best cluster to integrate the RBP sequence; 2) Multiple alignment process of the several sequences within the cluster and the initial sequence; 3) Mutation identification through the analysis of the produced alignment and verification if it represents a candidate SNP; 4) SNP scoring through the value consideration of multiple substitution matrices; 5) Relevant SNP filtering according to the defined impact value thresholds.

3.4 PhageSNP Tool

3.4.1 Usage Guidelines

PhageSNP tool is a Command Line execution program with a simple application for the user. It takes several input parameters to achieve the desirable output, all described under the "–help" command. The accepted parameters are the following:

1. –seq: Permits the user to introduce directly a sequence string to be analyzed by the PhageSNP tool.

2. –file: The tool development can predict RBPs within an unannotated genome. Therefore, since a genome is of great extension to introduce in a command line prompt, this parameter induces the extraction of the sequence written in a file.

3. –g: The RBPs uncovering is worthwhile solely when analyzing a genome. This parameter defaults to False to avoid searching for ORFs in unwanted cases. However, the user may introduce "–g True" to acquire this feature.

4. –t: This parameter defines the threshold value for each candidate polymorphism to undergo examination. The permitted value ranges between 40 and 60, so the user can increase the specificity of the uncovered SNPs, defaulting to 40 if not specified. Thus, higher values will reduce the number of returned SNPs but increase the efficiency, whereas lower values will have the opposite effect.

Considering this, PhageSNP provides various options to match the user's preference. Each tool invocation generates an output file in the user's "Downloads" directory. This file contains the predicted protein function of the multiple or single sequence inputted and the uncovered SNPs within, indicating their impact score.

3.4.2 Model Application

The tool's main objective centers on the RBP gene prediction in an unseen and unannotated genome. To accomplish this, we require a method to identify all the ORFs beforehand. These ORFs will then serve for sequence feature extraction and classification in the previously trained model. Therefore, the previously mentioned Pyrodigal package urges as extremely useful for effective ORF prediction.

Finally, the feature extraction procedure of the ORFs is similar to the one described in the previous sections, only differing in the usage of the CountVectorizer object created by the original dataset annotation. Then, we perform a straightforward prediction with the trained model and filter the negative classification to identify the RBPs in the genome. This section's algorithm details are described in the *predict_rbp_sequences.py* file.

3.4.3 Tool Overview

The tool overview, illustrated in figure 13, details the engaged stages to uncover all the RBP sequences of the user-inputted data and SNPs, indicating their theoretical impact score. The process begins with the prediction of RBP sequences from the data provided by the user, which could be a nucleotide sequence or an entire genome. The user is required to specify this through the "-g" parameter. Hereon, the Pyrodigal package identifies the existent ORFs, which are then classified and filtered by our top-performing model. Furthermore, we extract each SNP with an alternative allele frequency above the defined threshold. We accomplish it using a combination of clustering and alignment methods provided by the respective CD-HIT and Clustal Omega packages. With this, we evaluate the isolated SNPs by combining the theoretical scores of multiple substitution matrices weighted according to their importance. Finally, we generate an output file to provide the user with each RBP predicted sequence and respective evaluated SNPs.

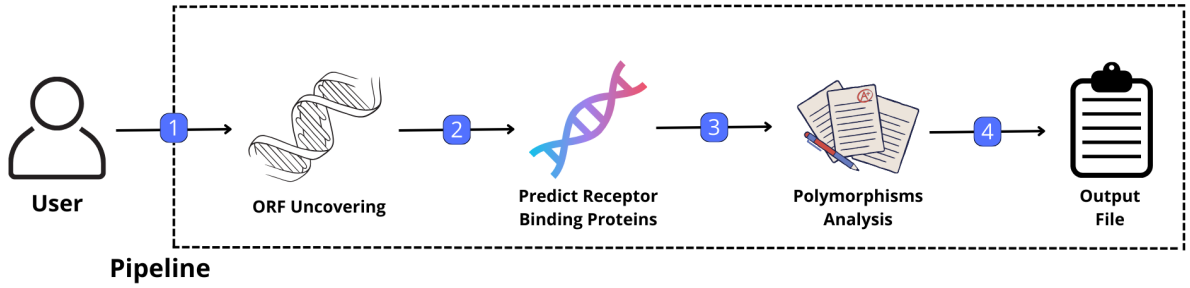
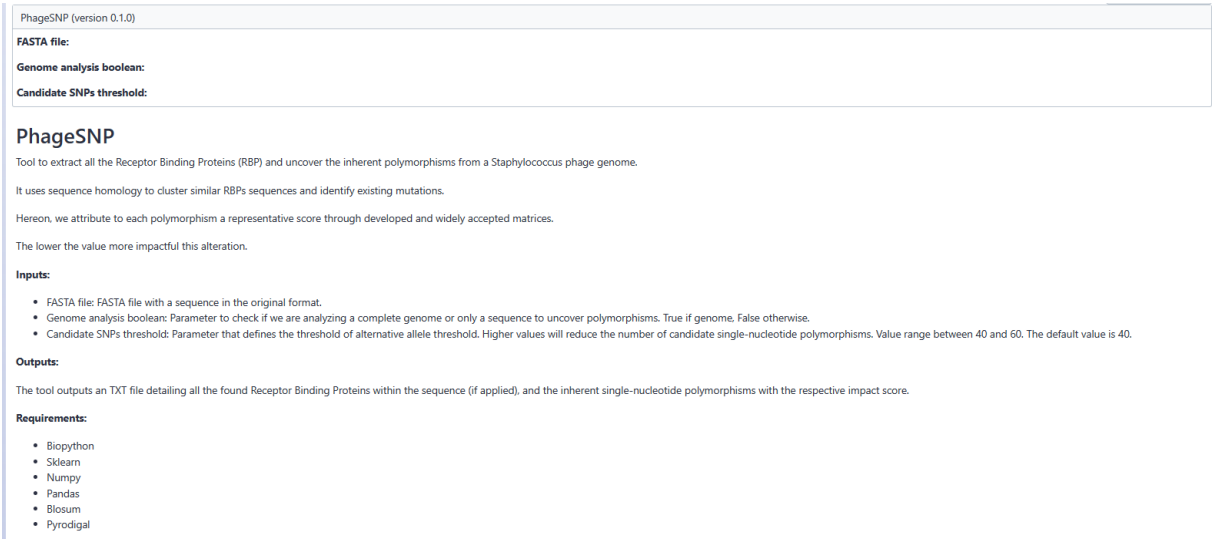


Figure 13: Tool overview resuming the principal steps executed to generate the output file of SNP scoring of each RBP sequence. 1) Given the input sequence or genome by the user, the Pyrodigal package returns the ORFs within the genome if applied; 2) Our chosen ML model predicts and filters the existent RBPs; 3) SNP identification according to the default threshold value or the inputted by the user, aided by clustering and alignment methods. Lastly, SNP scoring according to multiple substitution matrices; 4) Final pretty result, detailing the RBP nucleotide sequence, SNPs found, and respective impact score.

3.4.4 Galaxy Publishing

Galaxy publishing required multiple alterations to the PhageSNP code development to support a public interface execution. For this purpose, we placed the essential files on the same level for straightforward access and removed the dataset construction development, as it is unnecessary for the tool's acting. Furthermore, we altered the input parameters, excluding the "–seq" option due to its unsuitable integration in the interface. Additionally, we developed an XML file detailing the PhageSNP requirements and output

files. It comprehends the extensive interpretation of the tool and provides a simple manner for interface manipulation. Hereon, PhageSNP is published on the ToolShed platform, as visualized in figure 14, to be available for installation and usage on the Galaxy interface.


PhageSNP (version 0.1.0)

FASTA file:

Genome analysis boolean:

Candidate SNPs threshold:

PhageSNP

Tool to extract all the Receptor Binding Proteins (RBP) and uncover the inherent polymorphisms from a Staphylococcus phage genome.

It uses sequence homology to cluster similar RBPs sequences and identify existing mutations.

Hereon, we attribute to each polymorphism a representative score through developed and widely accepted matrices.

The lower the value more impactful this alteration.

Inputs:

- FASTA file: FASTA file with a sequence in the original format.
- Genome analysis boolean: Parameter to check if we are analyzing a complete genome or only a sequence to uncover polymorphisms. True if genome, False otherwise.
- Candidate SNPs threshold: Parameter that defines the threshold of alternative allele threshold. Higher values will reduce the number of candidate single-nucleotide polymorphisms. Value range between 40 and 60. The default value is 40.

Outputs:

The tool outputs an TXT file detailing all the found Receptor Binding Proteins within the sequence (if applied), and the inherent single-nucleotide polymorphisms with the respective impact score.

Requirements:

- Biopython
- Sklearn
- Numpy
- Pandas
- Blosum
- Pyrodigal

Figure 14: ToolShed preview of the PhageSNP interface, detailing the input parameters and the package requirements for normal execution. Moreover, it provides a simple interpretation and information about the output files. Page available through the URL: https://toolshed.g2.bx.psu.edu/view/phage_snp/phage_snp/465bfb24dcc5

Chapter 4

Results and Discussion

4.1 Constructed Dataset

The dataset dimension is fundamental for precise ML model training, as the value between each sample type urges balance. PhageSNP tool specializes in the analysis and prediction of RBPs. Accordingly, these are denoted as the positive sample sequences and offset by the negative ones. The strategies detailed in the methods "Positive Samples Extraction" section ultimately yielded a cumulative total of 2718 protein sequences, with the results of each approach documented in table 6.

Ultimately, each strategy contributed differently to the final dataset size. Despite the immense number of proteins analyzed, the initial method delivered only 276 sequences. This outcome stems from the inherent redundancy within the Entrez search mechanism, the exploration through genome records, and the imposed term limitation, which excludes prophage sequences.

Conversely, the UniProt was the least time-intensive, involving a mere 396 potential proteins for the dataset. Here, we extracted 22 valid sequences, whereas several remaining entries duplicated those obtained through the previous method.

Given the limited contribution of the initial strategies to the dataset, we focused on possible escaped sequences through conserved domains and protein families of RBPs. It yielded an additional 376 new proteins, enhancing the dataset's comprehensiveness.

As a last resort, the fourth method posed inherent volatility due to numerous DNA annotation gaps, which we aimed to address through subsequent procedures. Despite this, it significantly contributed to the dataset's dimensions, returning a substantial 2044 new samples. Nonetheless, numerous lacked DNA sequence annotations or did not overcome the verification sampling procedure.

Throughout the extraction of the 2718 total samples, systematic verification was in place to identify and avoid truncations and repetitions. Consequently, their posterior revision excluded a marginal number of proteins. Contrarily, the endeavors to recover gene annotations of the proteins proved to be mostly

ineffective. As a result, 718 samples retrieved by the fourth method were removed from the dataset, culminating in 2000 as the final dimension.

Table 6: Table representation of each method sequence results. Finally, the dataset encompassed 2718 samples, further reduced to 2000 by the validation dataset approaches. The 718 removed samples correspond to accession IDs retrieved by the fourth process, in which we only identified the amino acid sequences and failed to extract the nucleotide sequence through the complex emplaced strategies.

Method	**Accession IDs**	**Sequences Extracted**
1	73,112	276
2	396	22
3	3,382	376
4	73,758	2,044 **(-718)**
Total	150,646	2,718
Final	-	2,000

4.2 Evaluation of RBP Prediction Model

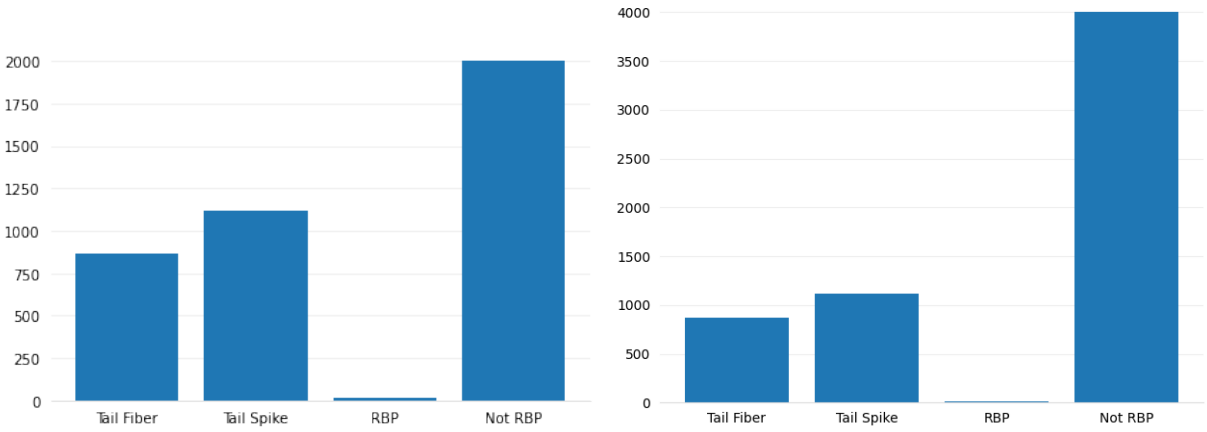
The RBP prediction model addresses the need for streamlining the process of identifying significant genes within a non-annotated genome, eliminating the necessity for numerous intricate steps. Phage gene annotation pipelines primarily revolve around homology, alignments, and unsupervised procedures (179). These use extensively trained ML models and assemble several bioinformatics tools (192). However, due to the extensive gene diversity and the vast array of phage strains, annotation tools that lack specificity can falter in accurately identifying genes. Moreover, the absence of clear homology provokes inadequate alignments, severely compromising the tool's accuracy (179). Hence, we intended to precisely identify the RBPs of the *Staphylococcus* phage by employing meticulous procedures tailored to achieve the intended goal.

4.2.1 Dataset Labeling

The mining of positive samples relies on the requirement that they include a keyword associated with RBPs, such as "tail fiber" and "tail spike." Due to the redundancy of some descriptions, we incremented the "receptor binding" keyword in the search mechanism. A classification dataset requires each sample

to be included in a predetermined class. As a result, we assigned a numerical value to each extracted sample annotation, leading to four distinct categories, including the classification of negative samples.

The two constructed datasets are represented in figure 15. As previously mentioned, they only differ in the frequency of the "Not RBP" or negative classification. It prevails over the others in each case, granting the model contact with several genes distinct from RBPs. On the other hand, both datasets possess only 15 samples with the "RBP" nomination, requiring these samples and classification removal to prevent model underfitting. Thus, the datasets remain with three categories of classification: "Tail Fiber", "Tail Spike", or "Not RBP".



(a) Dataset with 2,000 negative samples.

(b) Dataset with 4,000 negative samples.

Figure 15: Labeling class values. Due to the low occurrence frequency of the "RBP" classification, we opted not to include it to prevent the trained model from underfitting. As a result, only the remaining three classifications are taken into consideration.

4.2.2 K-mer Values Analysis

The variation in the k-mer size can produce a significant impact according to the intended purposes. Generally, while analyzing a complete genome, longer k-mers possess greater efficiency in the gene presence or absence labeling (184). However, as mentioned previously, limited computational resources limited us to studying the maximum length 6. To assess the impact of the different k-mer sizes, we summarize in table 7 all the obtained F1 scores for each model and used length.

Curiously, each model behaved differently according to the applied k-mer dimension. In both datasets, KNN, which displayed the worst F1 scores overall, significantly decreased its test values as the size increased. On the other hand, RF portrayed the opposite behavior while constantly increasing its score. The remaining models lack a clear proportional evolution between the F1 score and the k-mer size, except DT

in dataset B, which constantly increased their score. Despite LR and SVC exhibiting balanced values, both mostly demonstrate a maximum F1 score at the k-mer size 5, denoting better results than the highest length. With that in mind, increasing the "k" value and, consequently, the computational demand proves unnecessary due to the reduction in performance.

Yet, given the RF's variance, exploring higher parameter values could yield an intriguing outcome, potentially leading to a better model for our study's objective. In comparison, bioinformatics tools such as PlasForest (193) for sequence identification within genomes reach an f1-Score of 0.950 with an RF. The research highlights other k-mer-based methods with better performance in distinct predictions. Therefore, the 0.952 f1-score exhibits the RF high potential linked to our current feature extraction method.

At last, we opted for a k-mer size of 5, driven by the maximum F1-score of 0.963 with the SVC model.

Table 7: F1 scores comparison for each trained model and dataset with variation in the k-mer size. Dataset A corresponds to the one encompassing 2,000 negative samples, whereas dataset B represents the dataset with 4,000 "Not RBPs". We observe distinct results according to the model used, displaying diverse relations with the length variation. Ultimately, k-mer size of 5 portrayed the best F1 scores, promoting its choice as the parameter value for the final model development.

Dataset	K-mer Size	LR	KNN	DT	SVC	RF
A	3	0.958	0.898	0.898	0.955	0.932
	4	0.959	0.869	0.896	0.959	0.933
	5	0.961	0.838	0.900	0.963	0.941
	6	0.959	0.830	0.890	0.957	0.949
B	3	0.961	0.931	0.910	0.956	0.945
	4	0.962	0.911	0.916	0.962	0.949
	5	0.961	0.886	0.923	0.962	0.951
	6	0.959	0.885	0.929	0.962	0.952

4.2.3 Final Model Results

PhageSNP has the unique characteristic of identifying RBPs in Staphylococcus phages. The extraction of dataset features plays a pivotal role in maximizing the accuracy of model training. For every sequence within the dataset, we generate a counting array to assess the frequencies of k-mers. We consider the two datasets for 2,000 and 4,000 negative samples, referring to them as A and B, respectively. Our training

arrays have a dimension of 63,500 for dataset A and 64,668 for dataset B, aligning with the optimal k-mer size value. Furthermore, table 8 exhibits both dataset models' performance at different metric scores.

Dataset B exhibits overall improved performance compared to dataset A, given the higher F1-score displayed by the models. However, the linear models reveal similar scores between the datasets, with a slight advantage in the SVC method on dataset A. Despite the result that could stem from inherent randomness or a more advantageous data division, the lower F1-score on B could arise from the sensitivity of linear models to imbalanced data, which complicates pattern identification. Conversely, dataset A is more balanced, as linear models should conduct better in these conditions. Therefore, A will be the chosen dataset to build our best model, and further examination will focus on the results exhibited by this. Nevertheless, both datasets exhibit similar behavior across the strategies, whereas dataset B shows less pronounced differences.

Among all the supervised models tested, KNN exhibited the lowest accuracy. The chosen feature extraction method is not optimal for the KNN algorithm, as it is susceptible to high-dimensional data, commonly referred to as the "curse of dimensionality" (194). Thus, its under-performance is evident by the minimum value of 0.838 among all the metrics score. Within the tested range of neighbor numbers, the maximum value of 20 emerged as the optimal parameter choice. However, considering the unsatisfactory F1 score and the imminence of underfitting with the increase of the neighbors, the model proves inadequate for the use case.

In contrast, LR and SVC demonstrate the highest F1 scores of 0.961 and 0.963, respectively. Notably, the optimal hyperparameter for the SVC kernel is the "linear" option. Unsurprisingly, linear models excel in scenarios with less complex feature relationships similar to ours. Its effectiveness is evidenced by the consistent performance of linear models in classifying explicit attributes, often equated with more complex models (195). Considering the straightforward approach of k-mer counting and high-dimensional data, these proficiently discern patterns of nucleotide repetitions in each sequence classification, promoting the best testing values in the model training.

Lastly, RF models displayed an F1 score of 0.941, whereas a unique DT approach significantly reduced the F1 score to 0.900. The minor score relative to the linear models could be attributed to the moderately small dataset constructed. This is evident in dataset B, which has larger dimensions and portrays significantly better results. RFs thrive in large non-linear datasets, enabling them to develop various DTs (196). Consequently, it may be prone to overfitting and outliers, influenced not only by the dataset dimensions but also by the linear behavior of the use case.

In conclusion, the SVC model of dataset A stood out with the best metrics score. Therefore, we will use

it for the model performance assessment and constitute the mechanism for predicting RBP sequences within an unannotated genome.

Table 8: A summary of each model metrics score, displaying the optimal hyperparameters when applied. Dataset A corresponds to the one encompassing 2,000 negative samples, whereas dataset B represents the dataset with 4,000 "Not RBPs". The SVC model exhibits the best score on dataset A with 0.963, establishing as the best approach for our tool RBP prediction feature. Relative to all the models, the linear ones produced the best results, whereas KNN was the worst. Finally, RF and DT yielded satisfactory results, with RF standing out.

Dataset	Models	Accuracy	Precision	Recall	F1	Optimal HyperParameters
A	LR	0.961	0.961	0.961	0.961	-
	KNN	0.838	0.838	0.838	0.838	{ weights: "uniform", p: 2 n_neighbors: 20 }
	DT	0.900	0.900	0.900	0.900	-
	SVC	0.963	0.963	0.963	0.963	{ kernel: "linear" }
	RF	0.941	0.941	0.941	0.941	{ n_estimators: 50 }
B	LR	0.961	0.961	0.961	0.961	-
	KNN	0.886	0.886	0.886	0.886	{ weights: "uniform", p: 2 n_neighbors: 20 }
	DT	0.923	0.923	0.923	0.923	-
	SVC	0.962	0.962	0.962	0.962	{ kernel: "linear" }
	RF	0.951	0.951	0.951	0.951	{ n_estimators: 50 }

Additionally, we intended to evaluate potential DL models that could enhance our ability to predict RBP sequences compared to the chosen SVC model. For this purpose, we present in table 9 the performance of several NN strategies for both datasets since the behavior may vary with the negative sample frequency. In particular, the RNN model exhibits the lowest scores, showing a decline compared to the methods outlined in table 8. Although it has significantly improved in dataset B, it still proves the unfitting application of this NN for genomic purposes when not paired with CNNs. Conversely, while the FNN method has popularity on some biological issues and yields satisfactory results, it falls short compared to other alternatives.

Finally, the CNN model is distinguished as the best-performing model in both datasets, confirming its

reputation and the inference that this method is the most suitable for our genomic purpose (81). It exceeds slightly on dataset A, exhibiting an F1-score of 0.968 and surpassing the previously chosen SVC with the 0.963 F1-score. Other research, such as Gunasekaran et al. (182) or Helaly et al. (197) works, evaluate multiple DL strategies for sequence classification. Similarly, they conclude that CNN coupled with k-mer provides the best results. Therefore, incorporating a more complex CNN model could potentially boost the predictive capabilities of the PhageSNP tool for RBP sequences, a crucial task for future developments.

Table 9: A summary of each model metrics score, displaying the optimal hyperparameters when applied. Dataset A corresponds to the one encompassing 2,000 negative samples, whereas dataset B represents the dataset with 4,000 "Not RBPs". These DL approaches are taken posteriorly to the conclusion of the best ML model for the PhageSNP tool incorporation. Therefore, they are potential methods to improve the PhageSNP's ability to predict RBP sequences in the future. Specifically, the CNN model exhibits the best score of 0.968 with dataset A, establishing the best approach to integrate the tool RBP prediction feature.

Dataset	Models	Accuracy	Precision	Recall	F1	Optimal HyperParameters
A	FNN	0.957	0.957	0.957	0.957	{ hidden_layer_sizes: 200 }
	CNN	0.968	0.968	0.968	0.968	{ hidden_layers: 32, kernel_size: 2, activation_function: "relu", batch_size: 32 }
	RNN	0.488	0.488	0.488	0.488	{ hidden_layers: 32 }
B	FNN	0.956	0.956	0.956	0.956	{ hidden_layer_sizes: 200 }
	CNN	0.967	0.967	0.967	0.967	{ hidden_layers: 32, kernel_size: 2, activation_function: "relu", batch_size: 32 }
	RNN	0.649	0.649	0.649	0.649	{ hidden_layers: 32 }

4.2.4 Model Performance Assessment

Model training results are worthless when not validated in actual use cases. Therefore, the 100 random genome extraction from NCBI helps as a model validation procedure. Within the 100 genomes, Pyrodigal returned 12,011 ORFs for the following model classification. Unsurprisingly, approximately 98%, or 11 680 sequences, of these predictions are accurately labeled as "No Classification", corresponding to True Negatives (TN).

Despite the good results in predicting non-RBPs, the assessment of the correct labeling can be challenging due to the generic original gene description provided by the GenBank annotation. However, we can evaluate the 331 classifications left based on the precision provided by the actual characterization. In this context, the results initially indicate 18 wrongly categorized RBPs, denoting them as False Negatives (FN), and 29 mistakenly predicted as RBPs, the False Positives (FP). Thus, 47 sequences out of the 12,011 ORFs were inaccurately classified, representing approximately 0.39% of the dataset. On the other hand, the confirmed True Positives (TP) are of a considerably higher number of 114 correctly predicted tail fiber proteins.

Nonetheless, more than half of the predictions, or 170 sequences, require careful curation to comprehend the most suitable category to compose. Considering that these incorporate the RBPs labeling, they could integrate either the TP or FP analysis categories. Among these, the inconclusive protein description encompasses a wide range of real labelings, including hypothetical proteins, structural proteins, virion proteins, tail proteins, Domain of Uknown Function (DUF) proteins, phage uncharacterized proteins (XkdX), and many others. Therefore, protein BLAST has value in reaching a conclusive description for these classifications.

Manual curation through BLAST led to a definitive classification for the majority of predictions. However, some proteins revealed close relationships with others containing inconclusive descriptions. Consequently, these will comprise the FP category due to the absence of receptor binding protein resemblance. Finally, the TP and FP extended respectively to 212 and 101 sequences, corresponding to 64% correct and 39% wrong forecasts of actual RPB ORF sequences.

Nevertheless, as mentioned above, the misprediction percentage might be misleading. It occurs due to the inclusion of ORFs with ambiguous descriptions, and the remaining inaccurately predicted sequences are closely related to RBPs, such as baseplate and tail tape measure proteins, despite their different functions.

For further analysis, we enumerated in table 11 all the classification outcome values and multiple metric methods. The model demonstrated an efficiency of 99%, an F1-score of 0.78, and an MCC value of 0.79, with a misprediction of only 119 ORFs. However, due to the high prevalence of non-RBP sequences, the efficiency metric may be misleading, so the F1-score or MCC should be considered a more reliable measure. Our model struggles with distinguishing RBPs from closely related tail components and addressing ambiguous protein functions submitted to public databases. This is reflected in the precision metric, which yields a value of 0.68. On a brighter note, the recall metric highlights a strength of the model in well-performing in detecting RBP sequences.

Table 10: Table summary of the statistics observed in the validation process of the model developed to predict RBPs in Staphylococcus phages. We analyzed 12,011 sequences, with a majority belonging to the category of "No Classification" or ORFs apart from RBPs. This class demonstrates high efficiency, proving the model works well in discerning RBPs of other sequences. Relative to the tail spikes, they appear in much lower frequency with only three occurrences and accurately labeled in the validation process. Tail fibers are the most prominent among the RBPs and display difficulty distinguishing them from close-related tail sequences of other functions. However, the correctly predicted tail fibers exceed more than double the incorrectly labeled, being the misclassified composed of inconclusive sequence functions and similar ORFs to RBPs.

Actual / Predicted	**Tail Fiber**	**Tail Spike**	**No Classification**	**Total**
Tail Fiber	**209**	0	18	227
Tail Spike	0	**3**	0	3
No Classification	101	0	**11,680**	11,781
Total	310	3	11,698	**12,011**

Table 11: Confusion matrix enumerating all the outcome values obtained from the actual and predicted classifications. The outcome classification denotes the correct/wrong indication of the presence/absence of the studying characteristic. Therefore, we may use them to analyze the model's performance on multiple standards by metrics calculation. The accuracy metric can be misleading since it is susceptible to unbalanced data. This case study exemplifies the presence of unbalanced data, characterized by the infrequent appearance of RBP sequences within a genome. Thus, the remaining metrics are noteworthy because they provide better insight into the model's behavior. F1-score and MCC (Matthews correlation coefficient) are especially relevant since they embody the most effective and robust metric methods. With respective scores of 0.78 and 0.79, our model demonstrates the potential for significant improvement.

Outcome	**Value**	**Metric**	**Value**
True Positive (TP)	212	**Accuracy**	0.99
True Negative (TN)	11,680	**Precision**	0.68
False Positive (FP)	101	**Recall**	0.92
False Negative (FN)	18	**F1-score**	0.78
Total Population	12,011	**MCC**	0.79

4.3 SNP Scoring

For the SNP scoring stage, we use mechanisms such as clustering and alignment to identify mutations within and theoric substitution matrices for impact evaluation. In this regard, we produced 219 similarity clusters within the positive samples sub-dataset, being that only 62 respected the minimum sequence in-depth threshold.

Unfortunately, the absence of annotated SNPs in RBPs from phages infecting *Staphylococcus spp.* limits the possible validation approaches to sustain the SNP impact scores. Nevertheless, we propose candidate polymorphisms supported by cross-referencing multiple substitution matrices subject to extensive investigations (189). Hence, non-conservative alterations, representing substitutions of chemically and structurally different amino acids, are determined according to the various substitution scores correctly weighted to the use case.

To assess the value intervals retained by the PhageSNP scoring phase, we extracted all RBP SNPs that modify phage's host range from one of the few suitable investigations, the work from Botka et al. in

2019 (11). It researches phages' host range recognition of *Staphylococcus spp.* and details 13 different SNPs that arose in variants 812 and 812h1. We performed a boxplot analysis of the impact scores obtained from the PhageSNP tool, as visualized in figure 16 for boxplot A. It resulted in a distribution between 1 and -4, matching our final score interval threshold previously mentioned. Research centering on this topic is scarce, so we examined mutations from phages that recognize different bacterial species, such as *Shigella flexneri* (198), *Pseudomonas aeruginosa* (199), and *Streptococcus thermophilus* (200). Boxplot B represents the addition of all the 13 examined SNP scores plus the included in boxplot A. Out of the 26 values, only 4 are outliers for our threshold interval and represent SNPs from phages specific for bacterial species other than *Staphylococcus spp.*

Besides the SNP scoring interval assessment, we tested the entire PhageSNP tool pathway on the RBP sequences incorporating SNPs from boxplot A. With this, we highlight the 5 SNPs successfully identified, whereas the remaining did not fulfill the characteristics to be considered a candidate SNP. Furthermore, it retrieved an additional 11 candidate SNPs within the sequences that could yield some biological importance.

Thus, our score boundary is adequate for RBPs that recognize *Staphylococcus spp.* It eliminates insignificant mutations with minimal impact and drastic polymorphisms that shorten the length or alter the protein function. With this, the remaining are significant ones contributing to a minor protein change, such as the expansion of the recognizable hosts by the RBP.

Notably, this procedure is unique in the phage RBP field, as investigations centering on SNPs' impact on the host range are scarce. Tools like NanoCaller (148) and PhD-SNP (201) are renowned for their efficiency in the identification and scoring of SNPs. For this purpose, both develop ML models on datasets built from structured databases, promoting a more efficient strategy. In our case, the lack of annotated SNPs is a significant setback for a similar approach. Despite this, we introduce PhageSNP as an alternative preliminary step towards a more plain strategy for expanding the host range of RBPs.

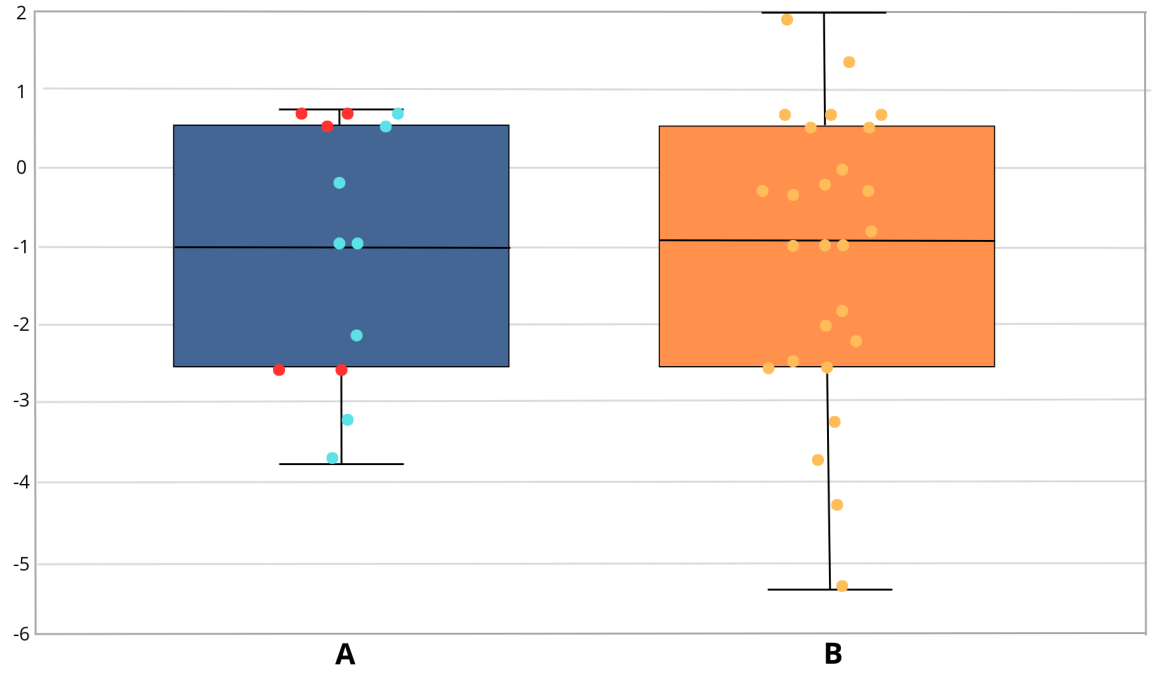


Figure 16: Boxplots that visually represent the impact scores of SNPs referenced in prior studies. Boxplot A represents the currently analyzed SNPs from phages that specifically infect *Staphylococcus spp.*, whereas B includes boxplot A plus other polymorphisms from phages with different host recognition. Relative to A, we observe a distribution of the scores between 1 and -4, with interval expansion when considering other species hosts. Additionally, we highlight in red the SNPs that the PhageSNP tool successfully identified as candidate polymorphisms. We computed all the obtained scores with the SNP scoring method implemented within PhageSNP.

4.4 Galaxy Application

In figure 17, we present the PhageSNP tool in the Galaxy platform available for public usage. We summarize its functioning, detailing the package and data requirements with a straightforward interface. Despite PhageSNP requiring minimal performance time, Galaxy allows the user multiple executions in an efficient environment. Therefore, it does not demand any computational efforts by the user.

Regardless, it produces an output file available for download, similar to the PhageSNP operation on the command line. As previously mentioned, we removed some features, such as direct sequence input, on the Galaxy integration to reduce the execution complexity. Figure 18 exemplifies the structure of the output file produced, listing the sequences and respective SNPs predicted. Here, we are considering a genome analysis, given the illustration of multiple RBPs. However, if the user specifies otherwise,

only a single sequence with a similar visual layout is displayed. In addition to the complete extracted sequence, we indicate its predicted classification and position within the genome. Likewise, we underline for the retrieved SNPs the nucleotide and amino acid changes, their position, and impact score on the protein.

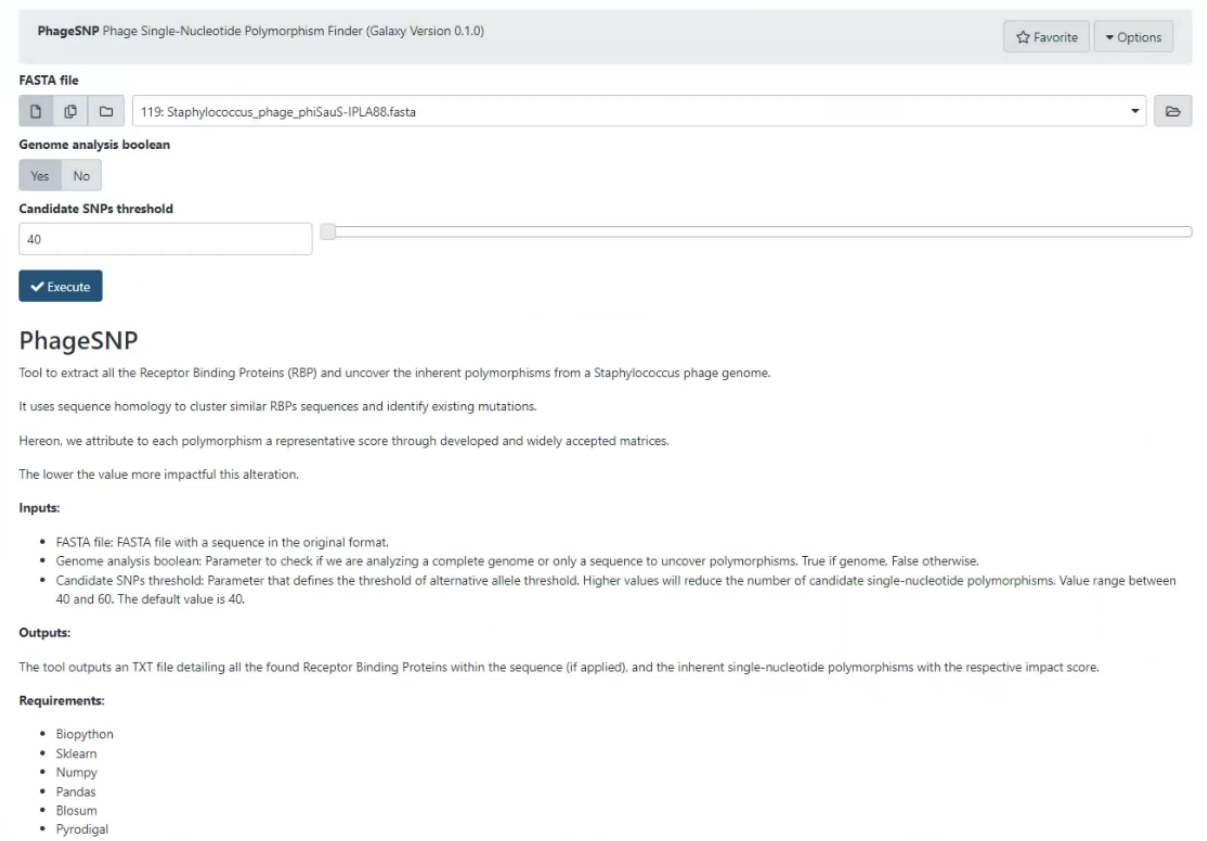


Figure 17: Visual representation of the PhageSNP tool on the Galaxy platform. We present a detailed description of its functioning, documenting package requirements and the input/output data. As input, it is instructed an unchanged FASTA with the sequence, indication of genome analysis, and the threshold value to consider. Page available in the URL: https://galaxy.bio.di.uminho.pt/root?tool_id=toolshed.g2.bx.psu.edu/repos/phage_snp/phage_snp/PhageSNP/0.1.0

```
Classification: Tail Fiber  ||  Position: (85671, 86046, 1)

Sequence:
ATGGCTGTTAAATATGATATAGGTAATAATGAGATAGTATTACATTTAAGAGAAGGTAAATATATAACAGGGTTTACAACAGTAGGAGGGTATGATAAGGA
GTTAGGACAAGTAAAAGTTAATAGAGAAATCTTACCTGCTTACTTCTTTGATAATTTTGCCTATGAAAGATATTTGTATTATAGTAAACCTGAAGAGGTTA
TAGAAAATAAAAACTATGTACCACCACAAATCAATGATGATGATGAGGAATCCCAACAAATTACTGTACCTAAAGAACAATATGATAGTTTAAAAGAAGAA
CTAGAGCTTATGAGAAAACAACAAGAAGCTATGATGGAAATGCTTCAAAAGCTCTTAGGTCAAAAGGGGTAA

Predictions:
        Position: 241  || Score: -1.895  || Substitutions: G -> A, G -> D
        Position: 82   || Score: -0.995  || Substitutions: C -> T, A -> V
        Position: 88   || Score: -0.835  || Substitutions: C -> G, A -> G
        Position: 25   || Score: -0.205  || Substitutions: G -> A, S -> N
        Position: 9    || Score: 0.055   || Substitutions: C -> A, Q -> K
        Position: 27   || Score: 0.77    || Substitutions: G -> A, D -> N
        Position: 237  || Score: 0.77    || Substitutions: A -> G, N -> D

--------------------------------------------------------
Classification: Tail Fiber  ||  Position: (86052, 87429, 1)

Sequence:
ATGGCATTAAATTTTACTACAATAACGGAAAACAATGTTATTAGAGACCTGACTACTCAGGTCAATAACATTGGAGAAGAATTAACAAAAGAAAGAAATAT
ATTTGACATTACCGATGATTTAGTTTATAATTTTAATAAATCACAGAAAATTAAACTAACTGATGATAAAGGATTAACTAAATCTTATGGAAACATAACAG
CCCTTAGAGATATAAAAGAACCTGGTTATTACTATATAGGTGCTAGAACATTAGCAACATTATTAGATAGACCTGATATGGAATCTCTTGATGTTGTTTTA
CATGTAGTACCTCTTGATACTTCTAGTAAGGTAGTTCAACATTTATATACACTATCTACTAACAATAACCAAATTAAAATGTTATATAGATTTGTCTCAGG
AAACTCTAGTTCAGAATGGCAATTTATTCAAGGATTACCTAGTAATAAAAATGCTGTTATATCAGGAACTAATATTTTAGATATAGCTTCACCAGGTGTTT
ACTTTGTTATGGGAATGACAGGAGGAATGCCTAGTGGAGTAAGCTCCGGATTTTTAGACTTAAGTGTAGATGCTAATGATAATAGATTAGCTAGACTAACT
GATGCTGAAACCGGTAAAGAATATACTAGCATTAAGAAACCTACAGGAACATACACAGCCTGGAAAAAAGAATTTGAGCCAAAAGATATGGAGAAATATCT
ACTAAGTAGTATTAGAGACGATGGTAGTGCATCATTCCCACTCCTAGTTTATACTAGTGATAGTAAAACATTTCAACAAGCTATTATAGACCATATAGATA
GAACAGGTCAAACAACCTTTACTTTCTATGTTCAAGGCGGTGTATCCGGTTCCCCTATGTCGAATAGTTGTCGAGGGTTATTCATGTCAGACACACCTAAT
ACTTCTAGTTTACATGGTGTTTACAATGCTATAGGTACAGATGGTAGAAATGTAACAGGTTCAGTGGTAGGTAGTAATTGGACTTCACCAAAAACATCCCC
TTCTCATAAAGAATTATGGACAGGAGCACAATCATTCTTATCTACAGGAACTACTAAGAATTTATCAGATGATATTAGTAACTACTCTTATGTAGAAGTTT
ATACTACACATAAGACAACAGAGAAGACTAAAGGTAATGACAATACAGGAACTATATGTCATAAGTTTTATTTAGATGGTAGTGGAACTTACGTTTGTTCA
GGTACATTTGTTTCCGGGGATAGAACCGATACAAAACCCCCTATCACGGAGTTTTATAGAGTAGGTGTATCTTTTAAAGGTTCTACATGGACTCTTGTAGA
TAGTGCAGTACAAAATAGTAAAACTCAATACGTTACAAGAATTATAGGTATTAATATGCCATAG

Predictions:
        Position: 1117 || Score: -1.445  || Substitutions: A -> C, K -> T
        Position: 769  || Score: -0.205  || Substitutions: A -> G, N -> S
        Position: 1152 || Score: 0.77    || Substitutions: G -> A, D -> N
```

Figure 18: An illustrative example of an output file with the genome analysis considered. It includes detailed information about the classification, location, and corresponding ORF sequence of the anticipated RBPs in the analyzed genome. Furthermore, for each predicted SNP, we document the nucleotide and amino acid changes, its position in the sequence, and the impact score on the resulting protein.

Chapter 5

Conclusions and future work

5.1 Conclusions

PhageSNP is a bioinformatics tool for *Staphylococcus* phage RBP prediction that integrates several approaches to identify and categorize SNPs' theoretical impact on the host range expansion. For this purpose, it comprehends an SVC model, displaying an f1-score of 0.963 and 0.780 in the testing and validation phases, respectively. Subsequently, it presents a groundbreaking strategy to annotate relevant SNPs aligned with our objective. Here, multiple substitution matrices serve as a basis to assess each SNP's potential impact on the resulting protein. Notably, it constrains the predictions by a range value corresponding to the interval where they exert the most influence on the host range expansion. Given the innovative feature of the PhageSNP tool, it exhibits significant potential for further development and refinement, mainly in the SNP annotation segment.

5.2 Prospect for future work

The future work resolves various aspects that could improve PhageSNP efficiency. Firstly, dataset quality improvement is viable through rejected sequences integration from the positive sample extraction methods. These would contribute to a better dataset since they represent RBP's closely related proteins, offering the model a better overview of how to distinct similar sequences. Moreover, considering the noticeable enhancement of the dataset with 4,000 negative samples, a significantly greater quantity of these could potentially lead to intriguing outcomes in non-linear ML methodologies.

Additionally, it would be beneficial to integrate a well-developed CNN model to replace our current RBP prediction strategy. As evidenced in the results and discussion section, it denotes better test performance despite its simple layer complexity, being a fundamental upgrade for the PhageSNP tool. Moreover, we urge the exploration of more advanced feature extraction approaches. While "k-mer counting" is an ex-

ceptional method that yields high metric values in our model, its basic structure might overlook crucial patterns. Therefore, new strategies could potentially enhance the performance of the PhageSNP model and, consequently, benefit from the integration of NN that thrive in complex associations.

Ultimately, we would mainly focus on integrating an ML-based SNP impact prediction. As the phage therapy method evolves, investigations surrounding RBPs increase. These display a fundamental role in recognizing the host bacteria, so SNPs to broaden their host range are vital. Therefore, as publicly available data increases, our objective of constructing a viable ML dataset becomes feasible. For this purpose, we intend to collect all the available data to maximize the dataset dimensions. With this, we would employ similar evaluation strategies for the ML segment and ideally follow a DL approach, given the data nature and complexity. Lastly, with the replacement of SNP scoring for an ML method, the PhageSNP tool performance could suffer substantial upgrades.

Bibliography

[1] C. Hulo, P. Masson, A. Toussaint, D. Osumi-Sutherland, E. De Castro, A. H. Auchincloss, S. Poux, L. Bougueleret, I. Xenarios, and P. Le Mercier, "Bacterial virus ontology; coordinating across databases," *Viruses*, vol. 9, no. 6, p. 126, 2017.

[2] F. L. Nobrega, M. Vlot, P. A. de Jonge, L. L. Dreesens, H. J. Beaumont, R. Lavigne, B. E. Dutilh, and S. J. Brouns, "Targeting mechanisms of tailed bacteriophages," *Nature Reviews Microbiology*, vol. 16, no. 12, pp. 760–773, 2018.

[3] X.-F. Wang, C.-Q. Yu, L. Li, Z.-H. You, W.-Z. Huang, Y.-C. Li, Z.-H. Ren, and Y.-J. Guan, "Kgdcmi: A new approach for predicting circrna–mirna interactions from multi-source information extraction and deep learning," *Frontiers in Genetics*, vol. 13, p. 958096, 08 2022.

[4] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[5] T. Balodi, "Convolutional neural network with python code explanation: Convolutional layer: Max pooling in cnn." [Online]. Available: https://www.analyticssteps.com/blogs/convolutional-neural-network-cnn-graphical-visualization-code-explanation

[6] A. M. Sloan and L. Dudjak, "Bedside nurses: Champions of antimicrobial stewardship," *Critical Care Nurse*, vol. 40, no. 6, pp. 16–22, 2020.

[7] J. M. Munita and C. A. Arias, "Mechanisms of antibiotic resistance," *Microbiology spectrum*, vol. 4, no. 2, pp. 4–2, 2016.

[8] H. Oliveira, M. Sampaio, L. D. Melo, O. Dias, W. H. Pope, G. F. Hatfull, and J. Azeredo, "Staphylococci phages display vast genomic diversity and evolutionary relationships," *BMC genomics*, vol. 20, no. 1, pp. 1–14, 2019.

[9] X. Zhang, D. Xiong, J. Yu, H. Yang, P. He, and H. Wei, "Genetic polymorphism drives susceptibility between bacteria and bacteriophages," *Frontiers in Microbiology*, vol. 12, p. 609, 2021.

[10] L. Richardson, "Beautiful soup documentation," *April*, 2007.

[11] T. Botka, R. Pantček, I. Mašlaňová, M. Benešík, P. Petráš, V. Ržičková, P. Havlíčková, M. Varga, H. Žemličková, I. Koláčková *et al.*, "Lytic and genomic properties of spontaneous host-range kayvirus mutants prove their suitability for upgrading phage therapeutics against staphylococci," *Scientific reports*, vol. 9, no. 1, pp. 1–12, 2019.

[12] M. I. Hutchings, A. W. Truman, and B. Wilkinson, "Antibiotics: past, present and future," *Current opinion in microbiology*, vol. 51, pp. 72–80, 2019.

[13] K. Bush and P. A. Bradford, "β-lactams and β-lactamase inhibitors: an overview," *Cold Spring Harbor perspectives in medicine*, vol. 6, no. 8, p. a025247, 2016.

[14] J. Davies and D. Davies, "Origins and evolution of antibiotic resistance," *Microbiology and molecular biology reviews*, vol. 74, no. 3, pp. 417–433, 2010.

[15] K. Bush, "Past and present perspectives on β-lactamases," *Antimicrobial agents and chemotherapy*, vol. 62, no. 10, pp. e01 076–18, 2018.

[16] J. Osei Sekyere and E. Mensah, "Molecular epidemiology and mechanisms of antibiotic resistance in enterococcus spp., staphylococcus spp., and streptococcus spp. in africa: a systematic review from a one health perspective," *Annals of the New York Academy of Sciences*, vol. 1465, no. 1, pp. 29–58, 2020.

[17] E. M. Ferreira, L. C. Romero, M. d. L. R. d. S. d. Cunha, W. Malagó Junior, C. H. Camargo, W. Barioni Júnior, and L. F. Zafalon, "Persistence of staphylococcus spp. in milk from cows undergoing homeopathy to control subclinical mastitis," *BMC Veterinary Research*, vol. 18, no. 1, p. 273, 2022.

[18] K. Lisowska-Łysiak, R. Lauterbach, J. Międzobrodzki, and M. Kosecka-Strojek, "Epidemiology and pathogenesis of bloodstream infections in humans: a review," *Polish Journal of Microbiology*, vol. 70, no. 1, pp. 13–23, 2021.

[19] A. Hassoun, P. K. Linden, and B. Friedman, "Incidence, prevalence, and management of mrsa bacteremia across patient populations—a review of recent developments in mrsa management and treatment," *Critical care*, vol. 21, no. 1, pp. 1–10, 2017.

[20] S. Rattanapunya, A. Deethae, S. Woskie, P. Kongthip, and K. R. Matthews, "Occurrence of antibiotic-resistant staphylococcus spp. in orange orchards in thailand," *International journal of environmental research and public health*, vol. 19, no. 1, p. 246, 2021.

[21] G. Dubin, "Extracellular proteases of staphylococcus spp." 2002.

[22] R. J. Gordon and F. D. Lowy, "Pathogenesis of methicillin-resistant staphylococcus aureus infection," *Clinical infectious diseases*, vol. 46, no. Supplement_5, pp. S350–S359, 2008.

[23] F. D. Lowy, "Staphylococcus aureus infections," *New England journal of medicine*, vol. 339, no. 8, pp. 520–532, 1998.

[24] C. Laux, A. Peschel, and B. Krismer, "Staphylococcus aureus colonization of the human nose and interaction with other microbiome members," *Microbiology Spectrum*, vol. 7, no. 2, pp. 7–2, 2019.

[25] S. Lakhundi and K. Zhang, "Methicillin-resistant staphylococcus aureus: molecular characterization, evolution, and epidemiology," *Clinical microbiology reviews*, vol. 31, no. 4, pp. e00 020–18, 2018.

[26] A. Jenkins, B. A. Diep, T. T. Mai, N. H. Vo, P. Warrener, J. Suzich, C. K. Stover, and B. R. Sellman, "Differential expression and roles of staphylococcus aureus virulence determinants during colonization and disease," *MBio*, vol. 6, no. 1, pp. e02 272–14, 2015.

[27] O. Davies, "Who publishes list of bacteria for which new antibiotics are urgently needed," Feb 2017. [Online]. Available: https://www.who.int/news/item/27-02-2017-who-publishes-list-of-bacteria-for-which-new-antibiotics-are-urgently-needed

[28] J. Otter, A. Kearns, G. French, and M. Ellington, "Panton–valentine leukocidin-encoding bacteriophage and gene sequence variation in community-associated methicillin-resistant staphylococcus aureus," *Clinical microbiology and infection*, vol. 16, no. 1, pp. 68–73, 2010.

[29] T. Funaki, T. Yasuhara, S. Kugawa, Y. Yamazaki, E. Sugano, Y. Nagakura, K. Yoshida, and K. Fukuchi, "Sccmec typing of pvl-positive community-acquired staphylococcus aureus (ca-mrsa) at a japanese hospital," *Heliyon*, vol. 5, no. 3, p. e01415, 2019.

[30] G. W. Coombs, S. L. Baines, B. P. Howden, K. M. Swenson, and F. G. O'Brien, "Diversity of bacteriophages encoding panton-valentine leukocidin in temporally and geographically related staphylococcus aureus," *PloS one*, vol. 15, no. 2, p. e0228676, 2020.

[31] G. Balakirski, G. Hischebeth, J. Altengarten, D. Exner, T. Bieber, J. Dohmen, and S. Engelhart, "Recurrent mucocutaneous infections caused by pvl-positive staphylococcus aureus strains: a challenge in clinical practice," *JDDG: Journal der Deutschen Dermatologischen Gesellschaft*, vol. 18, no. 4, pp. 315–322, 2020.

[32] M. Otto, "Community-associated mrsa: what makes them special?" *International Journal of Medical Microbiology*, vol. 303, no. 6-7, pp. 324–330, 2013.

[33] S. Liu, K. Hon, G. S. Bouras, A. J. Psaltis, K. Shearwin, P.-J. Wormald, and S. Vreugde, "Aptc-c-sa01: A novel bacteriophage cocktail targeting staphylococcus aureus and mrsa biofilms," *International Journal of Molecular Sciences*, vol. 23, no. 11, p. 6116, 2022.

[34] S. Matsuzaki, M. Yasuda, H. Nishikawa, M. Kuroda, T. Ujihara, T. Shuin, Y. Shen, Z. Jin, S. Fujimoto, M. Nasimuzzaman *et al.*, "Experimental protection of mice against lethal staphylococcus aureus infection by novel bacteriophage ϕmr11," *The Journal of infectious diseases*, vol. 187, no. 4, pp. 613–624, 2003.

[35] P. Turnbull, "Bacillus cereus toxins," *Pharmacology & therapeutics*, vol. 13, no. 3, pp. 453–505, 1981.

[36] T. F. El-Arabi, M. W. Griffiths, Y.-M. She, A. Villegas, E. J. Lingohr, and A. M. Kropinski, "Genome sequence and analysis of a broad-host range lytic bacteriophage that infects the bacillus cereus group," *Virology journal*, vol. 10, no. 1, pp. 1–11, 2013.

[37] M. Kong and S. Ryu, "Bacteriophage pbc1 and its endolysin as an antimicrobial agent against bacillus cereus," *Applied and environmental microbiology*, vol. 81, no. 7, pp. 2274–2283, 2015.

[38] G. F. Hatfull and R. W. Hendrix, "Bacteriophages and their genomes," *Current opinion in virology*, vol. 1, no. 4, pp. 298–303, 2011.

[39] H.-W. Ackermann, "Phage classification and characterization," pp. 127–140, 2009.

[40] G. P. Salmond and P. C. Fineran, "A century of the phage: past, present and future," *Nature Reviews Microbiology*, vol. 13, no. 12, pp. 777–786, 2015.

[41] D. Turner, A. N. Shkoporov, C. Lood, A. D. Millard, B. E. Dutilh, P. Alfenas-Zerbini, L. J. van Zyl, R. K. Aziz, H. M. Oksanen, M. M. Poranen *et al.*, "Abolishment of morphology-based taxa and change

to binomial species names: 2022 taxonomy update of the ictv bacterial viruses subcommittee," *Archives of Virology*, vol. 168, no. 2, p. 74, 2023.

[42] M. Zinke, G. F. Schröder, and A. Lange, "Major tail proteins of bacteriophages of the order caudovirales," *Journal of Biological Chemistry*, vol. 298, no. 1, 2022.

[43] C. G. Leon-Velarde, J. W. Jun, and M. Skurnik, "Yersinia phages and food safety," *Viruses*, vol. 11, no. 12, p. 1105, 2019.

[44] H.-W. Ackermann, H. M. Krisch, and A. M. Comeau, "Morphology and genome sequence of phage φ1402: A dwarf myovirus of the predatory bacterium bdellovibrio bacteriovorus," *Bacteriophage*, vol. 1, no. 3, pp. 138–142, 2011.

[45] A. Fokine and M. G. Rossmann, "Molecular architecture of tailed double-stranded dna phages," *Bacteriophage*, vol. 4, no. 2, p. e28281, 2014.

[46] W. Tam, L. G. Pell, D. Bona, A. Tsai, X. X. Dai, A. M. Edwards, R. W. Hendrix, K. L. Maxwell, and A. R. Davidson, "Tail tip proteins related to bacteriophage λ gpl coordinate an iron-sulfur cluster," *Journal of molecular biology*, vol. 425, no. 14, pp. 2450–2462, 2013.

[47] M. K. Mirzaei, H. Eriksson, K. Kasuga, E. Haggård-Ljungquist, and A. S. Nilsson, "Genomic, proteomic, morphological, and phylogenetic analyses of vb_ecop_su10, a podoviridae phage with c3 morphology," *PLoS One*, vol. 9, no. 12, p. e116294, 2014.

[48] D. Rakhuba, E. Kolomiets, E. S. Dey, and G. Novik, "Bacteriophage receptors, mechanisms of phage adsorption and penetration into host cell," *Polish journal of microbiology*, vol. 59, no. 3, p. 145, 2010.

[49] Z. J. Storms and D. Sauvageau, "Modeling tailed bacteriophage adsorption: Insight into mechanisms," *Virology*, vol. 485, pp. 355–362, 2015.

[50] T. Olszak, A. Latka, B. Roszniowski, M. A. Valvano, and Z. Drulis-Kawa, "Phage life cycles behind bacterial biodiversity," *Current medicinal chemistry*, vol. 24, no. 36, pp. 3987–4001, 2017.

[51] T. B. Broudy, V. Pancholi, and V. A. Fischetti, "Induction of lysogenic bacteriophage and phage-associated toxin from group a streptococci during coculture with human pharyngeal cells," *Infection and immunity*, vol. 69, no. 3, pp. 1440–1443, 2001.

[52] B. R. Lenneman, J. Fernbach, M. J. Loessner, T. K. Lu, and S. Kilcher, "Enhancing phage therapy through synthetic biology and genome engineering," *Current Opinion in Biotechnology*, vol. 68, pp. 151–159, 2021.

[53] S. Alhumaid, A. Al Mutair, Z. Al Alawi, A. J. Alzahrani, M. Tobaiqy, A. M. Alresasi, I. Bu-Shehab, I. Al-Hadary, N. Alhmeed, M. Alismail *et al.*, "Antimicrobial susceptibility of gram-positive and gram-negative bacteria: a 5-year retrospective analysis at a multi-hospital healthcare system in saudi arabia," *Annals of clinical microbiology and antimicrobials*, vol. 20, no. 1, p. 43, 2021.

[54] O. Schneewind and D. Missiakas, "Lipoteichoic acids, phosphate-containing polymers in the envelope of gram-positive bacteria," *Journal of bacteriology*, vol. 196, no. 6, pp. 1133–1142, 2014.

[55] J. Bertozzi Silva, Z. Storms, and D. Sauvageau, "Host receptors for bacteriophage adsorption," *FEMS microbiology letters*, vol. 363, no. 4, p. fnw002, 2016.

[56] R. Ruhal and R. Kataria, "Biofilm patterns in gram-positive and gram-negative bacteria," *Microbiological Research*, vol. 251, p. 126829, 2021.

[57] H. Oliveira, A. R. Costa, N. Konstantinides, A. Ferreira, E. Akturk, S. Sillankorva, A. Nemec, M. Shneider, A. Dötsch, and J. Azeredo, "Ability of phages to infect acinetobacter calcoaceticus-acinetobacter baumannii complex species through acquisition of different pectate lyase depolymerase domains," *Environmental microbiology*, vol. 19, no. 12, pp. 5060–5077, 2017.

[58] J. Klumpp, M. Dunne, and M. J. Loessner, "A perfect fit: Bacteriophage receptor-binding proteins for diagnostic and therapeutic applications," *Current Opinion in Microbiology*, vol. 71, p. 102240, 2023.

[59] M. Dunne and M. J. Loessner, "Modified bacteriophage tail fiber proteins for labeling, immobilization, capture, and detection of bacteria," pp. 67–86, 2019.

[60] A. Bhardwaj, A. S. Olia, and G. Cingolani, "Architecture of viral genome-delivery molecular machines," *Current opinion in structural biology*, vol. 25, pp. 1–8, 2014.

[61] O. I. North and A. R. Davidson, "Phage proteins required for tail fiber assembly also bind specifically to the surface of host bacterial strains," *Journal of Bacteriology*, vol. 203, no. 3, pp. e00 406–20, 2021.

[62] P. G. Leiman, F. Arisaka, M. J. Van Raaij, V. A. Kostyuchenko, A. A. Aksyuk, S. Kanamaru, and M. G. Rossmann, "Morphogenesis of the t4 tail and tail fibers," *Virology journal*, vol. 7, no. 1, pp. 1–28, 2010.

[63] J. Taslem Mourosi, A. Awe, W. Guo, H. Batra, H. Ganesh, X. Wu, and J. Zhu, "Understanding bacteriophage tail fiber interaction with host surface receptor: The key "blueprint" for reprogramming phage host range," *International Journal of Molecular Sciences*, vol. 23, no. 20, p. 12146, 2022.

[64] Y. Choi, H. Shin, J.-H. Lee, and S. Ryu, "Identification and characterization of a novel flagellum-dependent salmonella-infecting bacteriophage, ieps5," *Applied and environmental microbiology*, vol. 79, no. 16, pp. 4829–4837, 2013.

[65] M. Dunne, M. Hupfeld, J. Klumpp, and M. J. Loessner, "Molecular basis of bacterial host interactions by gram-positive targeting bacteriophages," *Viruses*, vol. 10, no. 8, p. 397, 2018.

[66] E. Yamashita, A. Nakagawa, J. Takahashi, K.-i. Tsunoda, S. Yamada, and S. Takeda, "The host-binding domain of the p2 phage tail spike reveals a trimeric iron-binding structure," *Acta Crystallographica Section F: Structural Biology and Crystallization Communications*, vol. 67, no. 8, pp. 837–841, 2011.

[67] J. Klumpp, M. Dunne, and M. J. Loessner, "A perfect fit: Bacteriophage receptor-binding proteins for diagnostic and therapeutic applications," *Current Opinion in Microbiology*, vol. 71, p. 102240, 2023.

[68] I. Vinga, C. Baptista, I. Auzat, I. Petipas, R. Lurz, P. Tavares, M. A. Santos, and C. São-José, "Role of bacteriophage spp1 tail spike protein gp21 on host cell receptor binding and trigger of phage dna ejection," *Molecular microbiology*, vol. 83, no. 2, pp. 289–303, 2012.

[69] A. Seul, J. J. Müller, D. Andres, E. Stettner, U. Heinemann, and R. Seckler, "Bacteriophage p22 tailspike: structure of the complete protein and function of the interdomain linker," *Acta Crystallographica Section D: Biological Crystallography*, vol. 70, no. 5, pp. 1336–1345, 2014.

[70] L. H. Cobb, J. Park, E. A. Swanson, M. C. Beard, E. M. McCabe, A. S. Rourke, K. S. Seo, A. K. Olivier, and L. B. Priddy, "Crispr-cas9 modified bacteriophage for treatment of staphylococcus aureus induced osteomyelitis and soft tissue infection," *PLoS One*, vol. 14, no. 11, p. e0220421, 2019.

[71] G. D. Hannigan, Q. Zheng, J. S. Meisel, S. S. Minot, F. D. Bushman, and E. A. Grice, "Evolutionary and functional implications of hypervariable loci within the skin virome," *PeerJ*, vol. 5, p. e2959, 2017.

[72] S. Le, X. He, Y. Tan, G. Huang, L. Zhang, R. Lux, W. Shi, and F. Hu, "Mapping the tail fiber as the receptor binding protein responsible for differential host specificity of pseudomonas aeruginosa bacteriophages pap1 and jg004," *PloS one*, vol. 8, no. 7, p. e68562, 2013.

[73] F. Sanger and E. Thompson, "The amino-acid sequence in the glycyl chain of insulin. 1. the identification of lower peptides from partial hydrolysates," *Biochemical Journal*, vol. 53, no. 3, p. 353, 1953.

[74] M. O. Dayhoff and R. S. Ledley, "Comprotein: a computer program to aid primary protein structure determination," in *Proceedings of the December 4-6, 1962, fall joint computer conference*, 1962, pp. 262–274.

[75] J. Gauthier, A. T. Vincent, S. J. Charette, and N. Derome, "A brief history of bioinformatics," *Briefings in bioinformatics*, vol. 20, no. 6, pp. 1981–1996, 2019.

[76] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," *Journal of molecular biology*, vol. 215, no. 3, pp. 403–410, 1990.

[77] F. S. Collins, M. Morgan, and A. Patrinos, "The human genome project: lessons from large-scale biology," *Science*, vol. 300, no. 5617, pp. 286–290, 2003.

[78] M. Egholm, M. Margulies, W. Altman, S. Attiya, J. Bader, L. Bemben, J. Berka, M. Braverman, Y. Chen, Z. Chen *et al.*, "Genome sequencing in open microfabricated high density picoliter reactors," *Nature*, vol. 437, pp. 376–380, 2005.

[79] W. R. McCombie, J. D. McPherson, and E. R. Mardis, "Next-generation sequencing technologies," *Cold Spring Harbor perspectives in medicine*, vol. 9, no. 11, 2019.

[80] V. C. Pezoulas, O. Hazapis, N. Lagopati, T. P. Exarchos, A. V. Goules, A. G. Tzioufas, D. I. Fotiadis, I. G. Stratis, A. N. Yannacopoulos, and V. G. Gorgoulis, "Machine learning approaches on high throughput ngs data to unveil mechanisms of function in biology and disease," *Cancer genomics & proteomics*, vol. 18, no. 5, pp. 605–626, 2021.

[81] W. S. Alharbi and M. Rashid, "A review of deep learning applications in human genomics using next-generation sequencing data," *Human Genomics*, vol. 16, no. 1, pp. 1–20, 2022.

[82] C. Notredame, "Recent progress in multiple sequence alignment: a survey," *Pharmacogenomics*, vol. 3, no. 1, pp. 131–144, 2002.

[83] Y. Wang, H. Wu, and Y. Cai, "A benchmark study of sequence alignment methods for protein clustering," *BMC bioinformatics*, vol. 19, no. 19, pp. 95–104, 2018.

[84] F. Sievers, A. Wilm, D. Dineen, T. J. Gibson, K. Karplus, W. Li, R. Lopez, H. McWilliam, M. Remmert, J. Söding *et al.*, "Fast, scalable generation of high-quality protein multiple sequence alignments using clustal omega," *Molecular systems biology*, vol. 7, no. 1, p. 539, 2011.

[85] M. Thakur, A. Bateman, C. Brooksbank, M. Freeberg, M. Harrison, M. Hartley, T. Keane, G. Kleywegt, A. Leach, M. Levchenko *et al.*, "Embl's european bioinformatics institute (embl-ebi) in 2022," *Nucleic Acids Research*, vol. 51, no. D1, pp. D9–D17, 2023.

[86] W. Li, L. Jaroszewski, and A. Godzik, "Clustering of highly homologous sequences to reduce the size of large protein databases," *Bioinformatics*, vol. 17, no. 3, pp. 282–283, 2001.

[87] W. Li and A. Godzik, "Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences," *Bioinformatics*, vol. 22, no. 13, pp. 1658–1659, 05 2006. [Online]. Available: https://doi.org/10.1093/bioinformatics/btl158

[88] L. Fu, B. Niu, Z. Zhu, S. Wu, and W. Li, "CD-HIT: accelerated for clustering the next-generation sequencing data," *Bioinformatics*, vol. 28, no. 23, pp. 3150–3152, 10 2012. [Online]. Available: https://doi.org/10.1093/bioinformatics/bts565

[89] S. J. Goodswen, J. L. Barratt, P. J. Kennedy, A. Kaufer, L. Calarco, and J. T. Ellis, "Machine learning and applications in microbiology," *FEMS microbiology reviews*, vol. 45, no. 5, p. fuab015, 2021.

[90] R. C. Deo, "Machine learning in medicine," *Circulation*, vol. 132, no. 20, pp. 1920–1930, 2015.

[91] L. Patel, T. Shukla, X. Huang, D. W. Ussery, and S. Wang, "Machine learning methods in drug discovery," *Molecules*, vol. 25, no. 22, p. 5277, 2020.

[92] R. Y. Choi, A. S. Coyner, J. Kalpathy-Cramer, M. F. Chiang, and J. P. Campbell, "Introduction to machine learning, neural networks, and deep learning," *Translational Vision Science & Technology*, vol. 9, no. 2, pp. 14–14, 2020.

[93] T. Jiang, J. L. Gradus, and A. J. Rosellini, "Supervised machine learning: a brief primer," *Behavior Therapy*, vol. 51, no. 5, pp. 675–687, 2020.

[94] A. H. Azam and Y. Tanji, "Peculiarities of staphylococcus aureus phages and their possible application in phage therapy," *Applied microbiology and biotechnology*, vol. 103, no. 11, pp. 4279–4289, 2019.

[95] Y. Nami, N. Imeni, and B. Panahi, "Application of machine learning in bacteriophage research," *BMC microbiology*, vol. 21, no. 1, pp. 1–8, 2021.

[96] C. Wang, J. Zhang, L. Cheng, J. Wu, M. Xiao, J. Xia, and Y. Bin, "Dpprom: a two-layer predictor for identifying promoters and their types on phage genome using deep learning," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 10, pp. 5258–5266, 2022.

[97] N. T. Issa, V. Stathias, S. Schürer, and S. Dakshanamurthy, "Machine and deep learning approaches for cancer drug repurposing," vol. 68, pp. 132–142, 2021.

[98] T. M. Mitchell, "Machine learning," 1997.

[99] R. P. Bonidia, L. D. Sampaio, D. S. Domingues, A. R. Paschoal, F. M. Lopes, A. C. de Carvalho, and D. S. Sanches, "Feature extraction approaches for biological sequences: a comparative study of mathematical features," *Briefings in Bioinformatics*, vol. 22, no. 5, p. bbab011, 2021.

[100] G. G. Sutton, O. White, M. D. Adams, and A. R. Kerlavage, "Tigr assembler: A new tool for assembling large shotgun sequencing projects," *Genome Science and Technology*, vol. 1, no. 1, pp. 9–19, 1995.

[101] Y. Guo, L. Hou, W. Zhu, and P. Wang, "Prediction of hormone-binding proteins based on k-mer feature representation and naive bayes," *Frontiers in Genetics*, vol. 12, p. 797641, 2021.

[102] M. Kokot, M. Długosz, and S. Deorowicz, "Kmc 3: counting and manipulating k-mer statistics," *Bioinformatics*, vol. 33, no. 17, pp. 2759–2761, 2017.

[103] S. Solis-Reyes, M. Avino, A. Poon, and L. Kari, "An open-source k-mer based machine learning tool for fast and accurate subtyping of hiv-1 genomes," *PloS one*, vol. 13, no. 11, p. e0206409, 2018.

[104] S. Sarmashghi, K. Bohmann, M. T. P Gilbert, V. Bafna, and S. Mirarab, "Skmer: assembly-free and alignment-free sample identification using genome skims," *Genome biology*, vol. 20, no. 1, pp. 1–20, 2019.

[105] R. P. Bonidia, D. S. Domingues, D. S. Sanches, and A. C. de Carvalho, "Mathfeature: feature extraction package for dna, rna and protein sequences based on mathematical descriptors," *Briefings in bioinformatics*, vol. 23, no. 1, p. bbab434, 2022.

[106] P. J. Cock, T. Antao, J. T. Chang, B. A. Chapman, C. J. Cox, A. Dalke, I. Friedberg, T. Hamelryck, F. Kauff, B. Wilczynski *et al.*, "Biopython: freely available python tools for computational molecular biology and bioinformatics," *Bioinformatics*, vol. 25, no. 11, pp. 1422–1423, 2009.

[107] Z. Mu, T. Yu, X. Liu, H. Zheng, L. Wei, and J. Liu, "Fegs: A novel feature extraction model for protein sequences and its applications," *BMC Bioinformatics*, vol. 22, no. 1, 2021.

[108] P. Yang, H. Huang, and C. Liu, "Feature selection revisited in the single-cell era," *Genome Biology*, vol. 22, no. 1, 2021.

[109] K. Tadist, S. Najah, N. S. Nikolov, F. Mrabti, and A. Zahi, "Feature selection methods and genomic big data: A systematic review," *Journal of Big Data*, vol. 6, no. 1, 2019.

[110] G. James, D. Witten, T. Hastie, R. Tibshirani *et al.*, *An introduction to statistical learning*. Springer, 2013, vol. 112.

[111] S. I. Bangdiwala, "Regression: simple linear," *International journal of injury control and safety promotion*, vol. 25, no. 1, pp. 113–115, 2018.

[112] J. Iranzo, I. Martincorena, and E. V. Koonin, "Cancer-mutation network and the number and specificity of driver mutations," *Proceedings of the National Academy of Sciences*, vol. 115, no. 26, pp. E6010–E6019, 2018.

[113] C. Oliveira, R. Ottman, and A. Orr-Urtreger, "Multi-center analysis of glucocerebrosidase mutations in parkinson disease," *N Engl J Med*, vol. 361, no. 17, pp. 1651–1661, 2009.

[114] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.

[115] R. de Vlaming, P. J. Groenen *et al.*, "The current and future use of ridge regression for prediction in quantitative genetics," *BioMed research international*, vol. 2015, 2015.

[116] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE transactions on information theory*, vol. 13, no. 1, pp. 21–27, 1967.

[117] B. Li, L. Yu, and L. Gao, "Cancer classification based on multiple dimensions: Snv patterns," *Computers in Biology and Medicine*, vol. 151, p. 106270, 2022.

[118] S. Li, E. J. Harner, and D. A. Adjeroh, "Random knn feature selection-a fast and stable alternative to random forests," *BMC bioinformatics*, vol. 12, pp. 1–11, 2011.

[119] G. Sahebi, P. Movahedi, M. Ebrahimi, T. Pahikkala, J. Plosila, and H. Tenhunen, "Gefes: A generalized wrapper feature selection approach for optimizing classification performance," *Computers in biology and medicine*, vol. 125, p. 103974, 2020.

[120] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18–28, 1998.

[121] W. S. Noble, "What is a support vector machine?" *Nature biotechnology*, vol. 24, no. 12, pp. 1565–1567, 2006.

[122] E. Byvatov and G. Schneider, "Support vector machine applications in bioinformatics." *Applied bioinformatics*, vol. 2, no. 2, pp. 67–77, 2003.

[123] C. Wang and C. Liang, "Msipred: a python package for tumor microsatellite instability classification from tumor mutation annotation data using a support vector machine," *Scientific Reports*, vol. 8, no. 1, p. 17546, 2018.

[124] L. Krause, A. C. McHardy, T. W. Nattkemper, A. Pühler, J. Stoye, and F. Meyer, "Gismo—gene identification using a support vector machine for orf classification," *Nucleic acids research*, vol. 35, no. 2, pp. 540–549, 2007.

[125] H.-F. Li, X.-F. Wang, and H. Tang, "Predicting bacteriophage enzymes and hydrolases by using combined features," *Frontiers in Bioengineering and Biotechnology*, vol. 8, p. 183, 2020.

[126] J. Adamo, "Fuzzy decision trees," *Fuzzy sets and systems*, vol. 4, no. 3, pp. 207–219, 1980.

[127] K. McNair, B. A. Bailey, and R. A. Edwards, "Phacts, a computational approach to classifying the lifestyle of phages," *Bioinformatics*, vol. 28, no. 5, pp. 614–618, 2012.

[128] Y. Zhang and Z. Li, "Rf_phage virion: Classification of phage virion proteins with a random forest model," *Frontiers in Genetics*, vol. 13, p. 1103783, 2023.

[129] T. Zheng, J. Li, Y. Ni, K. Kang, M.-A. Misiakou, L. Imamovic, B. K. Chow, A. A. Rode, P. Bytzer, M. Sommer *et al.*, "Mining, analyzing, and integrating viral signals from metagenomic data," *Microbiome*, vol. 7, pp. 1–15, 2019.

[130] O. B. Poirion, Z. Jing, K. Chaudhary, S. Huang, and L. X. Garmire, "Deepprog: an ensemble of deep-learning and machine-learning models for prognosis prediction using multi-omics data," *Genome medicine*, vol. 13, no. 1, pp. 1–15, 2021.

[131] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5–32, 2001.

[132] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[133] Y. Tian, J. Yang, M. Lan, and T. Zou, "Construction and analysis of a joint diagnosis model of random forest and artificial neural network for heart failure," *Aging (Albany NY)*, vol. 12, no. 24, p. 26221, 2020.

[134] S. Basu, K. Kumbier, J. B. Brown, and B. Yu, "Iterative random forests to discover predictive and stable high-order interactions," *Proceedings of the National Academy of Sciences*, vol. 115, no. 8, pp. 1943–1948, 2018.

[135] Z. Zhao, W. Yang, Y. Zhai, Y. Liang, and Y. Zhao, "Identify dna-binding proteins through the extreme gradient boosting algorithm," *Frontiers in Genetics*, vol. 12, p. 821996, 2022.

[136] N. Q. K. Le, Q. H. Kha, V. H. Nguyen, Y.-C. Chen, S.-J. Cheng, and C.-Y. Chen, "Machine learning-based radiomics signatures for egfr and kras mutations prediction in non-small-cell lung cancer," *International journal of molecular sciences*, vol. 22, no. 17, p. 9254, 2021.

[137] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[138] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Frontiers in neurorobotics*, vol. 7, p. 21, 2013.

[139] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," *arXiv preprint arXiv:1404.2188*, 2014.

[140] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[141] D. D. Wang, L. Ou-Yang, H. Xie, M. Zhu, and H. Yan, "Predicting the impacts of mutations on protein-ligand binding affinity based on molecular dynamics simulations and machine learning methods," *Computational and structural biotechnology journal*, vol. 18, pp. 439–454, 2020.

[142] J. Zhou, C. Y. Park, C. L. Theesfeld, A. K. Wong, Y. Yuan, C. Scheckel, J. J. Fak, J. Funk, K. Yao, Y. Tajima *et al.*, "Whole-genome deep-learning analysis identifies contribution of noncoding mutations to autism risk," *Nature genetics*, vol. 51, no. 6, pp. 973–980, 2019.

[143] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[144] M. K. Das and H.-K. Dai, "A survey of dna motif finding algorithms," *BMC bioinformatics*, vol. 8, no. 7, pp. 1–13, 2007.

[145] Q. Yu, X. Zhang, Y. Hu, S. Chen, and L. Yang, "A method for predicting dna motif length based on deep learning," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 20, no. 1, pp. 61–73, 2022.

[146] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko *et al.*, "Highly accurate protein structure prediction with alphafold," *Nature*, vol. 596, no. 7873, pp. 583–589, 2021.

[147] R. Poplin, P.-C. Chang, D. Alexander, S. Schwartz, T. Colthurst, A. Ku, D. Newburger, J. Dijamco, N. Nguyen, P. T. Afshar *et al.*, "A universal snp and small-indel variant caller using deep neural networks," *Nature biotechnology*, vol. 36, no. 10, pp. 983–987, 2018.

[148] M. U. Ahsan, Q. Liu, L. Fang, and K. Wang, "Nanocaller for accurate detection of snps and indels in difficult-to-map regions from long-read sequencing by haplotype-aware deep neural networks," *Genome biology*, vol. 22, pp. 1–33, 2021.

[149] A. Zell, *Simulation neuronaler netze*. Addison-Wesley Bonn, 1994, vol. 1, no. 5.3.

[150] Y. Chen, Y. Li, R. Narayan, A. Subramanian, and X. Xie, "Gene expression inference with deep learning," *Bioinformatics*, vol. 32, no. 12, pp. 1832–1839, 2016.

[151] S. Indolia, A. K. Goswami, S. P. Mishra, and P. Asopa, "Conceptual understanding of convolutional neural network-a deep learning approach," *Procedia computer science*, vol. 132, pp. 679–688, 2018.

[152] B. J. Wythoff, "Backpropagation neural networks: a tutorial," *Chemometrics and Intelligent Laboratory Systems*, vol. 18, no. 2, pp. 115–155, 1993.

[153] L. R. Medsker and L. Jain, "Recurrent neural networks," *Design and Applications*, vol. 5, no. 64-67, p. 2, 2001.

[154] E. Chalupová, O. Vaculík, J. Poláček, F. Jozefov, T. Majtner, and P. Alexiou, "Enngene: an easy neural network model building tool for genomics," *BMC genomics*, vol. 23, no. 1, pp. 1–11, 2022.

[155] N. Beknazarov and M. Poptsova, "Deepz: A deep learning approach for z-dna prediction," in *Z-DNA: Methods and Protocols*. Springer, 2023, pp. 217–226.

[156] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[157] T. Thireou and M. Reczko, "Bidirectional long short-term memory networks for predicting the subcellular localization of eukaryotic proteins," *IEEE/ACM transactions on computational biology and bioinformatics*, vol. 4, no. 3, pp. 441–446, 2007.

[158] D. Quang and X. Xie, "Danq: a hybrid convolutional and recurrent deep neural network for quantifying the function of dna sequences," *Nucleic acids research*, vol. 44, no. 11, pp. e107–e107, 2016.

[159] R. Trivedi and H. A. Nagarajaram, "Substitution scoring matrices for proteins-an overview," *Protein Science*, vol. 29, no. 11, pp. 2150–2163, 2020.

[160] S. Henikoff and J. G. Henikoff, "Amino acid substitution matrices from protein blocks." *Proceedings of the National Academy of Sciences*, vol. 89, no. 22, pp. 10 915–10 919, 1992. [Online]. Available: https://www.pnas.org/doi/abs/10.1073/pnas.89.22.10915

[161] J. Li and W. Wang, "Detailed assessment of homology detection using different substitution matrices," *Chinese Science Bulletin*, vol. 51, pp. 1538–1545, 2006.

[162] S. F. Altschul, J. C. Wootton, E. M. Gertz, R. Agarwala, A. Morgulis, A. A. Schäffer, and Y.-K. Yu, "Protein database searches using compositionally adjusted substitution matrices," *The FEBS Journal*, vol. 272, no. 20, pp. 5101–5109, 2005. [Online]. Available: https://febs.onlinelibrary.wiley.com/doi/abs/10.1111/j.1742-4658.2005.04945.x

[163] R. C. Edgar, “Muscle: Multiple sequence alignment with high accuracy and high throughput,” *Nucleic Acids Research*, vol. 32, no. 5, p. 1792–1797, 2004.

[164] K. Katoh, K. Misawa, K.-i. Kuma, and T. Miyata, “Mafft: a novel method for rapid multiple sequence alignment based on fast fourier transform,” *Nucleic acids research*, vol. 30, no. 14, pp. 3059–3066, 2002.

[165] H. Long, M. Li, and H. Fu, “Determination of optimal parameters of mafft program based on balibase3.0 database,” *SpringerPlus*, vol. 5, no. 1, 2016.

[166] N. Farheen, N. Sen, S. Nair, K. P. Tan, and M. Madhusudhan, “Depth dependent amino acid substitution matrices and their use in predicting deleterious mutations,” *Progress in Biophysics and Molecular Biology*, vol. 128, p. 14–23, 2017.

[167] M. Dayhoff, R. Schwartz, and B. Orcutt, “22 a model of evolutionary change in proteins,” *Atlas of protein sequence and structure*, vol. 5, pp. 345–352, 1978.

[168] T. Müller, R. Spang, and M. Vingron, “Estimating amino acid substitution models: a comparison of dayhoff’s estimator, the resolvent approach and a maximum likelihood method,” *Molecular biology and evolution*, vol. 19, no. 1, pp. 8–13, 2002.

[169] V. Polyanovsky, A. Lifanov, N. Esipova, and V. Tumanyan, “The ranging of amino acids substitution matrices of various types in accordance with the alignment accuracy criterion,” *BMC bioinformatics*, vol. 21, no. 11, pp. 1–25, 2020.

[170] R. C. Edgar, “Optimizing substitution matrix choice and gap parameters for sequence alignment,” *Bmc Bioinformatics*, vol. 10, pp. 1–11, 2009.

[171] F. Keul, M. Hess, M. Goesele, and K. Hamacher, “Pfasum: a substitution matrix from pfam structural alignments,” *BMC bioinformatics*, vol. 18, pp. 1–14, 2017.

[172] G. H. Gonnet, M. A. Cohen, and S. A. Benner, “Exhaustive matching of the entire protein sequence database,” *Science*, vol. 256, no. 5062, pp. 1443–1445, 1992.

[173] A. Prlić, F. S. Domingues, and M. J. Sippl, “Structure-derived substitution matrices for alignment of distantly related sequences,” *Protein Engineering*, vol. 13, no. 8, pp. 545–550, 2000.

[174] “The galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2022 update,” *Nucleic Acids Research*, vol. 50, no. W1, pp. W345–W351, 2022.

[175] D. Blankenberg, G. Von Kuster, E. Bouvier, D. Baker, E. Afgan, N. Stoler, G. Team, J. Taylor, and A. Nekrutenko, "Dissemination of scientific software with galaxy toolshed," *Genome biology*, vol. 15, pp. 1–3, 2014.

[176] D. L. Wheeler, T. Barrett, D. A. Benson, S. H. Bryant, K. Canese, V. Chetvernin, D. M. Church, M. DiCuccio, R. Edgar, S. Federhen *et al.*, "Database resources of the national center for biotechnology information," *Nucleic acids research*, vol. 36, no. suppl_1, pp. D13–D21, 2007.

[177] "Uniprot: the universal protein knowledgebase in 2021," *Nucleic acids research*, vol. 49, no. D1, pp. D480–D489, 2021.

[178] A. Marchler-Bauer, J. B. Anderson, P. F. Cherukuri, C. DeWeese-Scott, L. Y. Geer, M. Gwadz, S. He, D. I. Hurwitz, J. D. Jackson, Z. Ke *et al.*, "Cdd: a conserved domain database for protein classification," *Nucleic acids research*, vol. 33, no. suppl_1, pp. D192–D196, 2005.

[179] J. Armstrong, I. T. Fiddes, M. Diekhans, and B. Paten, "Whole-genome alignment and comparative annotation," *Annual review of animal biosciences*, vol. 7, pp. 41–64, 2019.

[180] D. Arndt, J. R. Grant, A. Marcu, T. Sajed, A. Pon, Y. Liang, and D. S. Wishart, "Phaster: a better, faster version of the phast phage search tool," *Nucleic acids research*, vol. 44, no. W1, pp. W16–W21, 2016.

[181] D. Santoro, L. Pellegrina, M. Comin, and F. Vandin, "Spriss: approximating frequent k-mers by sampling reads, and applications," *Bioinformatics*, vol. 38, no. 13, pp. 3343–3350, 2022.

[182] H. Gunasekaran, K. Ramalakshmi, A. Rex Macedo Arokiaraj, S. Deepa Kanmani, C. Venkatesan, and C. Suresh Gnana Dhas, "Analysis of dna sequence classification using cnn and hybrid models," *Computational and Mathematical Methods in Medicine*, vol. 2021, 2021.

[183] H. Yan, A. Bombarely, and S. Li, "Deepte: a computational method for de novo classification of transposons with convolutional neural network," *Bioinformatics*, vol. 36, no. 15, pp. 4269–4275, 2020.

[184] T. ValizadehAslani, Z. Zhao, B. A. Sokhansanj, and G. L. Rosen, "Amino acid k-mer feature extraction for quantitative antimicrobial resistance (amr) prediction by machine learning and model interpretation for biological insights," *Biology*, vol. 9, no. 11, p. 365, 2020.

[185] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[186] F. Chollet *et al.* (2015) Keras. [Online]. Available: https://github.com/fchollet/keras

[187] M. Larralde, "Pyrodigal: Python bindings and interface to Prodigal, an efficient method for gene prediction in prokaryotes," *Journal of Open Source Software*, vol. 7, no. 72, p. 4296, Apr. 2022. [Online]. Available: https://joss.theoj.org/papers/10.21105/joss.04296

[188] D. Hyatt, G.-L. Chen, P. F. LoCascio, M. L. Land, F. W. Larimer, and L. J. Hauser, "Prodigal: prokaryotic gene recognition and translation initiation site identification," *BMC Bioinformatics*, vol. 11, no. 1, p. 119, Dec. 2010. [Online]. Available: https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-11-119

[189] W. R. Pearson, "Selecting the right similarity-scoring matrix," *Current protocols in bioinformatics*, vol. 43, no. 1, pp. 3–5, 2013.

[190] C. Chrysostomou and H. Seker, "Novel protein weight matrix generated from amino acid indices," in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2015, pp. 8181–8184.

[191] D. W. Mount, "Using blosum in sequence alignments," *Cold Spring Harbor Protocols*, vol. 2008, no. 6, pp. pdb–top39, 2008.

[192] J. S. Andrade-Martínez, L. C. Camelo Valera, L. A. Chica Cardenas, L. Forero-Junco, G. López-Leal, J. L. Moreno-Gallego, G. Rangel-Pineros, and A. Reyes, "Computational tools for the analysis of uncultivated phage genomes," *Microbiology and Molecular Biology Reviews*, vol. 86, no. 2, pp. e00 004–21, 2022.

[193] L. Pradier, T. Tissot, A.-S. Fiston-Lavier, and S. Bedhomme, "Plasforest: a homology-based random forest classifier for plasmid detection in genomic datasets," *BMC bioinformatics*, vol. 22, no. 1, pp. 1–17, 2021.

[194] V. Pestov, "Is the k-nn classifier in high dimensions affected by the curse of dimensionality?" *Computers & Mathematics with Applications*, vol. 65, no. 10, pp. 1427–1437, 2013.

[195] M.-A. Schulz, B. T. Yeo, J. T. Vogelstein, J. Mourao-Miranada, J. N. Kather, K. Kording, B. Richards, and D. Bzdok, "Different scaling of linear models and deep learning in ukbiobank brain images versus machine-learning datasets," *Nature communications*, vol. 11, no. 1, p. 4238, 2020.

[196] A. Sarica, A. Cerasa, and A. Quattrone, "Random forest algorithm for the classification of neuroimaging data in alzheimer's disease: a systematic review," *Frontiers in aging neuroscience*, vol. 9, p. 329, 2017.

[197] M. A. Helaly, S. Rady, and M. M. Aref, "Convolutional neural networks for biological sequence taxonomic classification: A comparative study," in *Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2019*. Springer, 2020, pp. 523–533.

[198] S. Subramanian, J. A. Dover, K. N. Parent, and S. M. Doore, "Host range expansion of shigella phage sf6 evolves through point mutations in the tailspike," *Journal of Virology*, vol. 96, no. 16, pp. e00 929–22, 2022.

[199] M. Boon, D. Holtappels, C. Lood, V. van Noort, and R. Lavigne, "Host range expansion of pseudomonas virus luz7 is driven by a conserved tail fiber mutation," *Phage*, vol. 1, no. 2, pp. 87–90, 2020.

[200] M. Duplessis, C. M. Lévesque, and S. Moineau, "Characterization of streptococcus thermophilus host range phage mutants," *Applied and environmental microbiology*, vol. 72, no. 4, pp. 3036–3041, 2006.

[201] E. Capriotti and P. Fariselli, "Phd-snpg: a webserver and lightweight tool for scoring single nucleotide variants," *Nucleic acids research*, vol. 45, no. W1, pp. W247–W252, 2017.