Universidade do Minho
Escola de Engenharia

José Alexandre Graça Duarte

PhageDPO: Phage Depolymerase Finder

December 2021

PhageDPO: Phage Depolymerase Finder
José Alexandre Graça Duarte

UMinho | 2021

Universidade do Minho
Escola de Engenharia

José Alexandre Graça Duarte

PhageDPO: Phage Depolymerase Finder

Master Dissertation
Master's Degree in Bioinformatics

Dissertation supervised by
Doctor Hugo Alexandre Mendes de Oliveira
Doctor Óscar Manuel Lima Dias

December 2021

ACKNOWLEDGEMENTS

Começo por agradecer aos meus orientadores, Óscar Dias e Hugo Oliveira, por toda a ajuda, sugestões e correcções e por estarem sempre disponívies ao longo destes meses.

Aos meus colegas de mestrado, pelos bons momentos vividos nestes dois anos. Um agradecimento muito especial a minha namorada Inês Eulálio pelo permanente incentivo e preocupação com que sempre acompanhou este meu trabalho. Agradeço ainda a paciência e amor demonstrados nos meus momentos menos bons.

Por fim, quero agradecer a toda a minha família, especialmente aos meus pais e as minhas avós, por todo o apoio incondicional, pela paciência e por todos os sacrifícios que fizeram.

Este estudo contou com o apoio da Fundação para a Ciência e Tecnologia (FCT) portuguesa no âmbito do projeto PhageSTEC PTDC/CVT-CVT/29628/2017 [POCI-01-0145-FEDER-029628]

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.

RESUMO

A resistência aos antibióticos é um sério problema de saúde pública. Novos mecanismos de resistência estão a aparecer e a espalhar-se por todo o mundo, ameaçando a nossa capacidade de tratar infeções. Os bacteriófagos (fagos) surgem como uma solução pela sua capacidade de infeção e lise de bactérias. Os fagos são predadores naturais de bactérias: codificam um arsenal de proteínas especializadas para infeção dos seus hospedeiros. Uma proteína emergente é a depolimerase de polissacarídeos (DPOs) dos fagos, responsável pelo reconhecimento seletivo e degradação dos polissacarídeos presentes na superfície das bactérias, tornando-a suscetível a agentes externos. Devido à sua difícil localização no genoma do fago, foi desenvolvida a ferramenta PhageDPO, para previsão de DPOs, através de métodos de aprendizagem máquina.

Vários modelos foram desenvolvidos, com diferentes conjuntos de dados, e testados através de validação cruzada. Os conjuntos de dados são constituídos por sequências protéicas retiradas da base de dados NCBI *protein* e por números diferentes de casos negativos. Dois modelos foram incorporados na ferramenta: o modelo SVM treinado com dados de 4311 sequências e o modelo ANN treinado com dados de 7185 sequências. Num conjunto independente de dados de validação, o modelo SVM apresentou 95% de exatidão, 98% de precisão e 91% de sensibilidade e o modelo ANN apresentou 98% de exatidão, 99% de precisão e 96% de sensibilidade. Enquanto que a elevada exatidão e precisão do modelo SVM se foca na previsão de sequências corretamente classificadas, o modelo ANN assegura que todas as DPOs são identificadas devido a sua elevada sensibilidade. A PhageDPO foi testada com sucesso na previsão de DPOs de fagos previamente caracterizados.

PhageDPO foi integrado no Galaxy (https://bit.ly/3dOam2u), uma *framework* com interface gráfica para investigadores sem conhecimento de programação.

Palavras-Chave: Aprendizagem máquina, Bacteriófagos, Depolimerase, Galaxy

ABSTRACT

Antibiotic resistance is a severe public health problem. New resistance mechanisms are rapidly emerging and spreading globally, threatening our ability to treat infections. The bacteriophages (phages) arise as a possible solution through their capability of infecting and killing bacteria. Phages are natural bacterial predators: they encode an arsenal of specialized proteins to target their bacterial hosts. One emerging protein is Phages Depolymerases (DPOs), responsible for selective recognition and degradation of bacterial cell surface decorating polysaccharides, turning the bacteria susceptible to external agents. Due to the difficulty in locating these enzymes in the phage genome, we developed PhageDPO, a DPO prediction tool, through machine learning methods.

Several classifiers were created, using different datasets and algorithms and tested through cross-validation. The datasets were composed of protein sequences retrieved from the NCBI protein database and by a different number of negative cases. Two models were selected for integration in the tool: the Support Vector Machine (SVM) model created with a dataset containing data of 4311 sequences and the Artificial Neural Network (ANN) model created with a dataset containing data of 7185 sequences. On an independent validation dataset, the SVM model presented 95% accuracy, 98% precision and 91% recall and the ANN model presented 98% accuracy, 99% precision and 96% recall. While the high precision and PECC of the SVM focus on predicting true DPO sequences and avoiding false positives, the ANN ensures that all DPOs are identified due to its high recall. PhageDPO was successfully tested in predicting DPOs of, previously characterized, phages.

PhageDPO was integrated into the Galaxy framework (https://bit.ly/3dOam2u), providing a user-friendly graphical interface for wet-lab researchers without computational skills.

Keywords: Bacteriophages, Depolymerase, Galaxy, Machine Learning

CONTENTS

ACRONYMS

AAC Amino Acid Composition. 33
AI Artificial Intelligence. 2, 15
ANN Artificial Neural Network. vi, xii–xvi, 22, 23, 38, 44, 46, 49–71, 97, 102
AP Average Precision. 19, 38
AUC Area Under the Curve. 19

CDD Conserved Domains Database. 29–31, 41
CM Confusion Matrix. 16, 38, 46, 70
CPS Capsular Polysaccharide. 7, 9, 10, 12
CTD Composition Transition Distribution. 34, 36
CV Cross-Validation. 21, 38

DPC Dipeptide composition. 14, 36
DPO Depolymerase. vi, xi–xvii, 1–3, 9–15, 20, 30–32, 36, 42–44, 50, 58, 62–71, 87, 92, 97, 102, 107, 109, 112, 114, 117, 118, 121
DT Decision Trees. xii, xiii, 22, 24, 25, 38, 44, 46, 49–51, 123, 124

EPS Extracellular Polymeric Substance. 7, 9

FN False Negative. xii, 16, 18, 46, 50, 56–58, 64, 70
FP False Positive. xii, 16, 18, 46, 49, 50, 52, 56–58, 62, 64, 65, 70, 71

JSON JavaScript Object Notation. 42

kNN k-Nearest Neighbors. 22, 23, 37, 38, 44, 45

LOOCV Leave-one-out cross-validation. 21
LOS Lipooligosaccharide. 8
LPS Lipopolysaccharide. 7–10, 12
LR Logistic Regression. 25

MAD Mean of Absolute Deviation. 20
ML Machine Learning. 2, 14–16, 19, 21, 22, 25, 37–39, 70

NB Naïve Bayes. 22, 23, 38, 44, 46
NC Nucleotide Composition. 33
NGS Next Generation Sequencing. 1
NPV Negative Predictive Value. 16

OM Outer Membrane. 8
ORF Open Reading Frame. 2, 13, 43

PDB Protein Data Bank. 27, 29
PECC Percentage of Examples Correctly Classified. xiv, 16, 38, 44–47, 53, 54, 58, 65, 70
PG Peptidoglycan. 7
PIR Protein Information Resource. 27
PPV Positive Predictive Value. 16
PR Precision-Recall. xii, xiii, 18, 19, 38, 50–52, 55, 58–61, 71, 124–126
PRF Protein Research Foundation. 27
PVP Phage Virion Protein. 14

RAST Rapid Annotations using Subsystems Technology. 13, 14
RBP Receptor Binding Protein. 6, 9, 10, 14
RF Random Forest. xii–xv, 22, 24, 25, 38, 44, 46, 49–61, 70, 125, 126
RFE Recursive Feature Elimination. 38, 52
RMSE Root Mean Square Error. 20
ROC Receiver Operator Characteristics. xii, xiii, 19, 38, 50, 51, 55, 58–61, 123–125

SSE Sum of the Square error. 20
SVM Support Vector Machine. vi, xii–xvi, 22, 37, 38, 44, 46, 49–71, 97, 102, 124, 125

TN True Negative. 16, 51
TNR True Negative Rate. 16
TP True Positive. 16, 52, 58
TPA Third Party Annotation. 27, 28
TSP Tail Spike Protein. 6, 10, 11, 14

XML Extensible Markup Language. 40

LIST OF FIGURES

LIST OF TABLES

1

INTRODUCTION

1.1 MOTIVATION

Over the past decades, genome sequencing technologies have taken a giant leap forward, which led to a decreased operational cost and increased number and diversity of sequenced genomes. The extraordinary complexity of the revealed genome architecture led sequencing technologies to even more significant advancements. Producing millions of sequences while processing multiple DNA sequences in parallel, these high-throughput technologies, known as Next Generation Sequencing (NGS), are now a routine part of biological research [1].

Discovered more than one hundred years ago by William Twort and Félix d'Herelle, bacteriophages, or phages, are viruses that exclusively infect and replicate within bacteria [2]. However, more than a century later, the number of sequenced phages genomes has increased exponentially due to their therapeutic potential against antibiotic-resistant bacteria. Phages and phage derivatives, such as Depolymerase (DPO), have great potential as antibacterial or antivirulence agents for bacterial infections, thus representing an alternative therapy to fight these multidrug-resistant bacterial infections [3]. For the isolation of DPOs, the lab procedure is laborious and a time-consuming. This procedure requires: a) Isolation of novel phages from the environment able to form phage plaques with hazy rings phenotypes in the drop tests (the hallmark for the detection of phages carrying DPO); b) Extraction and sequencing of their genomic DNA; c) Amplification and

cloning of the putative gene encoding DPO that is difficult to detect within all phage-encoded proteins; d) Confirmation of the DPO activity by spotting tests.

Therefore, a sequence-based computational method is required to hasten these laborious tasks. However, there are many phage proteins of unknown function, ”hypothetical proteins”, in databases due to the increasing gap between predicted phage gene sequences and their functions. Thus, sequence similarity seems insufficient for identifying DPOs [4]. Therefore, the solution for this problem may rely upon Artificial Intelligence (AI). Despite challenging, identifying these proteins may lead to a superior comprehension of the interaction between phage-bacteria and the development of novel antibacterial applications [5].

This dissertation presents the development of a bioinformatics tool (PhageDPO), which predicts the existence of DPOs in a given phage relying on Machine Learning (ML) methods. This tool will also be capable of returning the corresponding Open Reading Frame (ORF) and the predicted probability of that ORF being a DPO.

1.2 Objectives

This project's main goal is to create an online tool that allows scientists from phage community to identify and locate DPOs. In detail, the objectives are to:

- review ML approaches and identify algorithms used to predict proteins in phages and other organisms;
- develop a software program using Python™ that will identify DPOs in a phage genome;
- deploy the tool on a web-based platform, making it user-friendly for scientists. The long-term objective is to create a suite of online bioinformatics tools.

Since the number of sequenced phage genomes is exponentially growing, the development of this framework is of the utmost importance. Ideally, and considering the large

genomic information available, which has been increasing exponentially, DPO can be in theory identified in phage genomes already deposited in the public databases, which could then be directly synthesized and used, shortening significantly the time needed to find and explore these proteins.

2

STATE OF THE ART

2.1 BACTERIOPHAGES

Bacteriophages, or phages, are viruses that exclusively infect and replicate within bacteria. Being the most abundant and diverse biological entities on earth, phages exist as part of a complex microbial ecosystem distributed in locations populated by bacterial hosts, such as aquatic environments, deserts, polar regions and even intestines of animals [6]. Many phages are known to infect or lyse bacteria of different taxa, as well as, directly influencing the evolution of their hosts genomes by carrying genes from one host to another in a process of transduction [7]. Unlike the broad spectrum of the antibiotics, most phages possess a narrow host range, sometimes restricted to a particular strain. Thus, an effective application of phage therapy requires prior knowledge of the infecting strain [8].

Phages genetic material may be single-stranded or double-stranded DNA or RNA (ssDNA, dsDNA, ssRNA, dsRNA), and varies widely from thousands to hundreds of thousands of basepairs (bp) [9]. Phages are classified according to the virion's morphological and its genomic content. According the International Committee on Taxonomy of Viruses (ICTV), phage morphology can be divided in tailed, polyhedral, filamentous or pleomorphic [10]. Phage classification is based on the follow three criteria: Type of nucleic acid (ssDNA, dsDNA, ssRNA, dsRNA); Shape of the capsid (tubular or icosahedral); Presence or absence of envelope. Tailed phages, representing 96% of all phages, belong to the order *Caudovirales*. With linear dsDNA and non-enveloped, this order

include fourteen families, from which three families are common and characterised by long straight contractile, long flexible noncontractile, and short non-contractile tails and named *Myoviridae*, *Siphoviridae*, and *Podoviridae*, respectively [11, 12].

Phages can also be categorised in terms of their infection strategies. They can either have lytic or lysogenic life cycles [2]. Lytic phages (virulent phages) infect their hosts and very quickly begin replication. When sufficient numbers of progeny are produced, the host cells lyse, killing it in the process, as depicted in Figure 1, adapted from [13]. Lysogenic phages (temperate phages) undergo very little replication in their host. During infection, they incorporate their genome, called ”prophage”, into that of their bacterial host. Phage’s genome is maintained as a plasmid-like form, designated ”episome”, where it remains and is passively replicated along with the host.

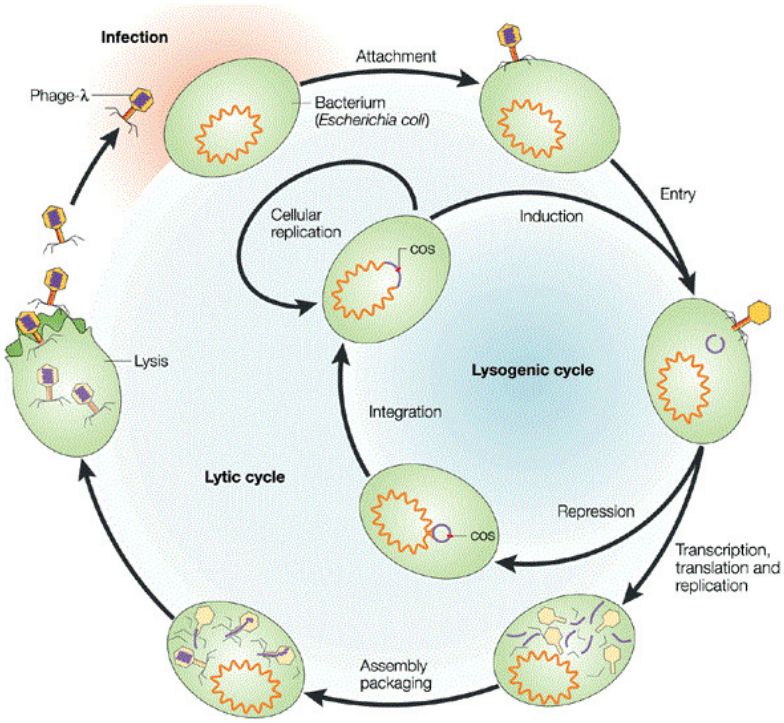


Figure 1: Lytic and lysogenic life cycles of phages.

This life cycle occurs when a host encounters unfavourable growth conditions, preserving the phage genome until an appropriate environment appears. However, lysogenic phages are able to become lytic with exposure to certain environmental stimuli such as an-

tibiotics and host inflammation [2, 14, 15]. While temperate phages can transfer virulence and resistance genes, virulent phages are preferably chosen for phage therapy since they destroy the host [16]. The conformational triggering that leads to phage infection is influenced by the interaction between proteins of the phages' adsorption apparatus and the conservative receptor structures of the bacterial cell surface. The adsorption kinetics of this highly specific interaction is one of phages's key aspects, determining whether the host strain will be sensitive to the respective phage and how efficiently the phage can control the strain population [17, 18]. Phage Receptor Binding Proteins (RBPs), are the key factors that determine specificity. The hosts are recognised through the binding of RBP to a specific receptor on the cell surface [19]. During adsorption, the phage initially binds reversibly to the bacterial cell surface. Such initial or reversible binding occurs through the interaction between phage Tail Spike Proteins (TSPs) or tail fibres and cell primary receptors. Following reversible binding, phages are committed upon irreversible binding with a secondary receptor, signaling the virion to release its genetic material into the bacterial cell [20]. RBP-encoding genes recognition based solely on sequence homology is often hopeless even when comparing with RBPs already characterised. This is due to the high diversity of phage-host interactions, where RBP was evolved to be structurally similar but with distinct primary sequences [19, 21]. Phages are able to target specific surface-accessible receptors distributed in a genus-specific, species-specific or even strain-specific manner. Thus, phages and phage-derived enzymes constitute an important and promising alternative to control bacterial pathogens. Moreover, these enzymes are also be suitable against intracellular pathogens, where phages have difficulties to reach due to the lack of receptors for eukaryotic cells. Phage-derived enzymes can be easily delivered into specific infection sites, acting locally in the infection and reducing side effects [22, 23].

2.2 Polysaccharide barriers between phages-bacteria interactions

Bacterial polysaccharides present in the cell wall are important structures that block the entry of antimicrobials (Figure 2, adapted from [24]), but also function to avoid phage predation [25]. In Gram-negative bacteria cell wall is composed of thin Peptidoglycan (PG) layers surrounded by an outer membrane containing a Lipopolysaccharide (LPS) and several proteins. As for Gram-positive bacteria, cells lack outer membranes, having instead a much thicker PG layer framed with diverse proteins and cell wall teichoic acids. Often, Gram-negative and positive cells also display an outermost Capsular Polysaccharide (CPS) [26]. The capsule is considered a virulence factor because it increases the ability of bacteria to cause disease and is involved in biofilm production. While in these communities, cells are protected from external factors, being halted together in close proximity by Extracellular Polymeric Substances (EPSs). EPS allow cell-cell communication and horizontal gene transfer events [18, 27]. In Gram-negative bacteria, CPS are connected to the outer membrane via a lipid anchor. CPS have different designations according to the bacterial species in which they are present. In *Escherichia coli*, *Klebsiella pneumoniae* and *Acinetobacter spp.* they are termed K-antigens, while in *Salmonella spp.* and *Citrobacter spp.* CPS are named Vi antigens [28]. CPS are usually larger than other polysaccharides with variations in monosaccharide units, glycosidic bonds and non-carbohydrates substitutions to lead to a wide range of chemical types of CPS. For example, *P. aeruginosa* only have one capsular type described, while *E. coli* and *K. pneumoniae* have 80 or more different antigens described so far [29]. Besides Gram-negative bacteria, Gram-positive species also have a huge diversity of CPS. While *Bacillus anthracis* only has one capsular type, *Streptococcus pneumoniae* has a total of 98 different capsule serotypes [30]. The molecules composing the capsules are also different, while, for example, in *S. aureus* CPS are mainly composed by poly-N-acetylglucosamine, *Streptococcus pyogenes* produces a hyaluronic acid capsule [28].

The LPS is vital to the majority of Gram-negative bacteria providing stability and protection to the cell. Three structural domains can be identified in the LPS molecule: lipid A, core oligosaccharide, and the O antigen [31]. Lipid A is an acylated β-1'-6-linked glucosamine disaccharide that forms the outer side of the Outer Membrane (OM). The core is a non-repeating oligosaccharide that is linked to the glucosamines of lipid A. The hypervariable O-antigen is attached to the core oligosaccharide and is composed of a repeating oligosaccharide of two to eight sugars. The structure of the O antigen varies at the strain level of a species and is sometimes absence in Gram-negative bacteria. For the latter, molecules composed of only lipid A and the core oligosaccharide, are denominated Lipooligosaccharide (LOS). LOS may be "rough" LPS, or "smooth" LPS, if they don't include or include the O antigen, respectively [32, 33].

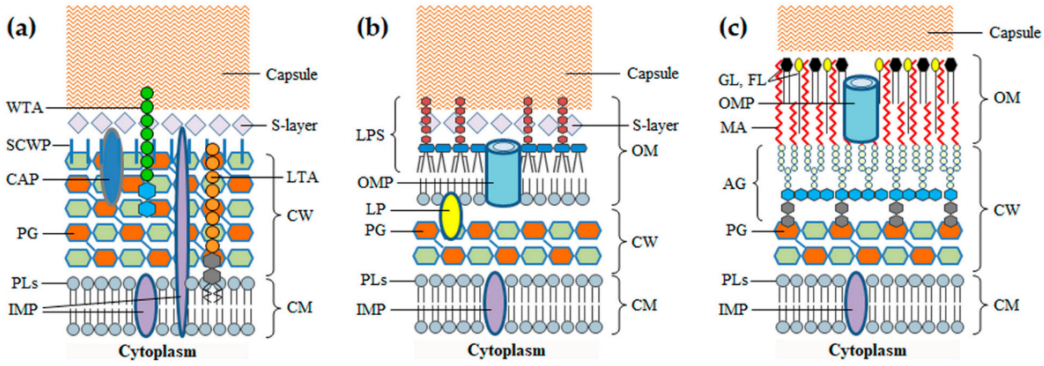


Figure 2: Depiction of the structure and composition of the bacterial cell envelope in Gram-positive bacteria (a), Gram-negative bacteria (b) and mycobacteria (c). CM, cytoplasmic membrane; CW, cell wall; OM, outer membrane; IMP, inner membrane proteins; PLs, phospholipids; AG, arabinogalactan; PG, peptidoglycan; LP, lipoprotein; LTA, lipoteichoic acids; CAP, covalently attached protein; SCWP, secondary cell wall polymers; WTA, wall teichoic acids; OMP, outer membrane protein; LPS, lipopolysaccharide; MA, mycolic acids; GL, glycolipids; FL, free lipids. The S-layer and capsule are extracellular structures. Branched lipoaraninomannan is not represented in the mycobacterial cell envelope (probably anchored to both the CM and OM).

All these referred layers pose a barrier against phage predation. Consequently, phages have developed many mechanisms to overcome them. Focus will be given to the

phage structural enzyme, DPO, capable of degrading these important above-mentioned carbohydrate barriers.

2.3 Polysaccharide Depolymerases

Tailed phages adsorb onto the host polysaccharide based-receptors, inject their DNA, replicate and lyse the cells to release their progeny. As mentioned before, bacterial cell surface decorating polysaccharides, CPS, EPS or LPS, also exhibit important functions in biofilm production, virulence, and with phage interaction. Therefore, tailed phages evolved to encode enzymes, such as DPOs, to recognise and degrade these external polymers. Phage-encoded DPOs, are located at the phage RBPs. This interaction between RBPs and polysaccharide-based receptors grants an irreversible bind to the bacterial host cell, allowing phage to initiate the infection [28, 34]. The role of these capsule polymers, as primary receptor, is vital for phage infection. Interestingly, in the continuous phage-host arms race, bacteria evolved to display a multitude of different LPS and CPS structures to shield themselves from phage predation. But phages have co-evolved to recognize these polymers as host receptors. Moreover, phages that bind to LPS/CPS also ultimately dependent on them. Experiments have shown that LPS/CPS-dependent phages drastically reduce adsorption onto cells previously treated with cognate recombinant LPS or CPS depolymerases, i.e. stripped from these phage receptor carbohydrate polymers [35, 36].

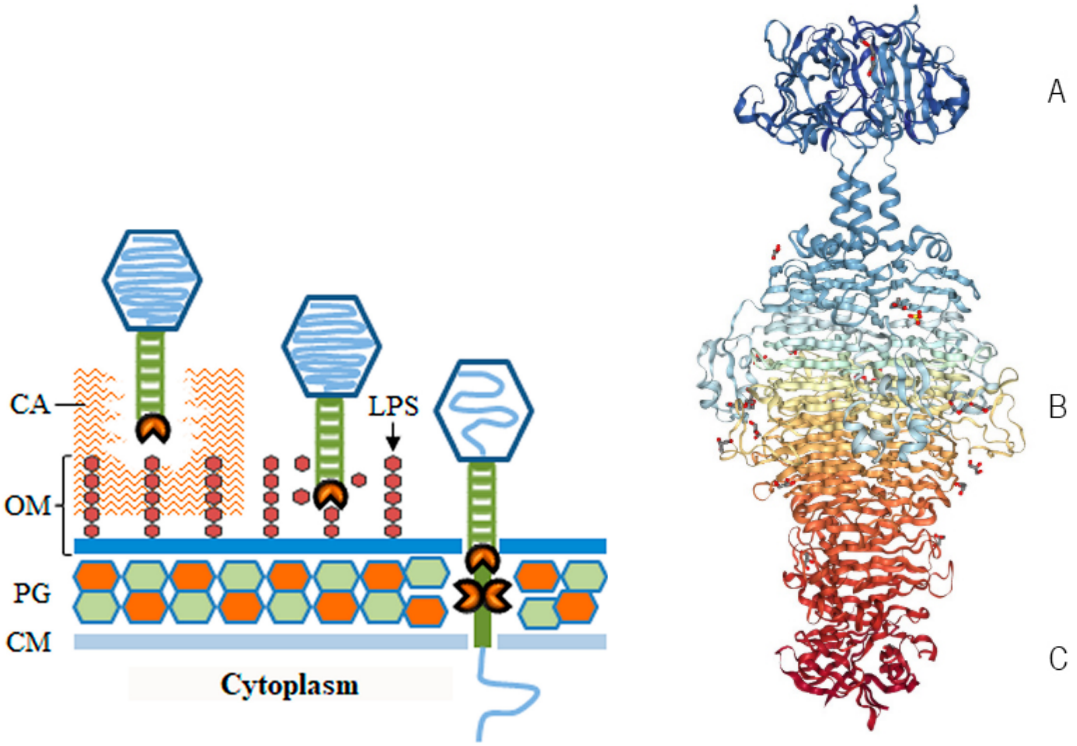
The presence of phage plaques surrounded by hazy rings that usually grow over time, has been the hallmark for the detection of phages carrying DPOs indicating the LPS/CPS degrading activity [37]. DPOs are known to strip off the cell's protective polysaccharides layer, often decreasing bacterial virulence and exposing it to environmental factors such as the immune system or antibiotics [38]. Therefore, DPO have a huge potential to be used as a novel anti-virulence to control bacterial pathogens, repeatedly demonstrated in several invertebrate and vertebrate in vivo models [28]. Moreover, the diversity of DPO

is vast. Several experiments demonstrated that recombinantly produced depolymerases are capable of degrading, for instance, the CPS of *E. coli* [39], *K. pneumoniae* [40] and *A. baumannii* [35, 36] hosts, and LPS O antigens of *P. aeruginosa* [41] and *Salmonella enterica* [42].

2.3.1 Location and structure

Despite several reports indicating the presence of DPO at the phage tail fibers, the current data available suggests that they are predominantly found at the phage tail spikes (e.g. *Acinetobacter phage vB_ApiP_P1*, *Pseudomonas phage LKA1*, *Escherichia phage K1F*) [35, 39, 43]. Tail Spike Protein (TSP) are shorter and carry domains that are assumed to have enzymatic activity, unlike tail fibers [44]. However, some DPOs from Gram-Positive phages have been identified at the baseplate (*Lactobacillus phage Ld17*) [45], and at the pre-neck proteins (*Bacillus phage phi29* and *Staphylococcus phage vB_SepiS-IPLA7*) [46, 47]. While most phages encode one single DPO, others encode more, to increase their host range. The *Escherichia phage K1-5*, for instance, encode both a K1 and a K5 depolymerase allowing the phage to infect K1 and K5 *E.coli* strains, respectively [48]. The *Klebsiella phage K64-1* is an extreme example encoding nine experimental validated DPOs to target specific *Klebsiella* CPS antigens (K1, K11, K21, K25, K30/K69, K35, K64, KN4, and KN5) [49].

Phage DPOs typically fold as trimers [50], with only one case reported as a tetramer [51]. A typical architecture of a phage RBP with DPO activity often consists of three domains, as can be seen in figure 3(b) adapted from [52]: an N-terminal dome-like structure domain, a central β-helical domain for host recognition and enzymatic activity (i.e depolymerase), and a C-terminal domain responsible for protein trimerisation [50, 53]. Possible DPO action is depicted in Figure 3(a), adapted from [24].



(a) (b)

Figure 3: (a) - Possible DPO action during recognition and penetration of the bacterial cell envelope (Gram-negative bacteria used as example). Depolymerase activity is generically depicted as a pacman symbol. CM, cytoplasmic membrane; PG, cell wall peptidoglycan; OM, outer membrane; LPS, lipopolysaccharide; CA, capsule.
(b) - Tail spike of *Salmonella phage P22*, illustrating a typical modular structure of RBPs. A) N-terminal domain. B) β-helical domain. C) C-terminal domain.

While the N-terminal and C-terminal domains of the TSP are conserved among phages belonging to the same group, the central domain is highly variable and can be changed to modulate the host range [50]. These three elongated and right-handed monomers with β-helical side-by-side topology seems to favor stability, enzymatic degrading activity and resistance to high temperatures and denaturing salts [36]. All this contributes to the robustness of these proteins, which evolved to endure demanding external conditions to maintain the phage infectivity.

2.3.2 Enzymatic activity

Phage DPOs can be generally classified according to their mechanism of action, as hydrolases or lyases. Both result in the cleavage of polysaccharides resulting in the breakdown of the carbohydrate barrier.

Hydrolases cleave glycosidic bonds by consuming a water molecule and include sialidases, levanases, xylanase, dextranases, rhamnosidases, glycanases and peptidases [54]. Sialidases, or endo-N-acetylneuraminidases, are primarily found in phages that degrade the α-linkage of polysialic acid. Some Gram-negative bacteria, including *Escherichia coli K1*, have α-2,8-linked polysialic acid as a CPS. Phages known to have this DPO domain include the *Escherichia phage K1E* and *K1F* [55], phage 63D [51] and phage ϕ92 [56]. Rhamnosidases, frequently found in phages infecting *Salmonella*, cleave the α-1,3 O-glycosidic bond between L-rhamnose and D-galactose present in Gram-negative LPS O-antigens [57]. Levanase, present in *Bacillus phage SP10*, cleaves the β-2,6-bond in levan. Levan is an important component in *Bacillus* biofilm and has been suggested to be part of *Pseudomonas* capsule, protecting it against phages [58]. Xylanase, responsible for the hydrolysis of the β-1,4 bonds within xylan, identified in the *Caulobacter phage Cr30*, while dextranase cleaving the α-1,6-linkages between glucose units in dextran is predicted in *Lactobacillus phage ϕPYB5* [50]. There are, among hydrolases, enzymes that cleave peptide bonds known as peptidases. *B. subtilis* phage *ϕNIT1* that produces a γ-PGA hydrolase PghP, is known to have a DPO with peptidase domain [59].

Lyases are a class of enzymes that cleave (1,4) glycosidic bonds by β-elimination mechanism. In this class of enzymes, we can find three groups of depolymerases: hyaluronate, alginate, and pectin/pectate lyases. Pectin/pectate lyases degrade CPS of *Klebsiella* [49] and *Acinetobacter* [35, 36], as well as enzymes that degrade the LPS O-antigen from *Pseudomonas* [43]. Finally, hyaluronate and alginate lyases are a less explored class of enzymes. Hyaluronidases, cleaving the β-1,4 bonds of the subunits of

hyaluronic acid, were found in prophages invading *Streptococcus pyogenes* and *S. equi*, which are both encapsulated by hyaluronic acid [60]. Alginate lyases are characteristic of *Pseudomonas* and *Azobacter* phages. Able to degrade the α-1,4 bond of alginate, a linear polysaccharide of β-D-mannuronate, and its C5 epimer α-L-guluronate common for mucoid strains infecting cystic fibrosis patients [61].

Enzymatic activity shows us that phage-encoded DPOs are diverse proteins not yet fully studied.

2.4 Predicting DPO based on genomic data

De novo assembly allows reconstructing a genome from many (short or long) DNA fragments (reads), with no previous knowledge of those fragments' correct sequence or order. Raw reads generated by sequencers are generally stored in FastQ files. A set of overlapping oriented reads is called *contig*. A single contig is constructed from two or more overlapping and oriented reads. The construction of two or more joined and oriented contigs is called a scaffold. The contigs may be overlapping or non-overlapping [62].

There are several tools developed with the purpose of genome assembly, such as SPAdes [63], Velvet [64] and CLC genomics workbench [65]. After the assembly of the genome, the next step is gene identification that can be performed with available bioinformatics programs, such as GeneMark [66], GLIMMER [67] and Prodigal [68]. Predicted gene products are further functional annotated by similarity searches against different protein databases, using several tools such as Blastp [69], HHpred [70] and InterProScan [71]. The Rapid Annotations using Subsystems Technology (RAST) is an automated service for identifying of protein-encoding genes, assessing gene functions, and for metabolic reconstruction [72]. By default, the RAST's pipeline uses GLIMMER to identify the ORFs, but other algorithms such as GeneMark [66] and Prodigal [68] can be used. RAST uses a combination of homology, chromosomal clustering, and subsystems

to assess proteins' functions. First, proteins are annotated based on homology to known proteins. If this initial search yields matches to proteins that are a component of a subsystem, RAST seeks other subsystem members that should be present in the same genome based on information from the previously annotated genomes [73].

Phage proteins can be classified into two classes: phage virion (structural) proteins and phage non-virion (non-structural) proteins. Phage Virion Proteins (PVPs) are mostly involved in determining the bacterial host receptors, such as DPOs. Phage non-virion proteins (non-PVPs) are not wrapped in phage virions but encoded by the viral genome. These proteins execute crucial functions in biological processes like transcription, viral genome replication and cell lysis. Identification of PVP may be an essential step for the identification of DPOs. The ML approach for identifying PVPs comprises:

- data collection;
- applying sequence-based feature descriptors;
- combining features and selecting the optimal using feature selection algorithms;
- feeding the optimal feature to a classifying model to generate final identifiers.

Performance comparison of PVP identifiers demonstrated that the g-gap Dipeptide composition (DPC) feature is a relevant biomarker for PVP classification. G-gap DPC calculates the frequency of two residues with g intervals [74].

According to Latka et al. (2019) [52] phage genomes encoding RBPs with putative DPO activity are identified through the analysis of their annotated tail fibers or TSPs with tools, such as BlastP [69], Phyre2 [75], SWISS-MODEL [76], HMMER [77] and HHPred [70]. The absence of tail fiber and tail spike genes lead to the analysis of all genes located in the proximity of structural annotated genes. Despite being longer than 200 residues and annotated as tail fiber/tail spike/hypothetical protein, the proteins must show homology to previously described domains, already known to be associated with DPO activity with a confidence higher than 40% in Phyre2 or the enzymatic domain recognized by at least SWISS-MODEL, HMMER, or BlastP. Homologies with DPO domains should hold at least

a 100 residues interval, and Phyre2 should predict a typical β-helical structure. Proteins with experimentally conformed DPO activity and proteins that partially fulfilled the above criteria were marked differently [52].

2.5 Supervised Machine Learning

2.5.1 Concepts and definitions

Machine Learning (ML) is a field of AI aimed at interpreting data and creating models for data prediction and classification.The success of a learning algorithm depends on the data; thus, ML is related to data analysis and statistics [78].

A Classification problem is a process in which the algorithm groups data based on predetermined characteristics. The data is organised in a tabular schema called a dataset, where each row represents an instance, and each column represents an attribute. The last column represents the output attribute, while the others represent the input attributes. An instance, or object, is an observation of the data described by features from which a model will learn. An attribute is a feature describing an instance. It can be continuous or categorical. A continuous attribute is a numeric attribute with infinite values between any two given values, such as distance or weight. Categorical attributes contain a finite number of categories or distinct groups and can also be divided into nominal and ordinal. In general approach, the instances are split into two sets: a) training set, with the data used to build the model; and b) test-set, to test the models's effectiveness. The model built by the algorithm is a predictive function representing what an ML system has learned from the training data's input. In supervised learning, the output is already known; thus, the data used to build the model already possess the output value for each instance. The objective is to use this function to map new instances, and, therefore, the learning algorithm must be capable of generalising from the training data to new cases. Supervised learning problems are mainly used for classification tasks when the output

variable is discrete, such as ”Positive” or ”Negative”, and regression tasks, when the output variable is continuous, such as ”time” [79, 80].

Besides supervised, ML algorithms encompass unsupervised and reinforced learning. In unsupervised learning, training data does not contain the output variable, and the goal is to model the data distribution. This type of problem can be the cluster analysis, where the algorithm manages to find a structure or pattern in a collection of uncategorised data, and association analysis, where the algorithm tries to discover relationships between variables. In reinforced learning, the ML algorithm performs suitable actions to maximise the reward in a particular situation. With no output data, the algorithm is bound to learn from its own experience [80].

2.5.2 Metrics and model evaluation

A Confusion Matrix (CM) can be generated for classification problems, consisting of a 2 X 2 matrix, representing a problem with two classes: Negative and Positive (Table 1), to evaluate a model's quality. For more than two classes, a CM is calculated for each class. In this matrix, rows represent the actual values, while columns represent the predicted values. The model's negative and positive instances correctly predicted are designated True Negative (TN) and True Positive (TP), respectively. On the other hand, if the model misclassifies a Negative instance for a Positive, it is considered an FP, and inversely if the model misclassifies a Positive instance for a Negative, it is considered an FN [81].

Based on the CM, several metrics can be calculated. For instance, the accuracy of the model, also known as PECC (equation 1), is the number of correct predictions to the total number of input samples. The Positive Predictive Value (PPV), also known as Precision, determines how many correctly predicted cases actually turned out to be positive (equation 2). Negative Predictive Value (NPV) is the proportion of true negatives (equation 3). Whereas sensitivity, or Recall, indicates the actual positive cases correctly predicted with our model (equation 4). Specificity, or True Negative Rate (TNR), measures

Table 1: Example of a Confusion Matrix of a 2-class problem and how the metric relates with each classification case

Real \ Predict	**Negative**	**Positive**	
Negative	True Negative (TN)	False Positive (FP) Type I Error	**Specificity** $\frac{TN}{FP+TN}$
Positive	False Negative (FN) Type II Error	True Positive (TP)	**Sensitivity** $\frac{TP}{TP+FN}$
	Negative Predictive Value $\frac{TN}{TN+FN}$	**Precision** $\frac{TP}{TP+FP}$	**Accuracy** $\frac{TP+TF}{TP+TF+FP+FN}$

the proportion of correctly identified negative cases (equation 5). Finally, the F1 Score measures a model's accuracy that considers both Recall and Precision (equation 6) [82].

$$Accuracy = \frac{TP + TF}{TP + TF + FP + FN} \tag{1}$$

$$PPV = Precision = \frac{TP}{TP + FP} \tag{2}$$

$$NPV = \frac{TN}{TN + FN} \tag{3}$$

$$Sensitivity = Recall = \frac{TP}{TP + FN} \tag{4}$$

$$TNR = Specificity = \frac{TN}{FP + TN} \tag{5}$$

$$F1Score = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * TP}{2 * TP + FP + FN} \tag{6}$$

Metrics are selected based on the problem being assessed. Imbalanced datasets are prevalent in a multitude of fields, and biology is no exception. In a binary classification with an imbalanced dataset, even a weak model that predicts majority class for all data instances may provide 95%+ accuracy, but that does not mean the model is working well. Accuracy should always go together with other metrics such as Precision and Recall. However, having a model with very high Precision means an FP value near zero. If the model is tweaked to get an FP count of zero, it may return a high FN count. The recall is essential in medical cases, but not only. For example, in the binary classification problem of COVID-19 Prediction, FN (Person has COVID-19 but model predicted Not having COVID-19) is more critical than FP (Person Not having COVID-19 but model predicted having COVID-19). So, in this case, Recall becomes crucial. Increasing Precision may reduce Recall, and increasing Recall may reduce Precision. This PR trade-off can be an essential tool when precision is more important than Recall or vice versa [83].

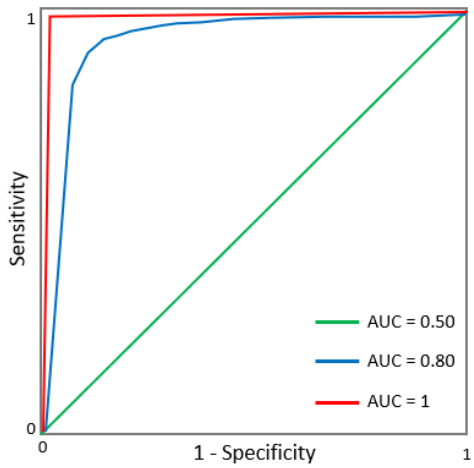


Figure 4: Representation of the ROC curves of 3 models. The red curve represents a model with perfect separation between 2 classes (AUC = 1). The blue curve represents a reasonably accurate model (AUC = 0.8). The green curve represents a model predicting randomly (AUC = 0.5)

The ROC is another important evaluation tool of the performance of Binary Classification. This tool relates sensitivity (y-axis) with the false positive rate (1-Specificity) (x-axis), demonstrating how well the model performs in separating two classes (Figure 4).

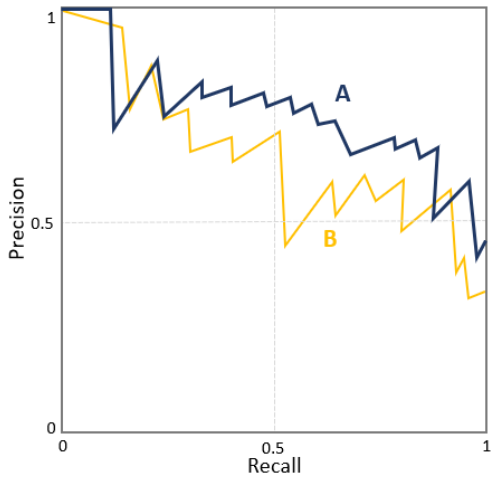


Figure 5: Precision-recall curves representing the performance of models A and B. In this example the performance of model A is superior to the performance of model B.

The model's accuracy can be measured by the Area Under the Curve (AUC). Curves closer to the upper left corner comprise a larger area, therefore higher accuracy, while diagonal curves with an area of 0.5 represent arbitrary predictions. However, when classes are imbalanced, generally with more instances for the negative class, PR curves are more suitable than ROC curves to evaluate the model's performance. Often zigzag frequently going up and down, PR curves plot the trade-off between precision (y-axis) and Recall (x-axis), discussed previously, for different thresholds (Figure 5).

Models with excellent performance are represented by a curve towards the coordinate of (1,1), top right corner. Average Precision (AP) is calculated as the area under a curve that measures the trade off between precision and recall at different thresholds. As for the ROC curves, a high area under the curve represents a model that returns high Precision and High recall [84].

A Density Plot visualises the distribution of data over a continuous interval or time period and is a useful way to visualize the distribution of the scores produced by a ML

model predictions. It uses the kernel density estimation (KDE) to estimate the probability density function (PDF) of a variable. The peaks of a Density Plot over an interval represent high concentration (high density) of values in that interval. The x-axis is the value of the variable and the y-axis is the probability density which is the probability per unit on the x-axis [85, 86]. For a model predicting the percentage of a certain condition to occur (ex: the probability of a protein being a DPO, ranging from 0% to 100%), the density of the predictions, for a dataset composed exclusively by positive cases, should be located near the left area of the plot with a rapid decrease towards zero (Figure 6).

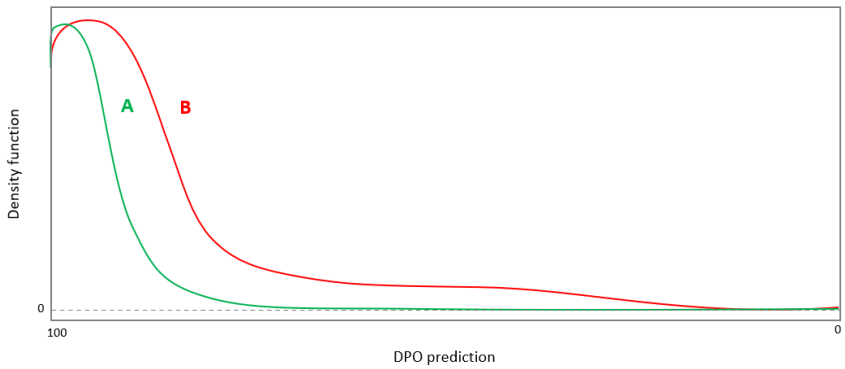


Figure 6: Density plot for the distribution of DPO predictions in a positive validation dataset, for models A and B. Model A outperforms model B since all its predicted values are located in the area of high percentage (100%) and with a rapid decrease towards zero. Model B predicted more values in a wider range of percentages indicating more false negative predictions.

For regression problems, the difference between the predicted value ($\hat{y}$) and the real value (y) is calculated by error metrics. Sum of the Square error (SSE) (equation 7), measures the variance of the predicted value from the real value of the data. Generally, a lower SSE value indicates that the regression model can better explain the data, while a higher SSE value indicates that the model poorly explains the data. Mostly used as a measure of variation within a cluster; Root Mean Square Error (RMSE) (equation 8), consists of the square root of SSE divided by the number of instances; and Mean of Absolute Deviation (MAD) (equation 9). The model with the lowest value is considered the most accurate [82].

$$SSE = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{7}$$

$$RMSE = \sqrt{(\frac{1}{n})\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{8}$$

$$MAD = (\frac{1}{n})\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{9}$$

As mentioned before, the dataset is initially divided into two sets, training and validation with the training set larger than the validation set. The resampling method of Cross-Validation (CV) uses all available data and is primarily used to measure the models's skill on unseen data. The method is repeated k times. Each time, one of the k subsets is used as test/validation set while the other k-1 subsets form the training set. The metrics used are averaged over all *k* trials to get the total performance of our model. The usage of all the data means significantly reduced bias. For imbalanced data, the method Stratified K Fold may return better results. This method aims to ensure that each class is equally represented across each test fold. Leave-one-out cross-validation (LOOCV) is a configuration of k-fold cross-validation where *k* is equal to the number of examples in the dataset [82, 87].

An important consideration in ML is how well the model generalises to new data. Generalisation is important because the collected data is a sample only. A good ML model must generalise well from the training data to any data from the problem domain. When a model learns the data too well, either by picking up noise or random fluctuations characteristic of the training data, it becomes unable to generalise to new data and overfitting occurs. Larger training sets and preventing the creation of overcomplex models are the best ways to reduce overfitting. Conversely, underfitting refers to a model that neither models the training data nor generalises new data. Underfitting is mostly associated with poor data quality, where the model is unable to detect any

trends. Overfitting and underfitting are the two leading causes for poor performance of ML algorithms [88].

2.5.3 Machine Learning algorithms

The suitable ML algorithm depends on several factors, including but not limited to data size, quality and diversity, and which conclusions we want to derive from that data. ML algorithms such as Support Vector Machine (SVM), Artificial Neural Network (ANN), k-Nearest Neighbors (kNN), Naïve Bayes (NB), Decision Trees (DT), Random Forest (RF) and Linear and logistic regression will be discussed next [89].

SVMs are mainly used for classification and regression problems, though more often for binary classification. SVMs separate the dataset into two classes by a hyperplane, maximising the margins between both, as depicted in Figure 7 [90]. Support vectors are data points closer to the hyperplane, influencing its position and orientation. Using these support vectors, the margin of the classifier is maximised. The Gaussian kernel is generally used when there is no prior knowledge about the data. SVM algorithms are advantageous when dealing with large amounts of disparate information and when the discriminant function uses only a small subset of the training set, making the computations significantly faster. However, SVMs have a slow training process [91].

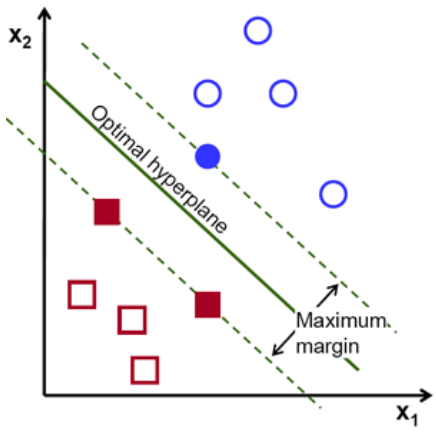


Figure 7: Example of the separation of two classes by the SVM hyperplane. The dashed lines are the support vectors.

ANNs are biologically inspired computational networks. ANNs possess an input layer and output layer, and in between, other hidden layers that perform the mathematical computations that help determine the decision or action. The basic unit of ANN is the artificial neuron, depicted in Figure 8. Each input is associated with some weight that will be added to the sum. ”Bias” is used to adjust the output along with the weighted sum of the inputs to the neuron. The sum is then filtered by an activation function, resulting in the output signal. The appropriate number of hidden layers and their sizes vary with the problem. The advantages of ANNs are their flexibility and robustness in capturing imprecise and incomplete data patterns. However, ANN’ outputs are influenced by factors like number of cases and the number of training cycles [92, 93].

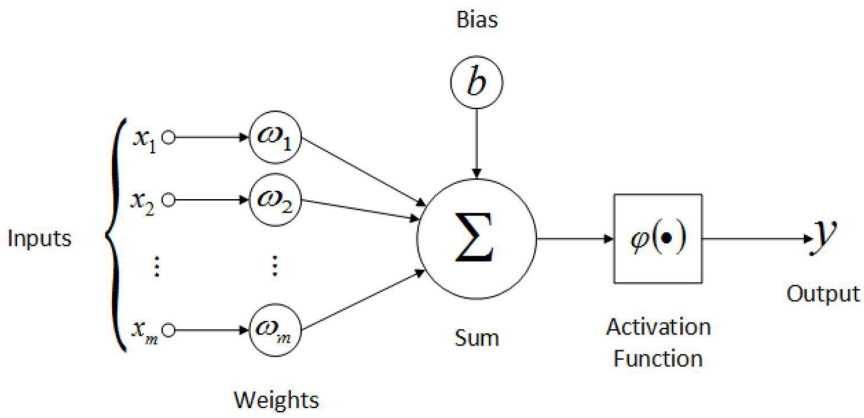


Figure 8: Structure of an artificial neuron.

kNNs is a learning method based on the assumption that similar things exist nearby. In classification problems, the algorithm calculates the k training examples most similar to the example to be classified, using the input features, and the predicted output will correspond to the most common output class in those k examples. The algorithm uses a similarity function that is usually the standard Euclidean distance [94].

NB is based on Bayes’ theorem for conditional probabilities, assuming that all input attributes have the same importance and occur independently. The algorithm calculates the frequency table for each attribute against the target. This table is used as a template

for the likelihood table, where the Naive Bayesian equation calculates the probability for each class. The predicted outcome belongs to the class with the highest probability [95].

DTs, also used for classification problems, generate models in a tree structure. Each node represents an input feature, and each branch that comes out from that node corresponds to a possible value of that feature. Several splits are made in the upcoming nodes and increasing numbers of branches are generated to partition the original data. This process stops on a node where all or almost all of the data belong to the same class and further splits or branches are no longer possible. Nodes with outgoing edges are the internal nodes, and all others are terminal nodes or leaves of the DT. Because the DTs algorithm is sensitive to variance, minor variations within the training set must be low; otherwise, the algorithm may generate a different tree.

RFs is an ensemble of decision tree algorithms and produces, even without hyper-parameter tuning, most times a good result. The training algorithm for RF applies the technique of bagging to tree learners which is an algorithm combines the predictions from many decision trees (Figure 9). Bagging is an effective ensemble algorithm because each DT is fit on a slightly different training dataset, resulting in a slightly different performance. In a classification problem, the prediction is the majority vote predicted across the decision trees [96, 97].

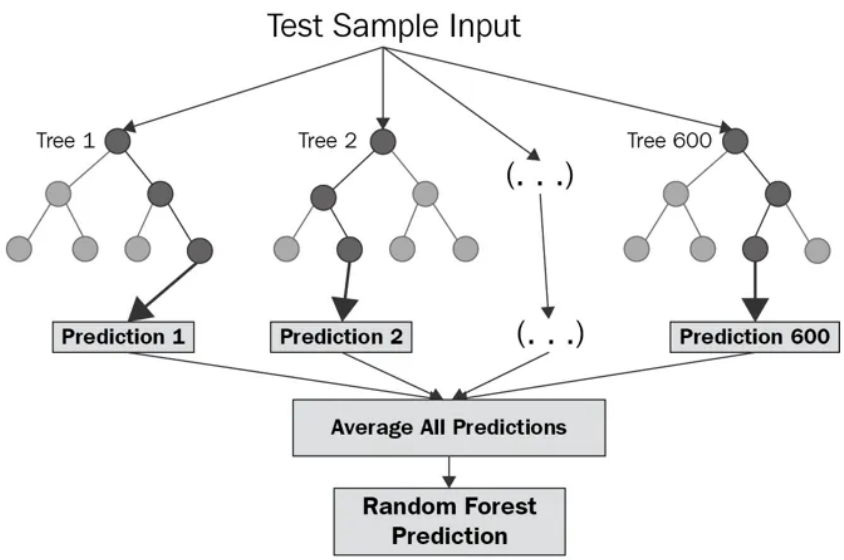


Figure 9: RF structure in a classification problem where the average prediction of the DTs is the prediction of the RF model.

Regression consists of modelling the relationship between iteratively refined variables through the error in the model's predictions. Logistic Regression (LR) models the linear relationship between a dependent variable (target) and an independent variable (predictors), through a linear function with parameters originated from the data. LR calculates a sigmoid function for estimating the probability of a binary output based on one or several independent variables. The main difference from linear regression is that the modelled output value is binary rather than numeric [98].

2.6 Development environments and tools

2.6.1 Scikit-learn Python Library

Scikit-learn (sklearn) is an open-source machine learning library. It is built upon NumPy, SciPy and Matplotib and contains several efficient ML and statistical modelling tools, including classification, regression, clustering, and dimensionality reduction. More additional information on sklearn is available in the documentation [99].

2.6.2 BIOPYTHON LIBRARY

Biopython is a Python™ library containing several modules and classes to treat and access biological data. This library allows the reading and writing of many file formats used in bioinformatics, such as FASTA and GenBank, and access to online services and databases, such as NCBI or UniProt. Biopython functionalities go from the automation of collecting biological information to tools such as BLAST and AlignIO. A complete description of all functionalities is available in Biopython Tutorial and Cookbook [100].

2.6.3 GALAXY

Galaxy is an open-source, web-based platform design not only for data-intensive biomedical research but also for biologists to analyse their own data. A researcher interacts with Galaxy through the web by uploading and analysing the data. Galaxy interacts with underlying computational infrastructure (servers that run the analyses and disks that store the data) without exposing it to the user [101]. It allows users with no programming experience to easily set parameters and run individual tools as well as larger workflows. Galaxy ensures reproducibility by capturing the information of each run so that any user can repeat and understand the complete computational analysis. Regarding Galaxy interface, users can upload their own data, choose tools, define inputs and specify parameters. Also, this platform enables researchers to share and publish their Galaxy objects such as: histories, which are computational analyses with specified input datasets and parameters as well as the output datasets; workflows, computational analyses that specify all the steps and parameters used, but none of the data, in order to run the same analysis on different sets of input data; datasets, which includes any input, intermediate, or output dataset, used or produced in an analysis; and pages, interactive, web-based documentation describing a complete analysis [102].

Galaxy consists of several components: Public Galaxy Server, an instance of the Galaxy software combined with many tools, visualizations and data sources; Galaxy software framework, an open source application; Galaxy Tool Shed, where developers tools are uploaded and available, as well as their configurations and guides for installation of required dependencies; and Galaxy Community, consisting in broad community of users, developers and administrators who maintain Galaxy instances [103].

2.6.4 Databases

The NCBI's Protein database is a collection of protein sequences from several sources, including translations from annotated coding regions in GenBank, Reference Sequence (RefSeq) and Third Party Annotation (TPA), as well as records from Swiss-Prot, Protein Information Resource (PIR), Protein Research Foundation (PRF), and Protein Data Bank (PDB).

Genbank is the National Institute of Health (NIH) genetic sequence database. This database comprises an annotated collection of all publicly available DNA sequences their protein translations. It is part of the International Nucleotide Sequence Database Collaboration, along with the DNA DataBank of Japan (DDBJ) and the European Molecular Biology Laboratory (EMBL). Nucleotide sequences are primarily obtained through submission from individual laboratories and batch submissions from large-scale sequencing projects, such as whole genome shotgun (WGS). GenBank consists of several divisions, most of which can be accessed through the Nucleotide database [104].

RefSeq is a collection of curated, non-redundant genomic DNA, RNA, and protein sequences. Providing stable reference for genome annotation, gene identification and characterization, mutation and polymorphism analysis and expression studies, this database is a synthesis of information integrated across multiple sources at a given time. All RefSeqs can be found in the Entrez Nucleotide or Protein databases and can be

accessed by adding ”*AND srcdb_refseq[property]*” to the query. Alternatively, an option is also provided in the results page to allow display only the RefSeq accessions [105].

TPA records are sequence annotations published by someone other than the original submitter of the primary sequence record in DDBJ/EMBL/GenBank. These records can be divided in three categories: experimental, data is supported by peer-reviewed wet-lab experimental evidence; inferential, data by inference and not been the subjected of direct experimentation; and reassembly, the objective is on providing a better assembly of the raw reads. TPA sequences are not released to the public until their accession numbers or sequence data and annotation appear in a peer-reviewed biological journal [104].

Universal Protein (UniProt) is a Consortium between the European Bioinformatics Institute (EMBL-EBI), the Protein Information Resource (PIR) and the Swiss Institute of Bioinformatics (SIB). This database is a repository of protein sequence and data annotation and is comprised of four major sectors optimized for different tasks: the UniProt Knowledgebase (UniProtKB), the UniProt Reference Clusters (UniRef) and the UniProt Archive (UniParc). The UniProtKB is a database partially curated and consists of two sections: UniProtKB/Swiss-Prot, containing manually annotated and reviewed entries; and UniProtKB/TrEMBL, containing automatically annotated unreviewed entries. UniRef comprises three databases of clustered sets, achieved with CD-HIT, of protein sequences from UniProtKB and UniParc records: the UniRef100 database, which groups identical sequences and sub-fragments into a single UniRef entry; UniRef90, built from UniRef100 by clustering its sequences at the 90% identity level; and UniRef50, built from UniRef90 by clustering its sequences at the 50% identity level. Clustering sequences reduces database size and increases the speed of similarity searches and detection of distant relationships [106, 107]. UniParc is the most comprehensive, non-redundant protein sequences database available, and contains only protein sequences, sequence versions and database cross-references. A protein sequence may exist in several different databases and with multiple copies in the same database. To overcome this, UniParc stores each unique protein sequence only once and assign an identifier, beginning in ”UPI” followed by 10 hexadecimal numbers, that is never removed or subject to reassign.

Proteins extracted from source databases are linked to their origins by using database cross-references, which links one protein in UniParc to an accession number in a source database. If a sequence is modified or removed, the cross-reference retires and the history of all changes is archived [106, 108].

PDB is a database of experimentally determined three-dimensional structure data of biological macromolecules. PDB structure submissions are restricted to atomic-level structures that have been determined by one or more of the supported experimental techniques: Macromolecular Crystallography (MX), Nuclear Magnetic Resonance spectroscopy (NMR), 3D Electron Microscopy (3DEM), powder diffraction and fiber diffraction. Each structure in PDB is identified with a four-character alphanumeric identifier [109].

NCBI's Conserved Domains Database (CDD) comprises a collection of sequence alignments and profiles representing protein conserved domains. Domains are distinct functional and/or structural units in a protein and may exist in multiple biological contexts. As a unit of molecular evolution used to establish evolutionary classifications, domains are usually associated to a particular protein function such as enzymatic activity, membrane transport, or nucleic-acid binding [110]. Proteins with different functions can have similar domains. The CDD in-house curated domain collection use the 3D structure to guide multiple sequence alignment (MSA) models, and are manually annotated with functional sites using the evidence from the 3D structure and published literature. CDD is integrated with several resources at the NCBI, including BLAST, Protein, and Gene, and can be accessed by using: CD-Search, for a single nucleotide or protein sequence query via sequence identifier or by pasting in the sequence in FASTA format; Batch CD-Search, for up to 4000 queries at a time; Standalone RPS-BLAST and rpsbproc, to compute and retrieve domain annotation programmatically [111]. For a specific domain entry, in the information block titled "Links", several options are available (Source, Taxonomy, Protein, and Superfamily) forwarding to the that domain associated proteins in the Protein database.

3

MATERIALS AND METHODS

3.1 PHAGEDPO WORKFLOW

For DPO prediction, one critical step is in the dataset compilation. The dataset contains true positive and true negative data. The true positive data contains phage genomes with DPOs and true negative data contains phage genomes lacking DPOs. Feature assembly and pre-processing prepare the data to be fed to the models. After feature selection, performance evaluation and model optimization, the best models are selected. The final task is to deploy the tool on Galaxy.

3.2 DATA COLLECTION

High-quality data is a fundamental factor when building a model that distinguishes between DPOs and non DPOs sequences. All the data obtained for this work was gathered in Aug-2021. To construct the positive dataset, 6 DPO associated domains within NCBI's CDD and their related proteins were obtained. These domains are described in Table 2.

Table 2: Domains associated with DPOs activity and number of related protein sequences obtained from the Conserved Domains Database. Domains Bibliography included.

Domain	Sequences obtained	Associated domain and protein
Pfam12708	517	Domain present in DPO YP_003347555 [112]
cd20481	287	Domain present in DPO ASN73504 [35]
Pfam12219	111	Domain present in DPO YP_338127 [39, 113]
cl22684	68	Domain present in DPO CBY99579 [114]
Pfam12217	75	Domain present in DPO YP_338127 [39, 113]
Pfam13472	693	Domain present in DPO ARB10970.1 [34]

DPO comprise a wide variety of proteins so, to depict that variety in the positive data. Thus, entries containing DPO and tail related words not present in the CDD were sought in the NCBI's protein database. The constraints for this query are the following:

```
viruses[porgn:__txid28883]
AND tail*
AND (*glycanase*[Text Word]
OR *alginate*[Text Word]
OR *rhamnosidase*[Text Word]
OR *hyaluronidase*[Text Word]
OR *hyaluronate*[Text Word]
OR *eps-degrading*[Text Word]
OR *levanase*[Text Word]
OR *dextranase*[Text Word]
OR *xylanase*[Text Word])
```

The query produced 677 more sequences, for a total of 2428 positive sequences. CD-HIT[115] was used to remove 100% identical sequences (redundant data), resulting in

1437 positive DPO sequences. The negative cases were obtained with the query on NCBI protein database:

```
(viruses[porgn:__txid28883]
AND srcdb_refseq[Properties])
NOT lyase*[Text Word]
NOT pectate*[Text Word]
NOT pectin*[Text Word]
NOT depolymerase*[Text Word]
NOT glycanase*[Text Word]
NOT endoglycosidase*[Text Word]
NOT alginate*[Text Word]
NOT rhamnosidase*[Text Word]
NOT hyaluronidase*[Text Word]
NOT hyaluronate*[Text Word]
NOT eps-degrading*[Text Word]
NOT levanase*[Text Word]
NOT dextranase*[Text Word]
NOT xylanase*[Text Word]
NOT pfam12708*[Text Word]
NOT cd20481*[Text Word]
NOT pfam12219*[Text Word]
NOT cl22684*[Text Word]
NOT sgnh_hydrolase*[Text Word]
NOT gdsl_hydrolase*[Text Word]
NOT pfam12217*[Text Word]
NOT pfam13472*[Text Word]
```

The negative case query was limited to the first 30.000 entries, whose Genbank records were obtained and curated through keyword check and the duplicated sequences removed with CD-HIT, resulting in 22.976 negative sequences.

3.3 FEATURES

Based on the sequence properties of both protein and DNA, for each sequence, 578 input features were created. Formulations of both sequences are indicated below by equations 10 and 11:

$$P = A_1 A_2 A_3 \ldots A_l \tag{10}$$

where A_i represents the ith amino acid in the protein P of length l.

$$DNA = N_1 N_2 N_3 \ldots N_l \tag{11}$$

where N_i represents the ith nucleotide in the DNA sequence of length l.
The features are as follows:

- (24 features) Amino Acid Composition (AAC) and Nucleotide Composition (NC).
  The AAC and NC are composed, respectively, of 20 and 4 vectors and calculated as:
  $$Comp_j = \sum_{i=1}^{l} \sigma_i \tag{12}$$
  where $Comp_j$ corresponds to the composition in j and:
  $\sigma_i = 1$ if the i occurrence is j-type.
  $\sigma_i = 0$ if the i occurrence is not j-type.
  For the length l of the sequence, j ranges from 1 to 20 in proteins and 1 to 4 in DNA sequences.
- (1 feature) Length of the amino acid sequence.
  The length l of P, composed of one vector.

- (5 features) Aromaticity, Isoelectric point and Secondary structure fraction.

  Depicted in equation 13, Aromaticity is the relative frequency of aromatic amino acids.

$$Aromaticity = \sum_{i=1}^{20} \gamma_i f_i \tag{13}$$

  where f_i is the relative frequency of *i* type amino acid in the protein and γ_i = 1 if the amino acid is Phe, Tyr or Trp, otherwise γ_i = 0. The Isoelectric point is the pH at which the net charge of the protein is zero. Both aromaticity and isoelectric point, are composed of one vector each. Secondary structure fraction calculates the fraction of amino acids that tend to be found in three secondary structures: α-helixes (equation 14), β-turns (equation 15) and sheets (equation 16).

$$helix = \sum_{i=1}^{20} \alpha_i f_i \tag{14}$$

$$turn = \sum_{i=1}^{20} \theta_i f_i \tag{15}$$

$$sheet = \sum_{i=1}^{20} \beta_i f_i \tag{16}$$

  Being f_i the relative frequency of *i* type amino acid in the protein:

  α_i = 1 if the amino acid is Val, Ile, Tyr, Phe, Trp or Leu in equation 14, otherwise α_i = 0.

  θ_i = 1 if the amino acid is Asn, Pro, Gly or Ser in equation 15, otherwise θ_i = 0.

  β_i = 1 if the amino acid is Glu, Met, Ala or Leu in equation 16, otherwise β_i = 0.

  Secondary structure fraction comprises, this way, 3 vectors.

- (147 features) Composition Transition Distribution (CTD).

  CTD consists in grouping the amino acids into three classes encoded by the indices 1, 2 and 3 according to which group they belong. Amino acid attributes

such as Hydrophobicity, Normalized van der Waals volume, Polarity, Polarizability, Charge, Secondary structure and Solvent Accessibility were used as properties. Composition (C) is the number of amino acids of a given property divided by the total amino acid number. Transition (T) characterizes the percent frequency with which amino acids of a particular property is followed by amino acids of a different property. Distribution (D) measures the chain length within which the first, 25, 50, 75 and 100% of the amino acids of a particular property is located respectively [116]. For a given property, the composition C is calculated through:

$$composition_j = \frac{1}{l}\sum_{i=1}^{l}\sigma_i \tag{17}$$

where j = 1,2,3 and $composition_j$ corresponds to the composition in j for the length l of the sequence, also:

$\sigma_i = 1$ if ith occurrence is equal to j.

$\sigma_i = 0$ if ith occurrence is not equal to j.

Transition from group 1 to 2 is the percentage frequency in which 1 is followed by 2 or 2 is followed by 1 in the encoded sequence:

$$transition_{mn} = \frac{D_{mn} + D_{nm}}{l - 1} \tag{18}$$

where mn = ”12”, ”13”, ”23” and D_{mn}, D_{nm} are the numbers of dipeptide encoded as ”mn” and ”nm”, respectively in the sequence.

Distribution of properties along the protein chain can be described with the help of equation 19.

$$R_{j,p} = \frac{P}{100}\sum_{i=1}^{l}\sigma_i \tag{19}$$

$R_{j,p}$ represents the number of residues for group j within a given percentage P for a protein of length l where:

P = 1, 25, 50, 75, 100 and j = 1, 2, 3.

$\sigma_i = 1$ if ith occurrence is equal to *j*.

$\sigma_i = 0$ if ith occurrence is not equal to *j*.

For example, 25% of all residues belonging to group 1 ($R_{1,25}$ = 4) are contained within the first 5 residues of the protein chain of length 25, distribution for that group is calculated (5/25)*100%, giving D1025 = 20%.

Each amino acid attribute produces 3, 3 and 15 vectors, since all 7 attributes are used, CTD produces 147 features.

- (400 features) Dipeptide composition (DPC)

  DPC is calculated according to equation 20. For any dipeptide D_j:

$$DPC_j = \frac{D_j}{l-1} \tag{20}$$

  where *j* = 1, 2,..., 400 and *l* is the length of the sequence. For each protein sequence, 400 features are produced.

The previous features were calculated through the modules SeqUtils.ProtParam from Biopython (version 1.78)[100] and Propy3 (version 1.0.0a2)[117]. The output feature in the training data, ”PDPO_Exists”, is binary and differentiates between a DPO (”1”) and a non-DPO (”0”) case.

3.4 DATASETS

The datasets created for this work are described in table 3.

Table 3: Created dataset dimensions, number of features, positive cases and negative cases.

Dataset Name	Features	Positives	Negatives	Total
d2874	579	1437	1437	2874
d4311	579	1437	2874	4311
d5748	579	1437	4311	5748
d5748R	579	1437	4311	5748
d7185	579	1437	5748	7185

Despite sharing the same positive cases and number of features (578 and 1 output feature), datasets d2874, d4311, d5748 and d7185 comprise different numbers of negative cases to test their influence on model performance. Despite having the same size, datasets d5748 and d5748R diverge in selected negative cases. Negative cases were chosen randomly from the total set of negative sequences; thus, d5748R was created to evaluate whether the selected negative cases impact on model performance.

3.5 PRE-PROCESSING

The most common data transformation is the center scaling of feature variables. ML algorithms that exploit distances or similarities in training data, such as kNN and SVM, tend to be sensitive to data scaling. The data standardisation was implemented using the function ”StandardScaler” from sklearn. This function centers a feature variable by subtracting the average feature value from all the values. In addition, to scale the data, each value of the feature variable is divided by its standard deviation. It was applied to all features of all datasets as all features are numerical with a wide range of values.

3.6 MODELS

Based on the previous datasets, several ML models were created, each one based on different ML algorithms:

- SVM using the ”SVC” function;
- ANN using the ”MLPClassifier” function;
- NB using the ”GaussianNB” function;
- DT using the ”DecisionTreeClassifier” function;
- kNN using the ”KNeighborsClassifier” function;
- RF using the ”RandomForestClassifier” function.

3.7 FEATURE SELECTION AND PERFORMANCE EVALUATION

Feature selection is the process of reducing the number of features. A reduced number of features reduces the computational cost of modelling and, in some cases, may help to improve the performance of the model. Recursive Feature Elimination (RFE) is a popular feature selection algorithm that follows the fit/transform pattern of sklearn. It works by removing features one at a time based on the weights given by a model, such as RF, in each iteration. The implementation of an RFE algorithm was achieved through the sklearn function ”RFECV” and the RF estimator. Cross-Validation (CV) is a resampling method that evaluates ML models on a limited data sample. CV with 5-fold was implemented together with metrics such as PECC, recall and precision to evaluate model performance. ROC and PR curves were also created along with AUROC values and AP values. Confusion Matrix (CM) were created to give an overall view on how well models are performing and what kind of errors their making.

3.8 MODEL OPTIMIZATION

ML models have hyperparameters, an external model configuration with a value that cannot be estimated from the training data. The approach used was objectively search different values for model hyperparameters and choose a subset that produces a model with the best performance. Table 4 describes the hyperparameters optimized and the range of their values. This selection was performed with sklearn's "GridSearchCV" function.

Table 4: Hyperparameters of each model and range of values tested.

Model	Parameter	Values GridSearch
ANN	'solver'	'adam', 'sgd'
	'activation'	'relu', 'tanh'
	'alpha'	0.0001, 0.001, 0.01
	'hidden_layer_sizes'	(10,), (15,), (25,), (50,), (100,)
SVM	'C'	0.1, 1, 10, 15, 20, 100
	'gamma'	'auto', 0.01, 0.001, 0.1, 1
	'kernel'	'linear', 'rbf', 'poly', 'sigmoid'
RF	'n_estimators'	100, 200
	'max_depth'	None
	'max_features'	'auto', 'sqrt'
	'min_samples_split'	2, 3, 6
	'min_samples_leaf'	2, 3, 6
	'bootstrap'	True, False
	'criterion'	'gini', 'entropy'

3.9 Galaxy Deployment

PhageDPO was deployed on Galaxy, with a provided a user-friendly interface. Using Planemo [118], an Extensible Markup Language (XML) file was generated, containing all the details for deploying the tool in Galaxy framework, namely: tool inputs, outputs and their formats, the Linux command line to run the script, an example of possible inputs and outputs and the dependencies, which corresponds to the python modules used by the model.

4

DEVELOPMENT

The code for this work was developed in Python 3.8, using PyCharm and constituted by 9 scripts, as depicted in Figure 10.

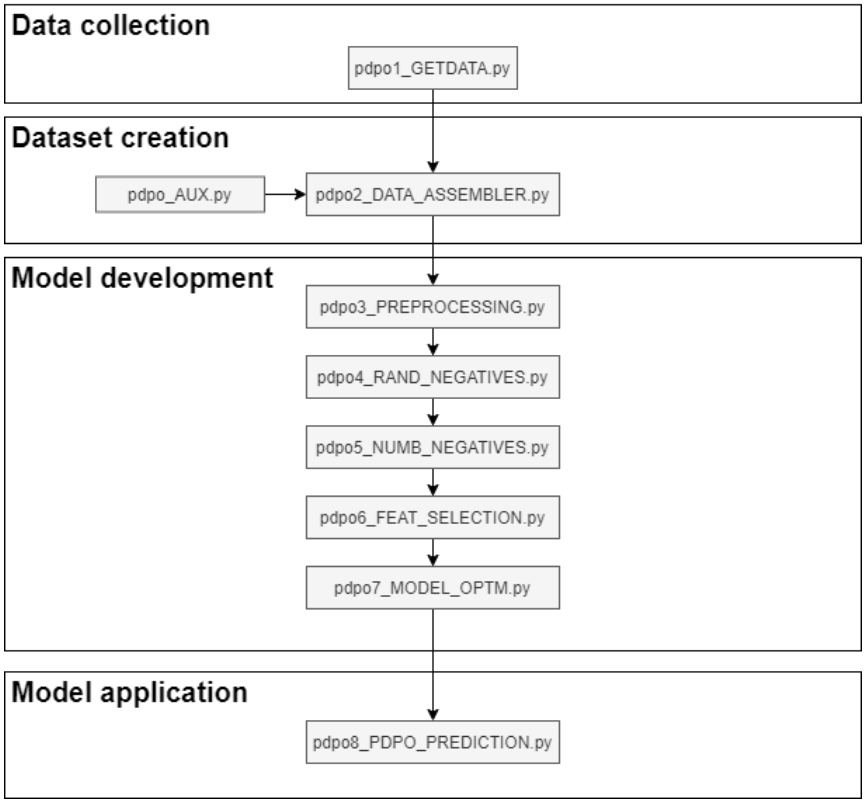


Figure 10: Flowchart of the development steps.

The positive data was collected from NBCI's CDD and NCBI's protein databases, as mentioned in section 3.1. The fasta aminoacids and fasta coding sequences files belonging to the domains related proteins and associated with the positive data query

were downloaded. The class PDPOdata, from script ”pdpo1_GETDATA.py”, is responsible for gathering and selecting information retrieved from NCBI. This class starts by importing a JavaScript Object Notation (JSON) file containing the following information:

1. Twenty four keywords associated with DPO presence;
2. NCBI query for positive data;
3. NCBI query for negative data;
4. Query for positive cases to complement the domains information.

This script only performed the CD-HIT clustering for positive cases, as these might contains duplicates. For negative cases, the script retrieved the fasta aminoacids and fasta coding sequences of all entries from the negative query, and their records in GenBank format, using *Biopython* function ”Entrez.efetch”. Because of the large number of hits produced by the negative case query (463228), the script limits the download to 30.000 entries. Data selection of negative cases was performed by keyword check. If one of the 24 keywords were present within the records features, the record would be deleted. The remaining records were submitted to CD-HIT with a 1.0 threshold to remove duplicated sequences. These steps were performed automatically by the class PDPOdata resulting in the files POSITIVE_DATA.json and NEGATIVE_DATA.json, having 1437 and 22976 sequences, respectively.

The assembly of the datasets was performed through random selection of negative cases for the total number of positive cases., with class PDPOAssembler from the script ”pdpo2_DATA_ASSEMBLER.py” that imports functions from ”pdpo_AUX.py”. After selecting negative cases for each sequence, the class created the features described in section 3.2 using *BioPython* and *Propy*.

Datasets were scaled in ”pdpo3_PREPROCESSING.py” using class PDPOpreprocessing and the metrics compared to evaluate the influence of data scaling. The influence of the selected negative cases was also tested with class PDPONEGATIVES from ”pdpo4_RAND

_NEGATIVES.py". Class PDPOMETRICS from "pdpo5_RAND_NEGATIVES.py" was used to determine the most adequate number of negatives. Then, the most important features were selected with the RFE method in class FeatureSelection from "pdpo6_FEAT_SELECTION.py". The reduced datasets and the scalers were saved with the *pickle* module. The optimization of hyperparameters was performed using class ModelOptimization from script "pdpo7_MODEL
_OPTM.py", and later the models were saved.

Finally, the best models were applied to new data to predict DPOs, using class PDPOPrediction from script "pdpo8_PDPO_PREDICTION.py" that inputs a list of ORFs from a sequenced phage genome in the fasta nucleotide format. This script uses *BioPython* to translate the ORF to amino acid sequence and, using each ORF and translated sequence, calculates the features and applies the scaler and the model. The output consists of an HTML table with the predicted ORF, including the probability of each ORF being a DPO. File "PhageDPO.xml" was generated using Planemo to include the script in the Galaxy platform.

5

RESULTS AND DISCUSSION

The impact of the number and random selection of negative cases was evaluated, as well as data standardization and feature selection. Datasets d2874, d4311, d5748 and d7185 were considered to measure the impact of data pre-processing and the optimal number of negative cases best suited for DPO prediction. The influence of the negative cases was determined by comparing the performance of models obtained with d5748 and d5748R, as these datasets encompass more negative cases. The models were then optimized, and the best-performing ones selected to integrate Galaxy and tested with novel data.

5.1 DATASET PRE-PROCESSING

After creating the datasets, the influence of data standardization in model performance was assessed by assessing the models' accuracy before and after applying the "Standard-Scaler" function to all features. The ANN and SVM models exhibited the most significant increase in PECC, whereas the kNN models showed a decrease in PECC. The other models (DT, RF and NB) did not show significant changes. ANN, SVM and RF exhibit the best PECC values for all datasets. These results are described in supplementary tables S1, S2, S3 and S4 and in Table 5 for the ANN and SVM models.

Table 5: Mean of PECC scores of models ANN, SVM after 5-fold CV for all the datasets

Models	Dataset	PECC (W/OUT SCALER)	PECC (W/ SCALER)
ANN	d2874	0,88	0,93
	d4311	0,89	0,94
	d5748	0,94	0,95
	d7185	0,92	0,96
SVM	d2874	0,81	0,93
	d4311	0,84	0,94
	d5748	0,86	0,95
	d7185	0,89	0,96

Pre-processed datasets created models with higher PECCs, even higher than non-scaled kNN whose PECC decreased with ”StandarScaler”. This way, data scaling proved to be an essential step of model development.

5.2 Influence of the negative cases

As previously mentioned, dataset d5748R was created to assess the impact of the random negative case selection. Datasets d5748 and d5748R were pre-processed using the same approach, and the means of PECC, precision and recall for each model are shown in in Table 6.

Table 6: Mean PECC, Precision and Recall of the models after 5-fold CV for the datasets d5748 and d5748R. The highest values of each metric are shaded in gray.

		KNN	DT	ANN	SVM	RF	NB
d5748	**PECC**	0,88	0,90	0,95	0,95	0,95	0,79
	Precision	0,70	0,80	0,91	0,91	0,93	0,55
	Recall	0,93	0,82	0,90	0,89	0,84	0,90
d5748R	**PECC**	0,87	0,91	0,95	0,95	0,95	0,79
	Precision	0,68	0,82	0,91	0,90	0,94	0,55
	Recall	0,93	0,81	0,90	0,90	0,85	0,90

There were no significant differences in the PECC, precision and recall values of both datasets, indicating that the selected negative cases do not influence model performance.

5.3 Number of negative cases

The impact of the number of negative cases was assessed with the datasets created in section 3.3. The models were created with a fixed number of positive cases and a varying number of negative cases, and their performance evaluated. Table 7 shows that the PECC increases with the number of negative cases, with both precision and recall changing slightly. The NB and kNN models performed worse and were rejected. Although the RF has lower recall, its values of PECC and precision were high. The DT, ANN and SVM models have the best balance between precision and recall. CM were created to assess the number of FP and FN, as shown in Tables 8 to 11.

Table 7: Mean PECC, Precision and Recall after 5-fold CV of KNN, DT, ANN, SVM, RF and NB for datasets d2874, d4311, d5748 and d7185. Models with highest, overall, metrics are shaded in gray.

		KNN	DT	ANN	SVM	RF	NB
d2874	**PECC**	0,76	0,86	0,93	0,93	0,92	0,83
	Precision	0,69	0,85	0,92	0,94	0,93	0,79
	Recall	0,96	0,88	0,93	0,93	0,90	0,90
d4311	**PECC**	0,83	0,89	0,94	0,94	0,94	0,81
	Precision	0,68	0,82	0,91	0,92	0,93	0,65
	Recall	0,94	0,85	0,91	0,91	0,87	0,90
d5748	**PECC**	0,88	0,90	0,95	0,95	0,95	0,79
	Precision	0,70	0,80	0,91	0,91	0,93	0,55
	Recall	0,93	0,83	0,90	0,89	0,84	0,90
d7185	**PECC**	0,90	0,91	0,96	0,96	0,95	0,78
	Precision	0,69	0,78	0,90	0,90	0,92	0,48
	Recall	0,92	0,79	0,90	0,88	0,83	0,90

Table 8: Confusion Matrix for DT, ANN, SVM and RF models created from d2874 after 5-fold CV.

	d2874								
	DT		**ANN**		**SVM**		**RF**		
Pred / Real	**Neg**	**Pos**	**Neg**	**Pos**	**Neg**	**Pos**	**Neg**	**Pos**	**Total**
Neg	1218	219	1328	109	1348	89	1344	93	1437
Pos	179	1258	104	1333	107	1333	137	1300	1437
Total	1397	1477	1432	1442	1455	1422	1481	1393	2874

Table 9: Confusion Matrix for DT, ANN, SVM and RF models created from d4311 after 5-fold CV.

	d4311								
	DT		ANN		SVM		RF		
Pred / Real	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Total
Neg	2604	270	2739	135	2758	116	2782	92	2874
Pos	212	1225	129	1308	132	1305	180	1257	1437
Total	2816	1495	2868	1443	2890	1421	2962	1349	4311

Table 10: Confusion Matrix for DT, ANN, SVM and RF models created from d5748 after 5-fold CV.

	d5748								
	DT		ANN		SVM		RF		
Pred / Real	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Total
Neg	4008	303	4184	127	4186	125	4219	92	4311
Pos	249	1188	141	1296	157	1280	223	1214	1437
Total	4257	1491	4325	1423	4343	1405	4442	1306	5748

Table 11: Confusion Matrix for DT, ANN, SVM and RF models created from d7185 after 5-fold CV.

	d7185								
	DT		ANN		SVM		RF		
Pred / Real	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Total
Neg	5428	320	5611	137	5608	140	5650	98	5748
Pos	298	1139	145	1295	170	1267	246	1191	1437
Total	5726	1459	5756	1432	5778	1407	5896	1289	7185

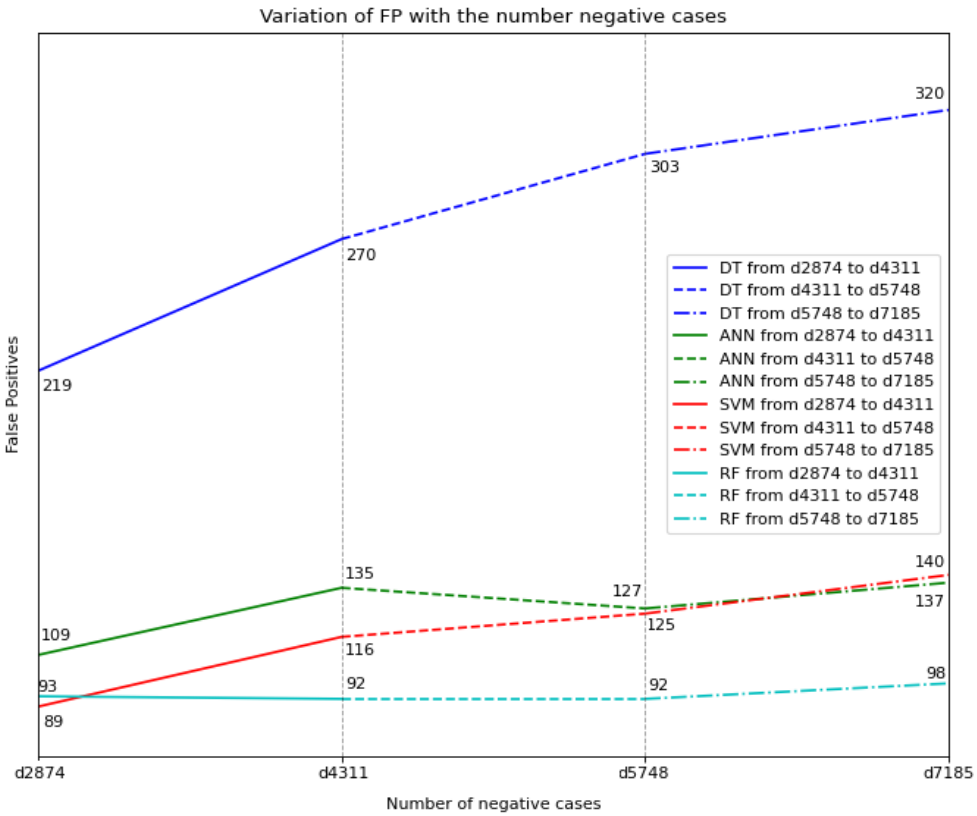


Figure 11: Variation of FP from dataset d2874 to d7185, for models DT, ANN, SVM and RF.

As shown in figure 11, DT was the model with the highest number and the most significant increase in FP. On the other hand, the RF model kept the same number of FP for all datasets. Both ANN and SVM demonstrated similar behaviour from dataset d2874 to d4311 but, from d4311 to d5748, the ANN model decreased the number of FP and the SVM model increased slightly.

From figure 11, DT was the model with the highest number of FP across all datasets and, from dataset d2874 to dataset d4311, the slope with the highest value of all models, indicated a big increased in FP. On the other hand, the RF model, kept practically the same number of FP for all datasets. Both ANN and SVM demonstrated similar behaviour from dataset d2874 to d4311, but from d4311 to d5748 ANN decreased the number of FP and SVM increased slightly. From d5748 to d7185, both ANN and SVM the FP number increased on a small scale.

Figure 12: Variation of FN from dataset d2874 to d7185, for models DT, ANN, SVM and RF.

As shown in Figure 12, increasing the number of negative cases also increased FN for all models. DT was the model with the highest number of FN and, together with RF, a steep increase from dataset d4311 to d5748, and from d5748 to d7185. Both ANN and SVM models had similar behaviour from datasets d2874 to d4311. From d4311 to d5748, both models increased the number of FN, and from d5748 to d7185 the ANN model maintained the number of FN while the SVM increased.

Therefore we can conclude that the RF model had the lowest number of FP, specially form d4311 to d7185, while the ANN and SVM models have fewer FN. These results also confirmed that dataset d2874 produced the models with the lowest number of FP and FN and the highest precision and recall metrics (Table 7). From a biological standpoint, as negative data is composed of a wide variety of phage proteins and the DPO proportion towards those proteins is very low, a dataset with a higher number of negative cases seems more appropriate to solve this problem. ROC and PR curves were created to

assess the models' positive class (minority class) classification performance. These curves are an effective diagnostic tool for imbalanced binary classification models, such as the models obtained from datasets d4311, d5748 and d7185. The curves for the ANN model are illustrated in Figures 13 and 14. Also, SVM and RF models' curves behaviour were similar to the ANN model, while the DT model exhibited the poorest performance of all models. Curves for these models are available in supplementary Figures S1 to S6.

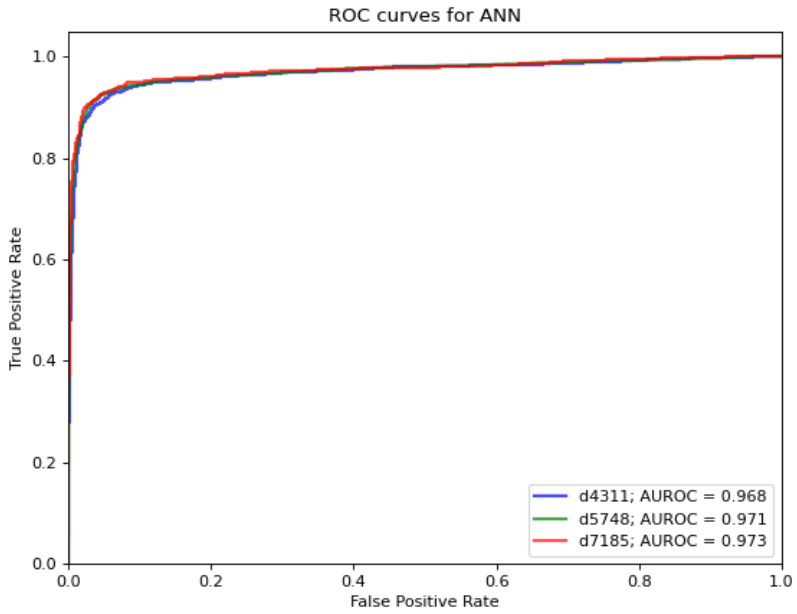


Figure 13: ROC curves representing the ANN model performance for the datasets d4311, d5748 and d7185 with corresponding AUROC values.

ROC curves near the upper left corner indicate a high capacity in separating both classes and have higher AUROC values. For datasets d4311, d5748 and d7185, both curves overlap, indicating similar separation capability, with high AUROC values (0,968, 0,971 and 0,973). As the number of TN is different from dataset d4311 to d7185, models were represented by PR curves.

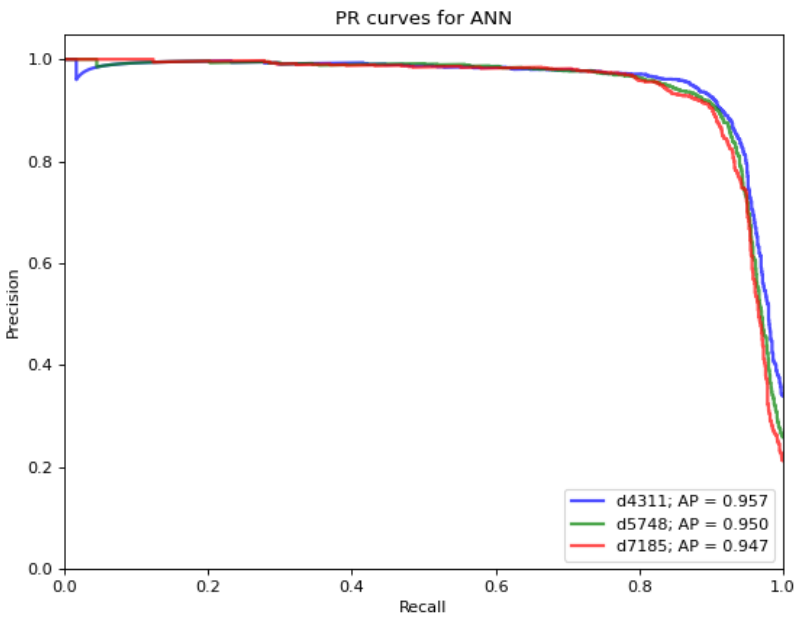


Figure 14: PR curves representing the ANN model performance for the datasets d4311, d5748 and d7185 with corresponding PR value.

As precision is not affected by a large number of negative cases (it measures the number of TP out of the samples predicted as positives TP+FP), its focus is on the positive class rather than the negative class, calculating the probability of correct detection of positive cases. The PR curve's AP value allows inferring if the model identifies positives cases correctly and thoroughly. As shown in Figure 14, both model curves overlap, with similar AP values, indicating a good trade-off between precision and recall. In general, the ANN, SVM and RF models demonstrated high AP values.

5.4 Feature selection

The method used to select the most relevant features of the dataset was the Recursive Feature Elimination (RFE). The RF model was used as an estimator, and RFE with a 5-fold CV, was executed to find the optimal number of features. In this method, features are given a rank number according to their importance; the higher the rank, the less important is the feature for the model. The top rank features (ranking 1) were selected for

both datasets, resulting in 45 features for dataset d4311, 54 features for dataset d5748 and 166 features for dataset d7185, as shown in supplementary table S5. Models were created using the algorithms ANN, SVM and RF, whose metrics are represented in Table 12.

Table 12: PECC, Precision and Recall after 5-fold CV for models ANN, SVM and RF using the reduced datasets.

		ANN	SVM	RF
d4311	**PECC**	0,93	0,93	0,94
	Precision	0,91	0,90	0,93
	Recall	0,90	0,90	0,89
d5748	**PECC**	0,95	0,94	0,95
	Precision	0,91	0,87	0,93
	Recall	0,89	0,88	0,86
d7185	**PECC**	0,96	0,96	0,95
	Precision	0,92	0,91	0,93
	Recall	0,89	0,87	0,84

When comparing the model's metrics obtained from the reduced datasets with the models from Table 7 for dataset d4311, these decreased slightly, except for the RF model, whose recall increased.

For dataset d5748, the SVM model's precision decreased (from 0,91 to 0,87) along with PECC and recall, whereas the RF model's recall increased. Feature reduction was not very significant for ANN models, whose metrics remained similar. However, the RF models recall increased, while all SVM models metrics decreased, mostly precision. For dataset d7185, PECC values remained the same, the precision slightly increased for as models and recall decrease slightly.

Overall, dataset d7185 produced the models with the highest values of PECC and precision, while the ANN, SVM and RF models (d4311) exhibit the highest recall values. As the metrics were not significantly affected, the models obtained from the reduced datasets were further optimized.

5.5 MODEL OPTIMIZATION

The hyperparameters (values described in table 4) for the ANN, SVM and RF models, from the reduced datasets d4311 and d5748, were optimized through GridSearch. The parameters that produced the best results are described from Tables 13 to 15, along with the corresponding PECC, precision and recall for each optimized model.

Table 13: Hyperparameter values from GridSearch output with PECC, Precision and Recall after 5-fold CV for models ANN, SVM, and RF obtained from dataset d4311.

Model	Hyperparameter	PECC	Precision	Recall
ANN	'activation': 'tanh', 'alpha': 0.001, 'hidden_layer_sizes': (50,), 'solver': 'adam'	0,94	0,92	0,91
SVM	'C': 10, 'gamma': 0.1, 'kernel': 'rbf'	0,95	0,96	0,89
RF	'bootstrap': False, 'criterion': 'entropy', 'max_depth': None, 'max_features': 'auto', 'min_samples_leaf': 2, 'min_samples_split': 6, 'n_estimators': 100	0,94	0,93	0,90

Table 14: Hyperparameter values from GridSearch output with PECC, Precision and Recall after 5-fold CV for models ANN, SVM, and RF obtained from dataset d5748.

Model	Hyperparameter	PECC	Precision	Recall
ANN	'activation': 'tanh', 'alpha': 0.01, 'hidden_layer_sizes': (100,), 'solver': 'adam'	0,95	0,91	0,89
SVM	'C': 10, 'gamma': 0.1, 'kernel': 'rbf'	0,96	0,95	0,87
RF	'bootstrap': False, 'criterion': 'entropy', 'max_depth': None, 'max_features': 'auto', 'min_samples_leaf': 2, 'min_samples_split': 2, 'n_estimators': 200	0,95	0,93	0,88

Table 15: Hyperparameter values from GridSearch output with PECC, Precision and Recall after 5-fold CV for models ANN, SVM, and RF obtained from dataset d7185.

Model	Hyperparameter	PECC	Precision	Recall
ANN	'activation': 'relu', 'alpha': 0.0001, 'hidden_layer_sizes': (100,), 'solver': 'adam'	0,96	0,92	0,89
SVM	'C': 10, 'gamma': 0.01, 'kernel': 'rbf'	0,97	0,94	0,89
RF	'bootstrap': False, 'criterion': 'entropy', 'max_depth': None, 'max_features': 'auto', 'min_samples_leaf': 2, 'min_samples_split': 6, 'n_estimators': 200	0,96	0,92	0,86

After optimization, and comparing with Table 7, the accuracy remained practically the same for all models. The SVM model showed a significant increase in precision for all datasets, while the recall decreased slightly. The ANN models metrics remained practically the same for all datasets. The RF model increased it's recall in all datasets, while the other metrics remained the same. The ANN and SVM models from dataset d4311 exhibited respectively, the highest recall (0,91%) and precision (0,96%). The ANN model maintained the best balance between precision and recall in all datasets. CMs (Tables from 16 to 18) and ROC and PR curves (Figures from 16 to 18) were created, for each dataset, to assess the models' performance.

Table 16: Confusion matrix for the optimized models ANN, SVM and RF originated from dataset d4311 with 5-fold CV

	d4311						
	ANN		**SVM**		**RF**		
Pred / Real	**Neg**	**Pos**	**Neg**	**Pos**	**Neg**	**Pos**	**Total**
Neg	2753	121	2816	58	2779	95	2874
Pos	132	1305	164	1273	147	1290	1437
Total	2885	1426	2980	1331	2926	1385	4311

For dataset d4311, the SVM model had the lowest FP (58) and the highest number of FN (164), whereas ANN model exhibited the highest FP (121) and the lowest number of FN (132). These values are shaded in gray in Table 16.

Table 17: Confusion matrix for the optimized models ANN, SVM and RF originated from dataset d5748 with 5-fold CV

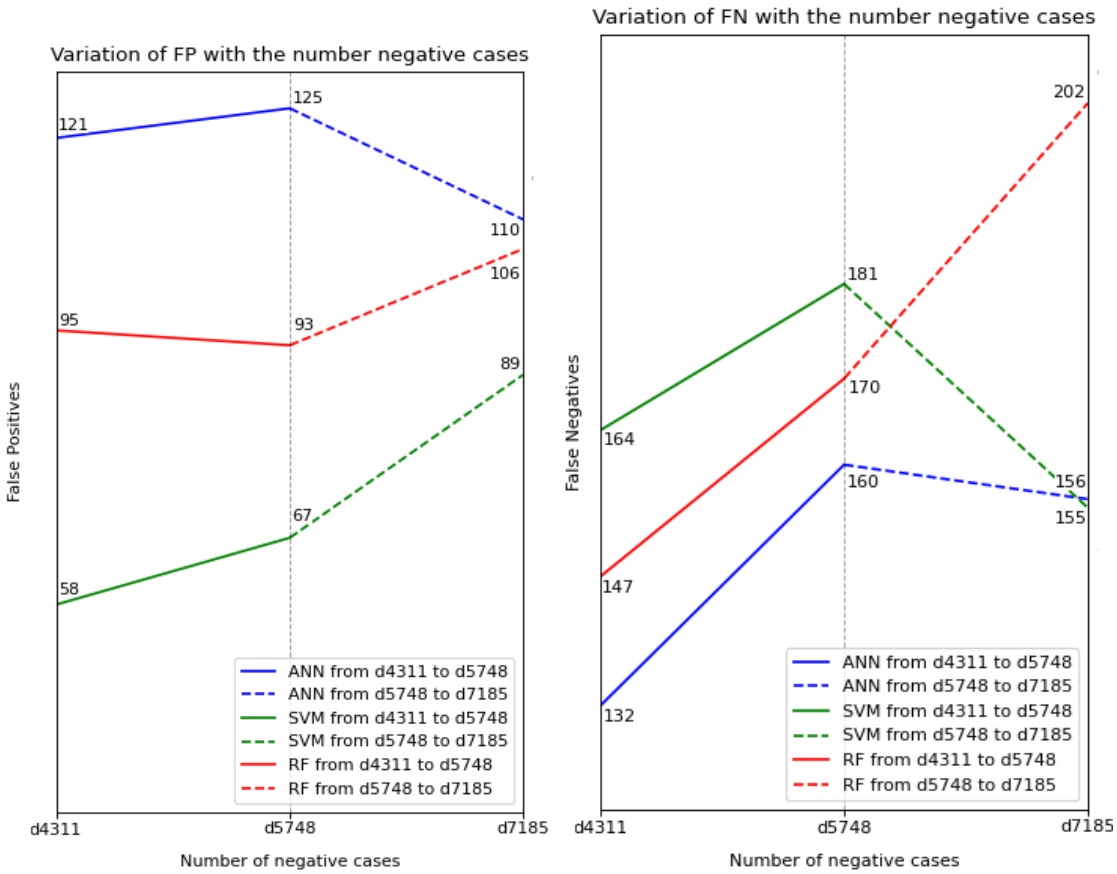
	d5748						
	ANN		**SVM**		**RF**		
Real \ Pred	**Neg**	**Pos**	**Neg**	**Pos**	**Neg**	**Pos**	**Total**
Neg	4186	125	4244	67	4218	93	4311
Pos	160	1277	181	1256	170	1267	1437
Total	4346	1402	4425	1323	4388	1360	5748

For dataset d5748, the SVM model showed the lowest FP (67) and the highest number of FN (181), while the ANN model exhibited the highest FP (125) and the lowest number of FN (160). These values are shaded in gray in Table 17.

For dataset d7185, the SVM model showed the lowest number of FP (89) and FN (155). The RF model exhibited the higher number of FN (202) and ANN the higher number of FP (110). These values are shaded in gray in Table 18.

Table 18: Confusion matrix for the optimized models ANN, SVM and RF originated from dataset d7185 with 5-fold CV

	d7185						
	ANN		**SVM**		**RF**		
Real \ Pred	**Neg**	**Pos**	**Neg**	**Pos**	**Neg**	**Pos**	**Total**
Neg	5638	110	5669	89	5642	106	5748
Pos	156	1281	155	1282	202	1235	1437
Total	5794	1391	5824	1371	5844	1341	7185



(a) (b)

Figure 15: Variation of FP (a) and FN (b) from dataset d4311 to d7185, for models ANN, SVM and RF.

Figure 15 (a) shows that, from d4311 to d5748, the increment of negative cases increased FP in both the SVM and ANN models . The rate of increase of FP on the SVM models was higher for the ANN models, whereas the RF models decreased the number of FP with the increment of negative cases. From d5748 to d7185, while both SVM and RF models increased the number of FP, the ANN model decreased FP number.

Figure 15 (b), from d4311 to d5748, demonstrates an increase of FN with the increment of negative cases, with ANN and RF models holding the highest increase rate. From d5748 to d7185, the SVM and the ANN models decreased the number of FN while the RF model increased significantly.

Overall, and after model optimization, the number of FN was higher than the number of FP. A possible explanation might be associated with type II errors while querying the negative sequences. The negative sequences were retrieved from a set of NCBI entries, whose only criteria for selection was not having keywords related to DPO; thus, part of such entries may be unidentified or poorly annotated DPOs entries.

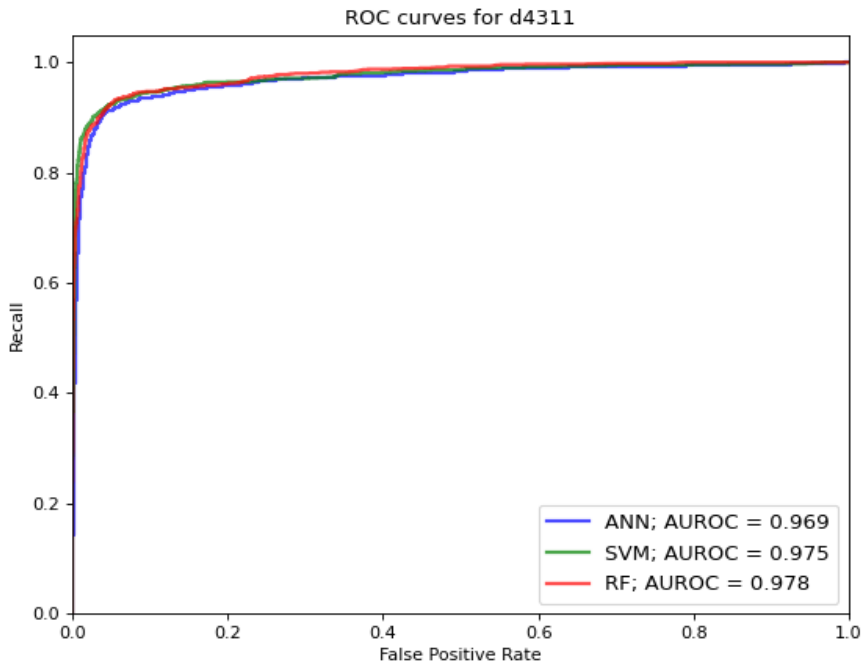
The ROC and PR curves of the models for each dataset (Figures from 16 to 18) showed overlapping curves, indicating a similar performance for all models.

For dataset d4311, the RF model had the highest value of AUROC and, along with the SVM model, the highest AP's values. The latter value is the metric employed when the model's focus is correctly identifying positive samples in an unbalanced dataset.

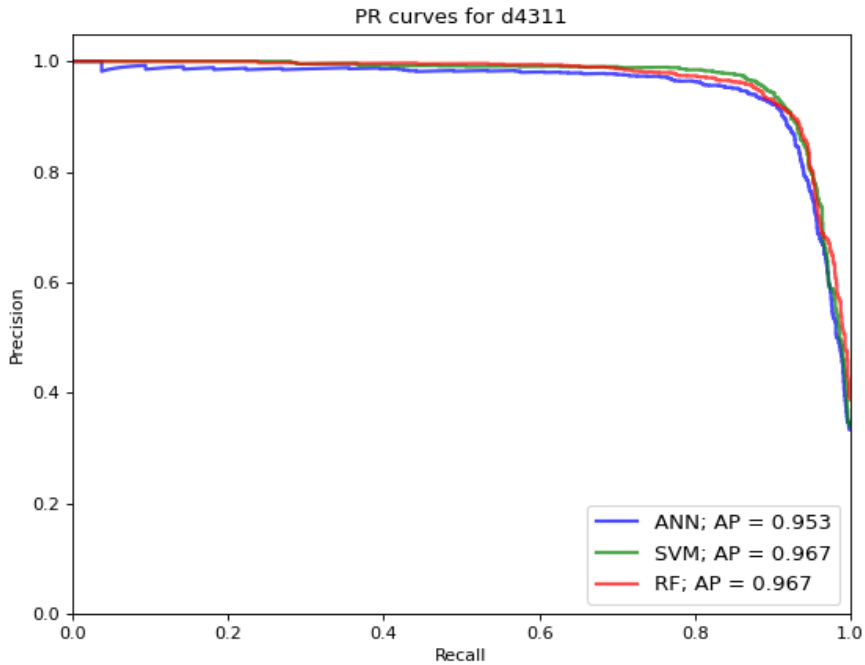
For dataset d5748, both RF and SVM models show the highest values of AUROC and AP, with the SVM model curve marginally above the others. For dataset d7185, the RF and SVM show the highest values of AUROC and the SVM the highest AP value.

The SVM model from dataset d4311 and the ANN model from dataset d7185 were selected for further testing and integration on the Galaxy platform. The SVM model from d4311 showed a good AP and AUROC values and the lowest number of FP. The ANN model from d7185 was the only model that reduced the number of FP from dataset d5748 to d7185 which means an improved training for negative cases.

The high precision, PECC and AP of the SVM (d4311) focus on TP detection while avoiding FP. The ANN (d7185) ensures that all DPOs are correctly identified due to it's high recall. The output of both models will be the probability of each protein being a DPO.

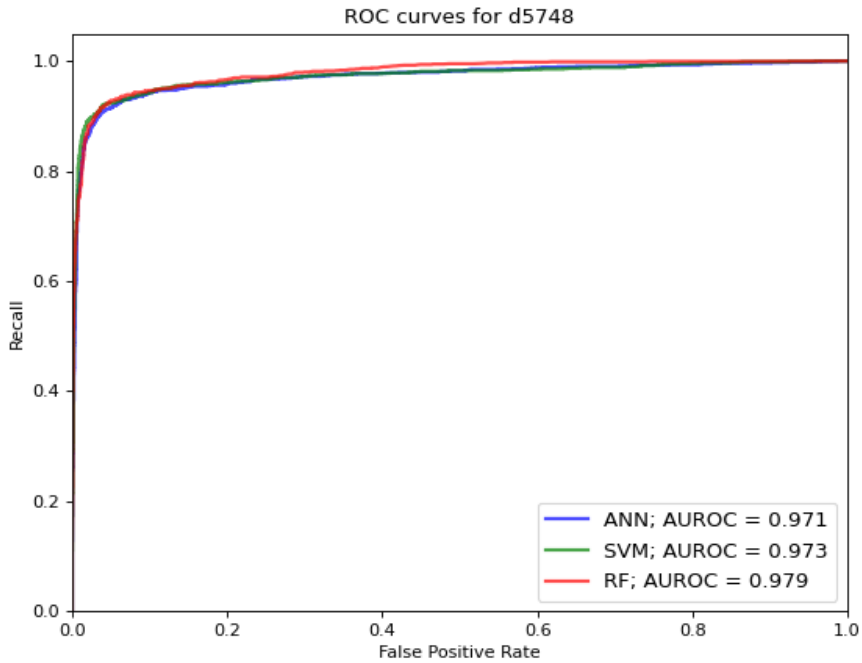


(a)

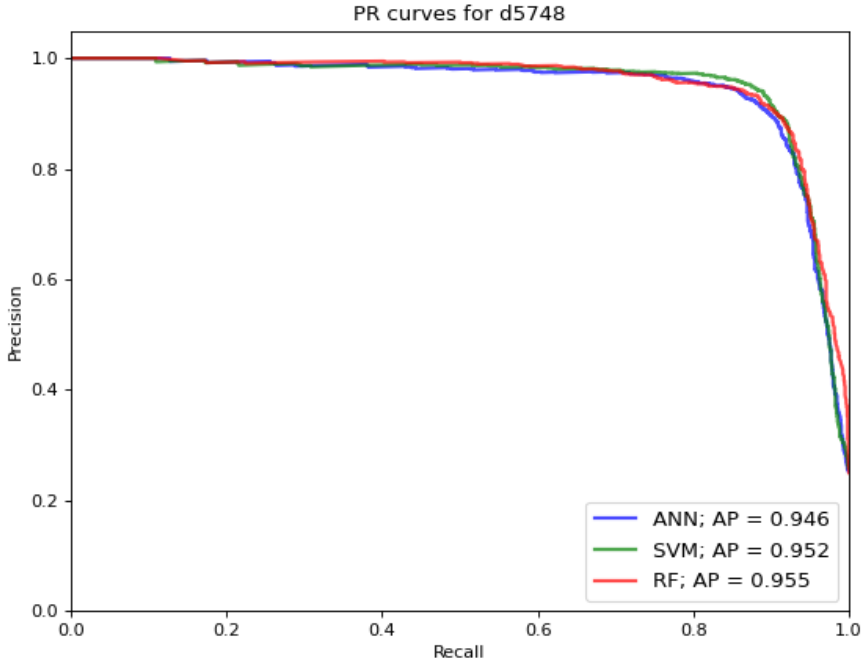


(b)

Figure 16: ROC and PR curves, (a) and (b) respectively, representing the ANN, SVM and RF models created from dataset d4311. Correspondent AUROC and AP values were calculated.

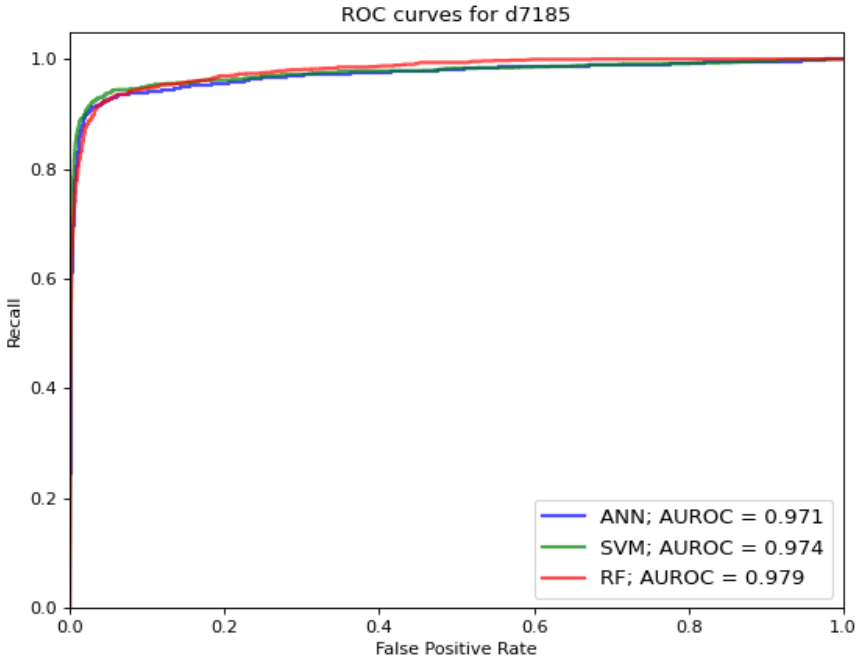


(a)

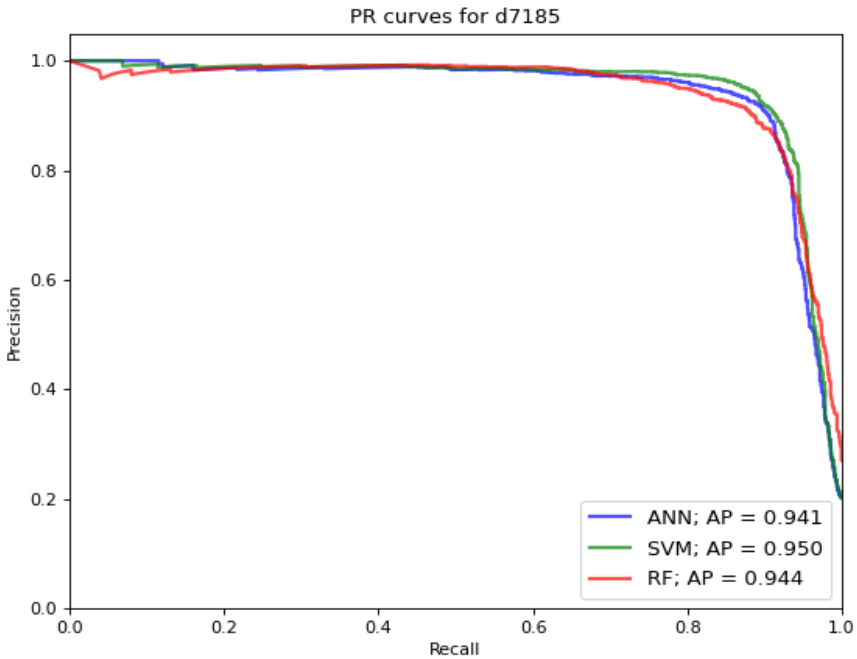


(b)

Figure 17: ROC and PR curves, (a) and (b) respectively, representing the ANN, SVM and RF models created from dataset d5748. Correspondent AUROC and AP values were calculated.



(a)



(b)

Figure 18: ROC and PR curves, (a) and (b) respectively, representing the ANN, SVM and RF models created from dataset d7185. Correspondent AUROC and AP values were calculated.

5.6 Publishing to Galaxy

PhageDPO was integrated into Galaxy to provide a user-friendly interface, as illustrated in Figure 19. In this interface is possible to see all tool's required inputs and advanced options. PhageDPO has a single input, in the form of a fasta file format, containing the DNA coding sequences with unique identifiers. As advanced option, users select the model to run, the SVM model (default) or the ANN model. The SVM model return fewer sequences with a higher probability of being real DPOs avoiding FP. On the other hand, the ANN model ensures that all DPOs are correctly identified due to it's high recall.

PhageDPO outputs an HTML table (Figure 20) with the sequence's identification and percentage of positive prediction for DPO, ranked from high probability to low probability. An explanation of how to use PhageDPO, including a description of its inputs and outputs, is available in the tool interface.

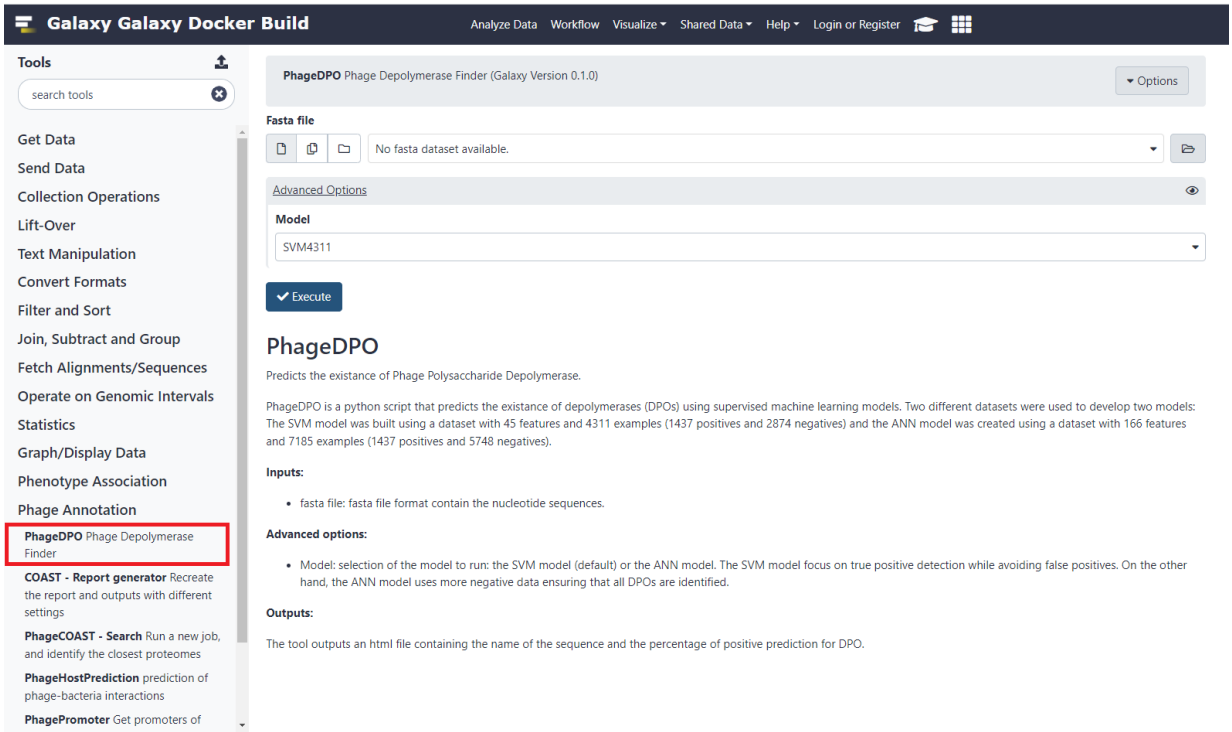
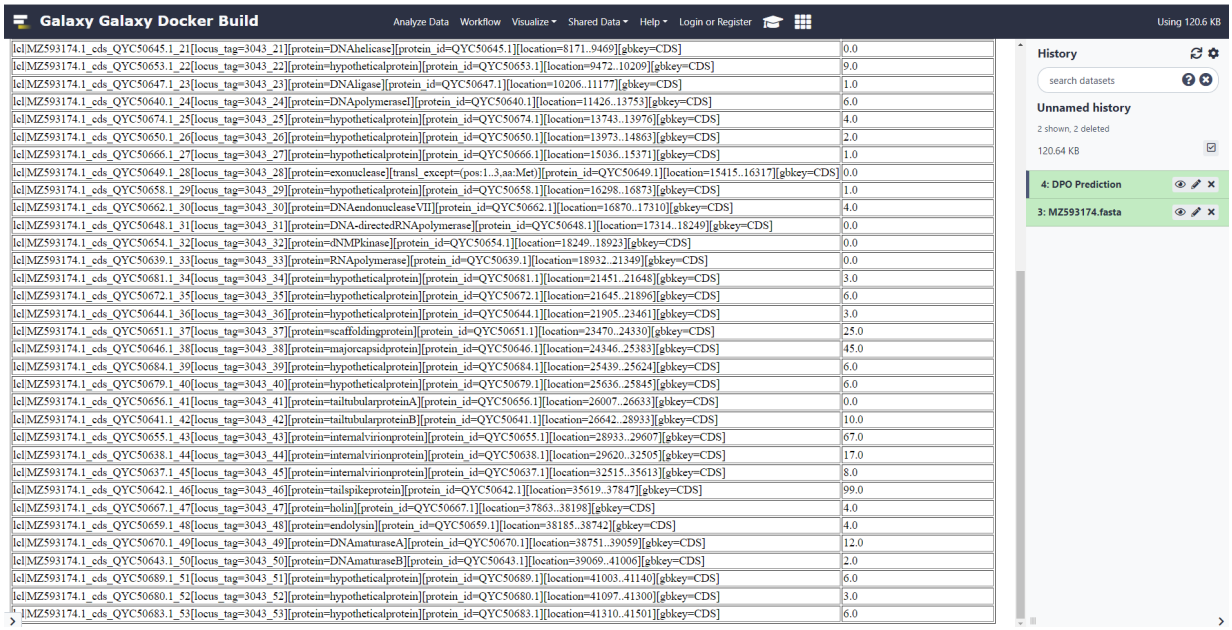


Figure 19: PhageDPO Galaxy interface.



lcl|MZ593174.1_cds_QYC50645.1_21[locus_tag=3043_21][protein=DNAhelicase][protein_id=QYC50645.1][location=8171..9469][gbkey=CDS]	0.0
lcl|MZ593174.1_cds_QYC50653.1_22[locus_tag=3043_22][protein=hypotheticalprotein][protein_id=QYC50653.1][location=9472..10209][gbkey=CDS]	9.0
lcl|MZ593174.1_cds_QYC50647.1_23[locus_tag=3043_23][protein=DNAligase][protein_id=QYC50647.1][location=10206..11177][gbkey=CDS]	1.0
lcl|MZ593174.1_cds_QYC50640.1_24[locus_tag=3043_24][protein=DNApolymeraseI][protein_id=QYC50640.1][location=11426..13753][gbkey=CDS]	6.0
lcl|MZ593174.1_cds_QYC50674.1_25[locus_tag=3043_25][protein=hypotheticalprotein][protein_id=QYC50674.1][location=13743..13976][gbkey=CDS]	4.0
lcl|MZ593174.1_cds_QYC50650.1_26[locus_tag=3043_26][protein=hypotheticalprotein][protein_id=QYC50650.1][location=13973..14863][gbkey=CDS]	2.0
lcl|MZ593174.1_cds_QYC50666.1_27[locus_tag=3043_27][protein=hypotheticalprotein][protein_id=QYC50666.1][location=15036..15371][gbkey=CDS]	1.0
lcl|MZ593174.1_cds_QYC50649.1_28[locus_tag=3043_28][protein=exonuclease][transl_except=(pos:1..3,aa:Met)][protein_id=QYC50649.1][location=15415..16317][gbkey=CDS]	0.0
lcl|MZ593174.1_cds_QYC50658.1_29[locus_tag=3043_29][protein=hypotheticalprotein][protein_id=QYC50658.1][location=16298..16873][gbkey=CDS]	1.0
lcl|MZ593174.1_cds_QYC50662.1_30[locus_tag=3043_30][protein=DNAendonucleaseVII][protein_id=QYC50662.1][location=16870..17310][gbkey=CDS]	4.0
lcl|MZ593174.1_cds_QYC50648.1_31[locus_tag=3043_31][protein=DNA-directedRNApolymerase][protein_id=QYC50648.1][location=17314..18249][gbkey=CDS]	0.0
lcl|MZ593174.1_cds_QYC50654.1_32[locus_tag=3043_32][protein=dNMPkinase][protein_id=QYC50654.1][location=18249..18923][gbkey=CDS]	0.0
lcl|MZ593174.1_cds_QYC50639.1_33[locus_tag=3043_33][protein=RNApolymerase][protein_id=QYC50639.1][location=18932..21349][gbkey=CDS]	0.0
lcl|MZ593174.1_cds_QYC50681.1_34[locus_tag=3043_34][protein=hypotheticalprotein][protein_id=QYC50681.1][location=21451..21648][gbkey=CDS]	3.0
lcl|MZ593174.1_cds_QYC50672.1_35[locus_tag=3043_35][protein=hypotheticalprotein][protein_id=QYC50672.1][location=21645..21896][gbkey=CDS]	6.0
lcl|MZ593174.1_cds_QYC50644.1_36[locus_tag=3043_36][protein=hypotheticalprotein][protein_id=QYC50644.1][location=21905..23461][gbkey=CDS]	3.0
lcl|MZ593174.1_cds_QYC50651.1_37[locus_tag=3043_37][protein=scaffoldingprotein][protein_id=QYC50651.1][location=23470..24330][gbkey=CDS]	25.0
lcl|MZ593174.1_cds_QYC50646.1_38[locus_tag=3043_38][protein=majorcapsidprotein][protein_id=QYC50646.1][location=24346..25383][gbkey=CDS]	45.0
lcl|MZ593174.1_cds_QYC50684.1_39[locus_tag=3043_39][protein=hypotheticalprotein][protein_id=QYC50684.1][location=25439..25624][gbkey=CDS]	6.0
lcl|MZ593174.1_cds_QYC50679.1_40[locus_tag=3043_40][protein=hypotheticalprotein][protein_id=QYC50679.1][location=25636..25845][gbkey=CDS]	6.0
lcl|MZ593174.1_cds_QYC50656.1_41[locus_tag=3043_41][protein=tailtubularproteinA][protein_id=QYC50656.1][location=26007..26633][gbkey=CDS]	0.0
lcl|MZ593174.1_cds_QYC50641.1_42[locus_tag=3043_42][protein=tailtubularproteinB][protein_id=QYC50641.1][location=26642..28933][gbkey=CDS]	10.0
lcl|MZ593174.1_cds_QYC50655.1_43[locus_tag=3043_43][protein=internalvirionprotein][protein_id=QYC50655.1][location=28933..29607][gbkey=CDS]	67.0
lcl|MZ593174.1_cds_QYC50638.1_44[locus_tag=3043_44][protein=internalvirionprotein][protein_id=QYC50638.1][location=29620..32505][gbkey=CDS]	17.0
lcl|MZ593174.1_cds_QYC50637.1_45[locus_tag=3043_45][protein=internalvirionprotein][protein_id=QYC50637.1][location=32515..35613][gbkey=CDS]	8.0
lcl|MZ593174.1_cds_QYC50642.1_46[locus_tag=3043_46][protein=tailspikeprotein][protein_id=QYC50642.1][location=35619..37847][gbkey=CDS]	99.0
lcl|MZ593174.1_cds_QYC50667.1_47[locus_tag=3043_47][protein=holin][protein_id=QYC50667.1][location=37863..38198][gbkey=CDS]	4.0
lcl|MZ593174.1_cds_QYC50659.1_48[locus_tag=3043_48][protein=endolysin][protein_id=QYC50659.1][location=38185..38742][gbkey=CDS]	4.0
lcl|MZ593174.1_cds_QYC50670.1_49[locus_tag=3043_49][protein=DNAmaturaseA][protein_id=QYC50670.1][location=38751..39059][gbkey=CDS]	12.0
lcl|MZ593174.1_cds_QYC50643.1_50[locus_tag=3043_50][protein=DNAmaturaseB][protein_id=QYC50643.1][location=39069..41006][gbkey=CDS]	2.0
lcl|MZ593174.1_cds_QYC50689.1_51[locus_tag=3043_51][protein=hypotheticalprotein][protein_id=QYC50689.1][location=41003..41140][gbkey=CDS]	6.0
lcl|MZ593174.1_cds_QYC50680.1_52[locus_tag=3043_52][protein=hypotheticalprotein][protein_id=QYC50680.1][location=41097..41300][gbkey=CDS]	3.0
lcl|MZ593174.1_cds_QYC50683.1_53[locus_tag=3043_53][protein=hypotheticalprotein][protein_id=QYC50683.1][location=41310..41501][gbkey=CDS]	6.0

Figure 20: HTML table returned by PhageDPO for the coding sequences extracted from NCBI of *Acinetobacter phage vB_Api_3043-K38*, with accession number MZ593174.1.

5.7 Model testing

The SVM and ANN models were tested to assess their predictive capabilities. Two phage proteins validation sets were created, available in supplementary material Table S6 and S7. The positive set consisted of 157 DPO positive proteins, and the negative set was composed of 157 DPO negative proteins. The two models were assessed with both validation sets, and the distribution of the variable DPO prediction was visualized through density plots (Figures 21 and 22).

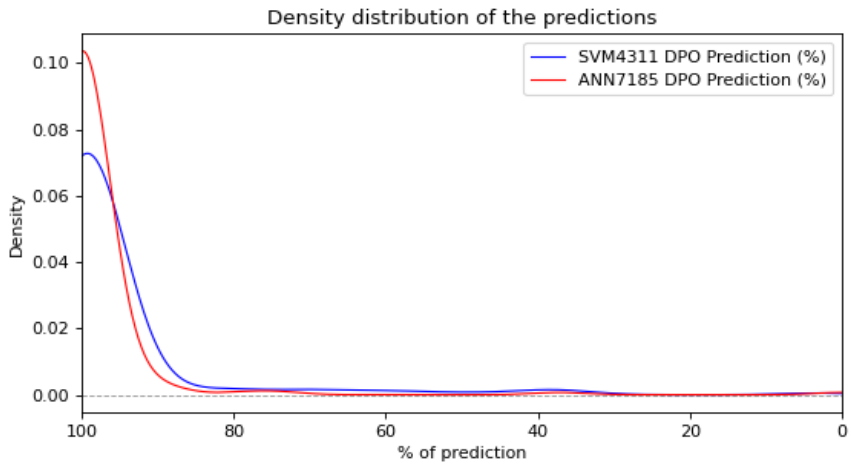


Figure 21: Density distribution for the output of the models ANN and SVM for DPO positive proteins.

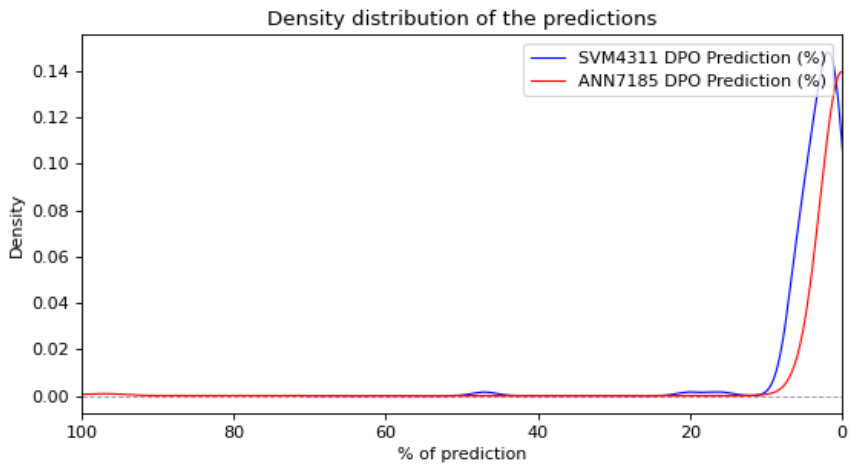


Figure 22: Density distribution for the output of the models ANN and SVM for DPO negative proteins.

As expected, for positive DPO proteins (Figure 21), the high-density prediction zone is near 100%. While for negative DPO proteins (Figure 22), the prediction zone moves towards 0% for all models. The models' predictions for both positive and negative cases of the independent validation dataset are shown in Supplementary Tables S8 and S9, respectively. FP and FN cases were obtained considering a threshold of 10.0% and 90.0%, respectively. Predictions below 90.0% in the positive validation dataset were considered FN, and predictions above 10.0% in the negative validation dataset were

considered FP. PECC, precision and recall of the three models were calculated as shown in Table 19.

Table 19: PECC, Precision and Recall for models ANN and SVM obtained from the validation dataset composed of 157 positive cases and 157 negative cases.

Model	PECC	Precision	Recall
SVM4311	0.95	0.98	0.91
ANN7185	0.98	0.99	0.96

Furthermore, the phageDPO predictive models were applied to four phage genomes, known to encode distinct DPOs to assess the robustness of our models, namely: *Acinetobacter phage vB_Api_3043-K38*, *Klebsiella phage RAD2*, *Pseudomonas phage LUZ19* and *Escherichia phage vB_EcoP_G7C* with Accession Numbers MZ593174.1, NC_055956.1, NC_010326 and NC_015933, respectively. Their CDSs were obtained from their respective records in NCBI.

Acinetobacter phage vB_Api_3043-K38 is a well known phage that encodes a single DPO (QYC50642) degrading capsule [119]. The models predictions for all its CDSs are shown in Supplementary Table S10. For this phage, as depicted in Table 20, both the SVM and ANN models predicted the expected protein (QYC50642.1) with 99% and 100% probability respectively, of being a DPO.

Table 20: DPO top prediction percentages of the SVM and ANN models for *Acinetobacter phage vB_Api_3043-K38* (MZ593174.1). The corresponding proteins' identifiers are also shown.

Organism	Model	Protein identifier	DPO Prediction (%)
Acinetobacter phage	SVM4311	QYC50642.1	99.0
vB_Api_3043-K38	ANN7185	QYC50642.1	100.0

Klebsiella phage RAD2 encodes a single DPO (YP_010115729.1) that targets the capsular polysaccharides of the *Klebsiella pneumoniae* [120]. The models predictions for

all its CDSs are shown in Supplementary Table S11. For this phage, as depicted in table 21, the SVM model predicted the correct protein with 99% probability, while the ANN model predicted the same protein with 100% probability. Both models also predicted, with lower percentage, a false positive protein (YP_010115728.1), annotated as a putative tail spike.

Table 21: DPO top prediction percentages of the SVM and ANN models for *Klebsiella phage RAD2* (NC_055956.1). The corresponding proteins' identifiers are also shown.

Organism	Model	Protein identifier	DPO Prediction (%)
Klebsiella phage RAD2	SVM4311	YP_010115729.1	99.0
		YP_010115728.1	81.0
	ANN7185	YP_010115729.1	100.0
		YP_010115728.1	78.0

During spot tests at the Center of Biological Engineering, *Pseudomonas phage LUZ19* exhibited an halo formation; however, an analysis of its genome did not reveal any gene responsible for encoding DPO. The predictions of both models are shown in Supplementary Table S12 and depicted in Table 22. For this phage the SVM model predicted a single protein (YP_001671985.1) with 96% probability of being a DPO, while the the ANN model predicted other protein (YP_001671979.1) with a maximum probability of 28%. Protein YP_001671985.1, predicted by the SVM model, is annotated as a tail fibre. This protein-encoding gene is associated with a phage tail, which is a good indicative that it may carry the predicted depolymerase activity. Whereas protein YP_001671979.1, predicted by the ANN model, is annotated as a tail tubular protein, which may indicate this as false positive result. Further lab test should be conducted to access the true DPO location. Nevertheless, these results may be used to guide the wet-lab analysis procedure.

Table 22: DPO top prediction percentages of the SVM and ANN models for *Pseudomonas phage LUZ19* (NC_010326). The corresponding proteins' identifiers are also shown.

Organism	Model	Protein identifier	DPO Prediction (%)
Pseudomonas phage LUZ19	SVM4311	YP_001671985.1	96.0
	ANN7185	YP_001671979.1	28.0

The *Escherichia phage vB_EcoP_G7C* has a new kind of DPO (YP_004782195.1), that modifies instead of degrading the LPS [34]. The predictions for this phage are shown in Supplementary Table S13 and depicted in Table 23. The SVM model predicted the correct protein (YP_004782195.1) with 94% probability and a second one (YP_004782196.1) with 100% probability. The ANN model predicted the same proteins (YP_004782195.1, YP_004782196.1) with 100% and 92% respectively, and third (YP_004782143.1) with 90% probability. The protein YP_004782195.1, predicted by both models, is annotated as a tail fibre. Further lab tests should be performed in order to find if this phage encodes two DPOs, as suggested by the predictive model. Protein YP_004782143.1, predicted by the ANN model, is annotated as RNA polymerase, and therefore, a false positive.

Table 23: DPO top prediction percentages of the SVM and ANN models for *Escherichia phage vB_EcoP_G7C* (NC_015933). The corresponding proteins' identifiers are also shown.

Organism	Model	Protein identifier	DPO Prediction (%)
Escherichia phage vB_EcoP_G7C	SVM4311	YP_004782196.1	100.0
		YP_004782195.1	94.0
	ANN7185	YP_004782196.1	100.0
		YP_004782195.1	92.0
		YP_004782143.1	90.0

Next, we also applied our models to search DPO in prophages. One of the great advances in using DPO as an anti-virulence weapon against pathogenic bacteria is to

harness these proteins in prophages inserted in bacterial genomes. Many bacterial genomes deposited in public databases can contain phage DNAs integrated (prophages) in the bacterial chromosome. Moreover, bacteria may contain multiple prophages in their chromosomes. The *Escherichia coli O157:H7* strain Sakai is the most extreme case, as it contains 18 prophage genome elements, which amount to 16% of its total genome content [121], demonstrating the vast diversity of prophage sequences in the bacterial population and possible DPO encoding proteins that could be used.

In our tests, we used prophages extracted from the *Acinetobacter baumannii strain A85* and *Acinetobacter baumannii ATCC 19606* genomes using Phage Search Tool Enhanced Release (Phaster), which is a widely used web servers for identifying putative prophages in bacterial genomes [122]. Results from Table 24 demonstrated that *A baumannii strain A85* contained one prophage (location 3477508-3510350) with one protein (tail function) identified as a possible DPO with high probability (SVM 99% and ANN 97%), which are shown in Supplementary Table S14.

Table 24: DPO top prediction percentages of the SVM and ANN models for *Acinetobacter baumannii strain A85* prophage located in (3477508-3510350). The corresponding proteins' identifiers are also shown.

Organism	Model	Protein identifier	DPO Prediction (%)
Acinetobacter baumannii strain A85 (3477508-3510350)	SVM4311	ASF78667.1	99.0
	ANN7185	ASF78667.1	97.0
		ASF78673.1	85.0

Table 25: DPO top prediction percentages of the SVM and ANN models for *Acinetobacter baumannii ATCC 19606* prophage located in (78042-120394). The corresponding proteins' identifiers are also shown.

Organism	Model	Protein identifier	DPO Prediction (%)
Acinetobacter baumannii ATCC 19606 (78042-120394)	SVM4311	ENW74131.1	62.0
	ANN7185	ENW74149.1	100.0
		ENW74133.1	99.0
		ENW74134.1	99.0
		ENW74148.1	85.0
		ENW74131.1	76.0

Table 26: DPO top prediction percentages of the SVM and ANN models for *Acinetobacter baumannii ATCC 19606* prophage located in (274341-319584). The corresponding proteins' identifiers are also shown.

Organism	Model	Protein identifier	DPO Prediction (%)
Acinetobacter baumannii ATCC 19606 (274341-319584)	SVM4311	ENW74324.1	91.0
	ANN7185	ENW74387.1	94.0

Similarly, *A baumannii ATCC* 19606 contained two prophages (located in 78042-120394 and 274341-319584), Tables 25 and 26, respectively. For the first prophage, the SVM and ANN models identified a tail protein with 62% and 76% probability, although four more proteins were predicted with high probability by the ANN model. In the second prophage, Table 26, the SVM model predicted with 91% probability that a tail protein would be a DPO, while the ANN model predicted a different protein with 91% probability. The full prediction for both prophages from *A baumannii ATCC 19606* are shown in Supplementary Tables S15 and S16. As so far, all DPO have been found in phage tails, such results show that these models, although not trained with prophage sequences data, are likely correctly predicting DPO proteins.

6

CONCLUSION AND FUTURE WORK

Depolymerases (DPOs) are emerging phage-derived proteins with enormous potential to be used to control pathogenic bacteria. This thesis aimed to develop the first bioinformatics tool based on ML algorithms to predict these genes in phage genomes, to hasten the isolation and exploration of DPOs in the biotechnology field.

The data pre-processing with StandardScaler generated better results for ANN and SVM models, whereas other models' performance did not show significant changes. Random selection of negative cases did not affect model performance as model metrics between datasets d5748 and d5748R remained the same. However, the influence of the number of negative cases affected model performance. A higher number of negatives led to higher PECC, with Precision and Recall both changing slightly. Models SVM and RF exhibited a low number of FP, while ANN and SVM revealed a low number of FN, as shown in the models' CMs. Results also confirmed that dataset d2874 produced the models with the lowest number of FP and FN, and the highest Precision and Recall; however, from a biological standpoint, this result would be unexpected, as the DPOs proportion towards the phage genome proteins is very low.

Feature selection and optimization were applied to datasets d4311, d5748 and d7185, identifying 45, 54 and 166 relevant features, respectively.

After optimization, SVM and ANN models' precision improved, with the SVM model's recall to slightly decrease. The RF model increased its recall in all datasets. Overall, the accuracy remained the same. The SVM model created with dataset d4311 presented 0,95% accuracy, 0,96% precision and 0,89% recall and the ANN model created with

dataset d7185 presented 0,96% accuracy, 0,92% precision and 0,89% recall. Both models were selected to be included in the PhageDPO tool implemented in the Galaxy platform. In an independent validation dataset, the SVM model presented 95% accuracy, 98% precision and 91% recall, the ANN model presented 98% accuracy, 99% precision and 96% recall. Both models displayed high average Precision values in the PR curves, and a good balance between metrics thus being assigned to be integrated in PhageDPO.

Although untrained with such data, PhageDPO's analysis of phage genomes known to encode distinct DPOs within prophages revealed encouraging results. Whereas the ANN model can produce more FP, the SVM model predicts less DPOs but with a higher probability of being DPO.

The goal proposed for this work was accomplished: a new online tool with a user-friendly interface for predicting DPO was developed. However, for future work, some aspects can be improved:

- Collect more true positive DPO sequences, to increase the datasets with improved curation for both positive and negative cases. DPOs data-mining over scientific publications should also be considered;
- Other features that can improve the models should be explored;
- Different hyperparameter values for model optimization should be tested;
- Employ ensemble methods to improve predictions. Multiple learning algorithms can be used to achieve better predictive performance than that that could be obtained from any of the learning algorithms alone.

All the code developed for this work is available in the platform GitLab in the following link: https://gitlab.bio.di.uminho.pt/josegduarte/PDPO

BIBLIOGRAPHY

[1] Goodwin, S., McPherson, J. D., & McCombie, W. R. (2016). Coming of age: ten years of next-generation sequencing technologies. *Nature Reviews Genetics*, *17*(6), 333–351.

[2] Clokie, M. R., Millard, A. D., Letarov, A. V., & Heaphy, S. (2011). Phages in nature. *Bacteriophage*, *1*(1), 31–45.

[3] Drulis-Kawa, Z., Majkowska-Skrobek, G., & Maciejewska, B. (2015). Bacteriophages and Phage-Derived Proteins – Application Approaches. *Current Medicinal Chemistry*, *22*(14), 1757–1773.

[4] Mohanraj, U., Wan, X., Spruit, C. M., Skurnik, M., & Pajunen, M. I. (2019). A Toxicity Screening Approach to Identify Bacteriophage-Encoded Anti-Microbial Proteins. *Viruses*, *11*(11), 1057.

[5] Liu, Y., Mi, Z., Mi, L., Huang, Y., Li, P., Liu, H., Yuan, X., Niu, W., Jiang, N., Bai, C., & Gao, Z. (2019). Identification and characterization of capsule depolymerase Dpo48 from Acinetobacter baumannii phage IME200. *PeerJ*, *7*(1), e6173.

[6] Díaz-Muñoz, S. L., & Koskella, B. (2014). Bacteria–Phage Interactions in Natural Environments. *Advances in applied microbiology* (1st ed., pp. 135–183). Elsevier Inc.

[7] Mohan Raj, J. R., & Karunasagar, I. (2019). Phages amid antimicrobial resistance. *Critical Reviews in Microbiology*, *45*(5-6), 701–711.

[8] Hill, C., Mills, S., & Ross, R. P. (2018). Phages and antibiotic resistance: are the most abundant entities on earth ready for a comeback? *Future Microbiology*, *13*(6), 711–726.

[9] Tulio Pardini G, M., Silva B, L., Aguiar A, L. A., & Elisa Soto L, M. (2017). Bacteriophage Genome Sequencing: A New Alternative to Understand Biochemical Interactions between Prokaryotic Cells and Phages. *Journal of Microbial and Biochemical Technology*, *09*(04), 169–173.

[10] ICTV. (2021). *International committee on taxonomy of viruses*. Retrieved May 11, 2021, from https://ictv.global/taxonomy/

[11] Fokine, A., & Rossmann, M. G. (2014). Molecular architecture of tailed double-stranded DNA phages. *Bacteriophage*, *4*(2), e28281.

[12] Bodier-Montagutelli, E., Morello, E., L'Hostis, G., Guillon, A., Dalloneau, E., Respaud, R., Pallaoro, N., Blois, H., Vecellio, L., Gabard, J., & Heuzé-Vourc'h, N. (2017). Inhaled phage therapy: a promising and challenging approach to treat bacterial respiratory infections. *Expert Opinion on Drug Delivery*, *14*(8), 959–972.

[13] Weiman, S. (2015). Harnessing the Power of Microbes as Therapeutics: Bugs as Drugs. *Microbe Magazine*, *10*(4), 164–164.

[14] Lawrence, Baldridge, & Handley. (2019). Phages and Human Health: More Than Idle Hitchhikers. *Viruses*, *11*(7), 587.

[15] Hobbs, Z., & Abedon, S. T. (2016). Diversity of phage infection types and associated terminology: the problem with 'Lytic or lysogenic' (A. Millard, Ed.). *FEMS Microbiology Letters*, *363*(7), fnw047.

[16] Criscuolo, E., Spadini, S., Lamanna, J., Ferro, M., & Burioni, R. (2017). Bacteriophages and Their Immunological Applications against Infectious Threats. *Journal of Immunology Research*, *2017*, 1–13.

[17] de Jonge, P. A., Nobrega, F. L., Brouns, S. J., & Dutilh, B. E. (2019). Molecular and Evolutionary Determinants of Bacteriophage Host Range. *Trends in Microbiology*, *27*(1), 51–63.

[18] Letarov, A. V., & Kulikov, E. E. (2017). Adsorption of bacteriophages on bacterial cells. *Biochemistry (Moscow)*, *82*(13), 1632–1658.

[19] Simpson, D., Sacher, J., & Szymanski, C. (2016). Development of an Assay for the Identification of Receptor Binding Proteins from Bacteriophages. *Viruses*, *8*(1), 17.

[20] Maghsoodi, A., Chatterjee, A., Andricioaei, I., & Perkins, N. C. (2019). How the phage T4 injection machinery works including energetics, forces, and dynamic pathway. *Proceedings of the National Academy of Sciences*, *116*(50), 25097–25105.

[21] Dowah, A. S. A., & Clokie, M. R. J. (2018). Review of the nature, diversity and structure of bacteriophage receptor binding proteins that target Gram-positive bacteria. *Biophysical Reviews*, *10*(2), 535–542.

[22] Domingo-Calap, P., & Delgado-Martínez, J. (2018). Bacteriophages: Protagonists of a Post-Antibiotic Era. *Antibiotics*, *7*(3), 66.

[23] Yang, Y., Shen, W., Zhong, Q., Chen, Q., He, X., Baker, J. L., Xiong, K., Jin, X., Wang, J., Hu, F., & Le, S. (2020). Development of a Bacteriophage Cocktail to Constrain the Emergence of Phage-Resistant Pseudomonas aeruginosa. *Frontiers in Microbiology*, *11*(March), 1–12.

[24] Fernandes, S., & São-José, C. (2018). Enzymes and mechanisms employed by tailed bacteriophages to breach the bacterial cell barriers. *Viruses*, *10*(8), 1–22.

[25] Labrie, S. J., Samson, J. E., & Moineau, S. (2010). Bacteriophage resistance mechanisms. *Nature Reviews Microbiology*, *8*(5), 317–327.

[26] Nazir, R., Rehman, S., Nisa, M., & ali Baba, U. (2019). Exploring bacterial diversity. *Freshwater microbiology* (pp. 263–306). Elsevier.

[27] Safari, F., Sharifi, M., Farajnia, S., Akbari, B., Karimi Baba Ahmadi, M., Negahdaripour, M., & Ghasemi, Y. (2020). The interaction of phages and bacteria: the co-evolutionary arms race. *Critical Reviews in Biotechnology*, *40*(2), 119–137.

[28] Knecht, L. E., Veljkovic, M., & Fieseler, L. (2020). Diversity and Function of Phage Encoded Depolymerases. *Frontiers in Microbiology*, *10*.

[29] Sachdeva, S., Palur, R. V., Sudhakar, K. U., & Rathinavelan, T. (2017). E. coli Group 1 Capsular Polysaccharide Exportation Nanomachinary as a Plausible Antivirulence Target in the Perspective of Emerging Antimicrobial Resistance. *Frontiers in Microbiology*, *8*(JAN), 1–19.

[30] Paton, J. C., & Trappetti, C. (2019). Streptococcus pneumoniae Capsular Polysaccharide (V. A. Fischetti, R. P. Novick, J. J. Ferretti, D. A. Portnoy, M. Braunstein, & J. I. Rood, Eds.). *Microbiology Spectrum*, *7*(2), 1–15.

[31] Wang, X., & Quinn, P. J. (2010). Lipopolysaccharide: Biosynthetic pathway and structure modification. *Progress in Lipid Research*, *49*(2), 97–107.

[32] Bertani, B., & Ruiz, N. (2018). Function and Biogenesis of Lipopolysaccharides (J. M. Slauch, Ed.). *EcoSal Plus*, *8*(1), ecosalplus.ESP–0001–2018.

[33] Simpson, B. W., May, J. M., Sherman, D. J., Kahne, D., & Ruiz, N. (2015). Lipopolysaccharide transport to the cell surface: biosynthesis and extraction from the inner membrane. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *370*(1679), 20150029.

[34] Prokhorov, N. S., Riccio, C., Zdorovenko, E. L., Shneider, M. M., Browning, C., Knirel, Y. A., Leiman, P. G., & Letarov, A. V. (2017). Function of bacteriophage G7C esterase tailspike in host cell adsorption. *Molecular Microbiology*, *105*(3), 385–398.

[35] Oliveira, H., Costa, A. R., Konstantinides, N., Ferreira, A., Akturk, E., Sillankorva, S., Nemec, A., Shneider, M., Dötsch, A., & Azeredo, J. (2017). Ability of phages to infect Acinetobacter calcoaceticus-Acinetobacter baumannii complex species through acquisition of different pectate lyase depolymerase domains. *Environmental Microbiology*, *19*(12), 5060–5077.

[36] Oliveira, H., Mendes, A., Fraga, A. G., Ferreira, A., Pimenta, A. I., Mil-Homens, D., Fialho, A. M., Pedrosa, J., & Azeredo, J. (2019). K2 Capsule Depolymerase Is Highly Stable, Is Refractory to Resistance, and Protects Larvae and Mice from Acinetobacter baumannii Sepsis (D. W. Schaffner, Ed.). *Applied and Environmental Microbiology*, *85*(17), 1–12.

[37] Bartell, P. F., Orr, T. E., & Lam, G. K. H. (1966). Polysaccharide Depolymerase Associated with Bacteriophage Infection. *Journal of Bacteriology*, *92*(1), 56–62.

[38] Majkowska-Skrobek, G., Latka, A., Berisio, R., Squeglia, F., Maciejewska, B., Briers, Y., & Drulis-Kawa, Z. (2018). Phage-Borne Depolymerases Decrease Klebsiella pneumoniae Resistance to Innate Defense Mechanisms. *Frontiers in Microbiology*, *9*(OCT), 1–12.

[39] Lin, H., Paff, M. L., Molineux, I. J., & Bull, J. J. (2017). Therapeutic Application of Phage Capsule Depolymerases against K1, K5, and K30 Capsulated E. coli in Mice. *Frontiers in Microbiology*, *8*(NOV), 1–11.

[40] Solovieva, E. V., Myakinina, V. P., Kislichkina, A. A., Krasilnikova, V. M., Verevkin, V. V., Mochalov, V. V., Lev, A. I., Fursova, N. K., & Volozhantsev, N. V. (2018). Comparative genome analysis of novel Podoviruses lytic for hypermucoviscous Klebsiella pneumoniae of K1, K2, and K57 capsular types. *Virus Research*, *243*(October), 10–18.

[41] Olszak, T., Latka, A., Roszniowski, B., Valvano, M. A., & Drulis-Kawa, Z. (2017). Phage Life Cycles Behind Bacterial Biodiversity. *Current Medicinal Chemistry*, *24*(36).

[42] Steinbacher, S., Baxa, U., Miller, S., Weintraub, A., Seckler, R., & Huber, R. (1996). Crystal structure of phage P22 tailspike protein complexed with Salmonella sp. O-antigen receptors. *Proceedings of the National Academy of Sciences*, *93*(20), 10584–10588.

[43] Olszak, T., Shneider, M. M., Latka, A., Maciejewska, B., Browning, C., Sycheva, L. V., Cornelissen, A., Danis-Wlodarczyk, K., Senchenkova, S. N., Shashkov, A. S., Gula, G., Arabski, M., Wasik, S., Miroshnikov, K. A., Lavigne, R., Leiman, P. G.,

Knirel, Y. A., & Drulis-Kawa, Z. (2017). The O-specific polysaccharide lyase from the phage LKA1 tailspike reduces Pseudomonas virulence. *Scientific Reports*, *7*(1), 16302.

[44] Nobrega, F. L., Vlot, M., de Jonge, P. A., Dreesens, L. L., Beaumont, H. J. E., Lavigne, R., Dutilh, B. E., & Brouns, S. J. J. (2018). Targeting mechanisms of tailed bacteriophages. *Nature Reviews Microbiology*, *16*(12), 760–773.

[45] Cornelissen, A., Sadovskaya, I., Vinogradov, E., Blangy, S., Spinelli, S., Casey, E., Mahony, J., Noben, J.-P., Dal Bello, F., Cambillau, C., & van Sinderen, D. (2016). The Baseplate of Lactobacillus delbrueckii Bacteriophage Ld17 Harbors a Glycerophosphodiesterase. *Journal of Biological Chemistry*, *291*(32), 16816–16827.

[46] Gutiérrez, D., Briers, Y., Rodríguez-Rubio, L., Martínez, B., Rodríguez, A., Lavigne, R., & García, P. (2015). Role of the Pre-neck Appendage Protein (Dpo7) from Phage vB_SepiS-ϕIPLA7 as an Anti-biofilm Agent in Staphylococcal Species. *Frontiers in Microbiology*, *6*(NOV), 1–10.

[47] Myers, C. L., Ireland, R. G., Garrett, T. A., & Brown, E. D. (2015). Characterization of Wall Teichoic Acid Degradation by the Bacteriophage ϕ29 Appendage Protein GP12 Using Synthetic Substrate Analogs. *Journal of Biological Chemistry*, *290*(31), 19133–19145.

[48] Leiman, P. G., Battisti, A. J., Bowman, V. D., Stummeyer, K., Mühlenhoff, M., Gerardy-Schahn, R., Scholl, D., & Molineux, I. J. (2007). The Structures of Bacteriophages K1E and K1-5 Explain Processive Degradation of Polysaccharide Capsules and Evolution of New Host Specificities. *Journal of Molecular Biology*, *371*(3), 836–849.

[49] Pan, Y.-j., Lin, T.-l., Chen, C.-c., Tsai, Y.-t., Cheng, Y.-H., Chen, Y.-Y., Hsieh, P.-F., Lin, Y.-T., & Wang, J.-T. (2017). Klebsiella Phage ΦK64-1 Encodes Multiple Depolymerases for Multiple Host Capsular Types (R. M. Sandri-Goldin, Ed.). *Journal of Virology*, *91*(6), 1–16.

[50] Latka, A., Maciejewska, B., Majkowska-Skrobek, G., Briers, Y., & Drulis-Kawa, Z. (2017). Bacteriophage-encoded virion-associated enzymes to overcome the carbohydrate barriers during the infection process. *Applied Microbiology and Biotechnology*, *101*(8), 3103–3119.

[51] Machida, Y. (2000). Structure and function of a novel coliphage-associated sialidase. *FEMS Microbiology Letters*, *182*(2), 333–337.

[52] Latka, A., Leiman, P. G., Drulis-Kawa, Z., & Briers, Y. (2019). Modeling the Architecture of Depolymerase-Containing Receptor Binding Proteins in Klebsiella Phages. *Frontiers in Microbiology*, *10*(November).

[53] Seul, A., Müller, J. J., Andres, D., Stettner, E., Heinemann, U., & Seckler, R. (2014). Bacteriophage P22 tailspike: structure of the complete protein and function of the interdomain linker. *Acta Crystallographica Section D Biological Crystallography*, *70*(5), 1336–1345.

[54] Pires, D. P., Oliveira, H., Melo, L. D. R., & Azeredo, J. (2016). Bacteriophage-encoded depolymerases: their diversity and biotechnological applications. *Applied Microbiology and Biotechnology*, *100*(5), 2141–2151.

[55] Jakobsson, E., Jokilammi, A., Aalto, J., Ollikka, P., Lehtonen, J. V., Hirvonen, H., & Finne, J. (2007). Identification of amino acid residues at the active site of endosialidase that dissociate the polysialic acid binding and cleaving activities in Escherichia coli K1 bacteriophages. *Biochemical Journal*, *405*(3), 465–472.

[56] Schwarzer, D., Browning, C., Stummeyer, K., Oberbeck, A., Mühlenhoff, M., Gerardy-Schahn, R., & Leiman, P. G. (2015). Structure and biochemical characterization of bacteriophage phi92 endosialidase. *Virology*, *477*, 133–143.

[57] Guichard, J. A., Middleton, P. C., & McConnell, M. R. (2013). Genetic analysis of structural proteins in the adsorption apparatus of bacteriophage epsilon 15. *World Journal of Virology*, *2*(4), 152.

[58] Marvasi, M., Visscher, P. T., & Casillas Martinez, L. (2010). Exopolymeric substances (EPS) from Bacillus subtilis : polymers and genes encoding their synthesis. *FEMS Microbiology Letters*, *313*(1), 1–9.

[59] Fujimoto, Z., Shiga, I., Itoh, Y., & Kimura, K. (2009). Crystallization and preliminary crystallographic analysis of poly-γ-glutamate hydrolase from bacteriophage ΦNIT1. *Acta Crystallographica Section F Structural Biology and Crystallization Communications*, *65*(9), 913–916.

[60] Martinez-Fleites, C., Smith, N. L., Turkenburg, J. P., Black, G. W., & Taylor, E. J. (2009). Structures of two truncated phage-tail hyaluronate lyases from Streptococcus pyogenes serotype M1. *Acta Crystallographica Section F Structural Biology and Crystallization Communications*, *65*(10), 963–966.

[61] Glonti, T., Chanishvili, N., & Taylor, P. (2010). Bacteriophage-derived enzyme that depolymerizes the alginic acid capsule associated with cystic fibrosis isolates of Pseudomonas aeruginosa. *Journal of Applied Microbiology*, *108*(2), 695–702.

[62] Miller, J. R., Koren, S., & Sutton, G. (2010). Assembly algorithms for next-generation sequencing data. *Genomics*, *95*(6), 315–327.

[63] Bankevich, A., Nurk, S., Antipov, D., Gurevich, A. A., Dvorkin, M., Kulikov, A. S., Lesin, V. M., Nikolenko, S. I., Pham, S., Prjibelski, A. D., Pyshkin, A. V., Sirotkin, A. V., Vyahhi, N., Tesler, G., Alekseyev, M. A., & Pevzner, P. A. (2012). SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing. *Journal of Computational Biology*, *19*(5), 455–477.

[64] Zerbino, D. R., & Birney, E. (2008). Velvet: Algorithms for de novo short read assembly using de Bruijn graphs. *Genome Research*, *18*(5), 821–829.

[65] QIAGEN. (2021, February 13). *CLC Genomics Workbench*. Retrieved February 13, 2021, from https://digitalinsights.qiagen.com/products-overview/discovery-insights-portfolio/analysis-and-visualization/qiagen-clc-genomics-workbench/

[66] Borodovsky, M., & McIninch, J. (1993). GENMARK: Parallel gene recognition for both DNA strands. *Computers and Chemistry*, *17*(2), 123–133.

[67] Delcher, A. L., Bratke, K. A., Powers, E. C., & Salzberg, S. L. (2007). Identifying bacterial genes and endosymbiont DNA with Glimmer. *Bioinformatics*, *23*(6), 673–679.

[68] Hyatt, D., Chen, G.-L., LoCascio, P. F., Land, M. L., Larimer, F. W., & Hauser, L. J. (2010). Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC Bioinformatics*, *11*(1), 119.

[69] NCBI. (2021, February 13). *BLAST: Basic Local Alignment Search Tool*. Retrieved February 13, 2021, from https://blast.ncbi.nlm.nih.gov/Blast.cgi

[70] Soding, J., Biegert, A., & Lupas, A. N. (2005). The HHpred interactive server for protein homology detection and structure prediction. *Nucleic Acids Research*, *33*(Web Server), W244–W248.

[71] Jones, P., Binns, D., Chang, H.-Y., Fraser, M., Li, W., McAnulla, C., McWilliam, H., Maslen, J., Mitchell, A., Nuka, G., Pesseat, S., Quinn, A. F., Sangrador-Vegas, A., Scheremetjew, M., Yong, S.-Y., Lopez, R., & Hunter, S. (2014). InterProScan 5: genome-scale protein function classification. *Bioinformatics*, *30*(9), 1236–1240.

[72] Aziz, R. K., Bartels, D., Best, A. A., DeJongh, M., Disz, T., Edwards, R. A., Formsma, K., Gerdes, S., Glass, E. M., Kubal, M., Meyer, F., Olsen, G. J., Olson, R., Osterman, A. L., Overbeek, R. A., McNeil, L. K., Paarmann, D., Paczian, T., Parrello, B., . . . Zagnitko, O. (2008). The RAST Server: Rapid Annotations using Subsystems Technology. *BMC Genomics*, *9*(1), 75.

[73] McNair, K., Aziz, R. K., Pusch, G. D., Overbeek, R., Dutilh, B. E., & Edwards, R. (2018). Phage Genome Annotation Using the RAST Pipeline. *Methods in molecular biology* (pp. 231–238).

[74] Meng, C., Zhang, J., Ye, X., Guo, F., & Zou, Q. (2020). Review and comparative analysis of machine learning-based phage virion protein identification methods. *Biochimica et Biophysica Acta (BBA) - Proteins and Proteomics*, *1868*(6), 140406.

[75] Kelley, L. A., Mezulis, S., Yates, C. M., Wass, M. N., & Sternberg, M. J. E. (2015). The Phyre2 web portal for protein modeling, prediction and analysis. *Nature Protocols*, *10*(6), 845–858.

[76] Bordoli, L., & Schwede, T. (2011). Automated Protein Structure Modeling with SWISS-MODEL Workspace and the Protein Model Portal. *Methods in molecular biology* (pp. 107–136).

[77] Finn, R. D., Clements, J., & Eddy, S. R. (2011). HMMER web server: interactive sequence similarity searching. *Nucleic Acids Research*, *39*(suppl), W29–W37.

[78] Mohri, M., Afshin, R., & Talwalkar, A. (2012). *Foundations of Machine Learning Regression* (Second). The MIT Pres.

[79] Parab, R. L. (2020). *Performance Evaluation Metrics for ML Models*. Retrieved February 9, 2021, from https://medium.com/swlh/performance-evaluation-metrics-for-machine-learning-models-ad0dd480d5af

[80] Brownlee, J. (2016). *Supervised and Unsupervised ML Algorithms*. Retrieved February 9, 2021, from https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/

[81] Narkhede, S. (2018). *Understanding Confusion Matrix*. Retrieved February 9, 2021, from https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

[82] Srivastava, T. (2019). *11 Important Model Evaluation Metrics for Machine Learning Everyone should know*. Retrieved February 9, 2021, from https://www.analyticsvidhya.com/blog/2019/08/11-important-model-evaluation-error-metrics/

[83] Upadhyay, A. (2020). *Precision/Recall Tradeoff*. Retrieved February 9, 2021, from https://medium.com/analytics-vidhya/precision-recall-tradeoff-79e892d43134

[84] Huilgol, P. (2020). *Precision vs. Recall – An Intuitive Guide for Every Machine Learning Person*. Retrieved February 9, 2021, from https://www.analyticsvidhya.com/blog/2020/09/precision-recall-machine-learning/

[85] Documentation, P. (2021, November 4). *Generate Kernel Density Estimate plot using Gaussian kernels*. Retrieved November 4, 2021, from https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.plot.density.html

[86] Koehrsen, W. (2018). *Histograms and Density Plots in Python*. Retrieved December 1, 2021, from https://towardsdatascience.com/histograms-and-density-plots-in-python-f6bda88f5ac0

[87] Brownlee, J. (2018). *A Gentle Introduction to k-fold Cross-Validation*. Retrieved February 9, 2021, from https://machinelearningmastery.com/k-fold-cross-validation/

[88] Anas, A.-M. (2019). *What Are Overfitting and Underfitting in Machine Learning*. Retrieved February 11, 2021, from https://towardsdatascience.com/what-are-overfitting-and-underfitting-in-machine-learning-a96b30864690

[89] Ray, S. (2017). *Commonly used Machine Learning Algorithms*. Retrieved February 18, 2021, from https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/

[90] OpenCV. (2019). *Introduction to Support Vector Machines*. Retrieved November 3, 2021, from https://docs.opencv.org/2.4/doc/tutorials/ml/introduction%5C_to%5C_svm/introduction%5C_to%5C_svm.html

[91] Patel, S. (2017). *Chapter 2 : SVM (Support Vector Machine) — Theory*. Retrieved February 11, 2021, from https://medium.com/machine-learning-101/chapter-2-svm-support-vector-machine-theory-f0812effc72

[92] Zou, J., Han, Y., & So, S.-S. (2008). Overview of Artificial Neural Networks. *Methods in molecular biology* (pp. 14–22).

[93] Coimer. (2019). *Neural Networks Components*. Retrieved November 3, 2021, from https://coimer.medium.com/neural-networks-components-a28c03d9dec

[94] Schott, M. (2019). *K-Nearest Neighbors (KNN) Algorithm for Machine Learning*. Retrieved February 11, 2021, from https://medium.com/capital-one-tech/k-nearest-neighbors-knn-algorithm-for-machine-learning-e883219c8f26

[95] Sayad, S. (2021). *Introduction to Data Science - Naive Bayesian*. Retrieved February 18, 2021, from http://www.saedsayad.com/naive%5C_bayesian.htm

[96] Liberman, N. (2017). *Decision Trees and Random Forests*. Retrieved February 11, 2021, from https://towardsdatascience.com/decision-trees-and-random-forests-df0c3123f991

[97] Afroz, C. (2019). *Random Forest Regression*. Retrieved November 3, 2021, from https://medium.com/swlh/random-forest-and-its-implementation-71824ced454f

[98] Bhatia, R. (2017). *Top 6 Regression Algorithms Used In Analytics and Data Mining*. Retrieved February 11, 2021, from https://analyticsindiamag.com/top-6-regression-algorithms-used-data-mining-applications-industry/

[99] Scikit-Learn. (2021, February 11). *User guide - scikit-learn 0.24.1 documentation*. Retrieved February 11, 2021, from https://scikit-learn.org/stable/user_guide.html

[100] Cock, P. J. A., Antao, T., Chang, J. T., Chapman, B. A., Cox, C. J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B., & de Hoon, M. J. L. (2009). Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics*, *25*(11), 1422–1423.

[101] Galaxy. (2021, February 14). *Galaxy Community Hub*. Retrieved February 14, 2021, from https://galaxyproject.org/

[102] Goecks, J., Nekrutenko, A., Taylor, J., & Galaxy Team, T. (2010). Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome Biology*, *11*(8), R86.

[103] Afgan, E., Baker, D., van den Beek, M., Blankenberg, D., Bouvier, D., Čech, M., Chilton, J., Clements, D., Coraor, N., Eberhard, C., Grüning, B., Guerler, A., Hillman-Jackson, J., Von Kuster, G., Rasche, E., Soranzo, N., Turaga, N., Taylor, J., Nekrutenko, A., & Goecks, J. (2016). The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2016 update. *Nucleic Acids Research*, *44*(W1), W3–W10.

[104] Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., & Sayers, E. W. (2011). GenBank. *Nucleic Acids Research*, *39*(Database), D32–D37.

[105] Pruitt, K., Brown, G., Tatusova, T., & Maglott, D. (2012). The NCBI Handbook. The Reference Sequence (RefSeq) Database. *https://www.ncbi.nlm.nih.gov/books/NBK21101/*, (1500).

[106] Apweiler, R. (2007). The Universal Protein Resource (UniProt). *Nucleic Acids Research*, *36*(Database), D190–D195.

[107] UniRef: Comprehensive and non-redundant UniProt reference clusters. (2007). *Bioinformatics*, *23*(10), 1282–1288.

[108] Leinonen, R., Garcia Diez, F., Binns, D., Fleischmann, W., Lopez, R., & Apweiler, R. (2004). UniProt archive. *Bioinformatics*, *20*(17), 3236–3237.

[109] Protein Data Bank: The single global archive for 3D macromolecular structure data. (2019). *Nucleic Acids Research*, *47*(D1), D520–D528.

[110] Yang, M., Derbyshire, M. K., Yamashita, R. A., & Marchler-Bauer, A. (2020). NCBI's Conserved Domain Database and Tools for Protein Domain Analysis. *Current Protocols in Bioinformatics*, *69*(1), 1–25.

[111] CDD: conserved domains and protein three-dimensional structure. (2012). *Nucleic Acids Research*, *41*(D1), D348–D352.

[112] Squeglia, F., Maciejewska, B., Łatka, A., Ruggiero, A., Briers, Y., Drulis-Kawa, Z., & Berisio, R. (2020). Structural and Functional Studies of a Klebsiella Phage Capsule Depolymerase Tailspike: Mechanistic Insights into Capsular Degradation. *Structure*, *28*(6), 613–624.e4.

[113] Stummeyer, K., Dickmanns, A., Mühlenhoff, M., Gerardy-Schahn, R., & Ficner, R. (2005). Crystal structure of the polysialic acid–degrading endosialidase of bacteriophage K1F. *Nature Structural and Molecular Biology*, *12*(1), 90–96.

[114] Kwiatkowski, B., Boschek, B., Thiele, H., & Stirm, S. (1983). Substrate specificity of two bacteriophage-associated endo-N-acetylneuraminidases. *Journal of Virology*, *45*(1), 367–374.

[115] Fu, L., Niu, B., Zhu, Z., Wu, S., & Li, W. (2012). CD-HIT: accelerated for clustering the next-generation sequencing data. *Bioinformatics*, *28*(23), 3150–3152.

[116] Dubchak, I., Muchnik, I., Holbrook, S. R., & Kim, S. H. (1995). Prediction of protein folding class using global description of amino acid sequence. *Proceedings of the National Academy of Sciences*, *92*(19), 8700–8704.

[117] Cao, D. (2021). *Propy3 version: 1.0.0a2 from PyPI*. Retrieved July 1, 2021, from https://propy3.readthedocs.io/en/latest/PyPro.html

[118] Planemo. (2014). *Building Galaxy Tools - Planemo documentation*. Retrieved November 6, 2021, from https://planemo.readthedocs.io/en/latest/writing.html

[119] Domingues, R., Barbosa, A., Santos, S. B., Pires, D. P., Save, J., Resch, G., Azeredo, J., & Oliveira, H. (2021). Unpuzzling Friunavirus-Host Interactions One Piece at a Time: Phage Recognizes Acinetobacter pittii via a New K38 Capsule Depolymerase. *Antibiotics*, *10*(11), 1304.

[120] Dunstan, R. A., Bamert, R. S., Belousoff, M. J., Short, F. L., Barlow, C. K., Pickard, D. J., Wilksch, J. J., Schittenhelm, R. B., Strugnell, R. A., Dougan, G., & Lithgow, T. (2021). Mechanistic Insights into the Capsule-Targeting Depolymerase

from a Klebsiella pneumoniae Bacteriophage (J. B. Goldberg, Ed.). *Microbiology Spectrum*, *9*(1), 1–15.

[121] Chen, C., Lewis, C. R., Goswami, K., Roberts, E. L., DebRoy, C., & Dudley, E. G. (2013). Identification and Characterization of Spontaneous Deletions within the Sp11-Sp12 Prophage Region of Escherichia coli O157:H7 Sakai. *Applied and Environmental Microbiology*, *79*(6), 1934–1941.

[122] Arndt, D., Grant, J. R., Marcu, A., Sajed, T., Pon, A., Liang, Y., & Wishart, D. S. (2016). PHASTER: a better, faster version of the PHAST phage search tool. *Nucleic Acids Research*, *44*(W1), W16–W21.

A

SUPPORT MATERIAL

A.1 SUPPLEMENTARY TABLES

Table S1: Mean PECC scores of the models after 5-fold CV for dataset d2874

Models	**PECC (W/OUT SCALER)**	**PECC (W/ SCALER)**
KNN	0,87	0,76
DT	0,86	0,86
ANN	0,88	0,93
SVM	0,81	0,93
RF	0,92	0,92
NB	0,82	0,83

Table S2: Mean PECC scores of the models after 5-fold CV for dataset d4311

Models	**PECC (W/OUT SCALER)**	**PECC (W/ SCALER)**
KNN	0,90	0,83
DT	0,89	0,90
ANN	0,89	0,94
SVM	0,84	0,94
RF	0,94	0,94
NB	0,80	0,81

Table S3: Mean PECC scores of the models after 5-fold CV for dataset d5748

Models	**PECC (W/OUT SCALER)**	**PECC (W/ SCALER)**
KNN	0,91	0,88
DT	0,90	0,91
ANN	0,94	0,95
SVM	0,86	0,95
RF	0,94	0,95
NB	0,78	0,79

Table S4: Mean PECC scores of the models after 5-fold CV for dataset d7185

Models	**PECC (W/OUT SCALER)**	**PECC (W/ SCALER)**
KNN	0,92	0,90
DT	0,92	0,91
ANN	0,92	0,96
SVM	0,89	0,96
RF	0,95	0,95
NB	0,78	0,78

Table S5: List of rank 1 features obtained from RFE using RF estimator.

Datasets	Rank 1 Features
d4311	”DNA-A”, ”DNA-T”, ”DNA-G”, ”DNA-GC”, ”AA_Len”, ”G”, ”A”, ”S”, ”T”, ”N”, ”Turn”, ”Sheet”, ”_PolarizabilityC1”, ”_PolarizabilityC3”, ”_SolventAccessibilityC1”, ”_SecondaryStrC1”, ”_SecondaryStrC2”, ”_SecondaryStrC3”, ”_ChargeC2”, ”_ChargeC3”, ”_PolarityC1”, ”_NormalizedVDWVC1”, ”_NormalizedVDWVC3”, ”_HydrophobicityC2”, ”_HydrophobicityC3”, ”_SecondaryStrT23”, ”_NormalizedVDWVT13”, ”_PolarizabilityD1001”, ”_SolventAccessibilityD1001”, ”_SolventAccessibilityD2001”, ”_SolventAccessibilityD3001”, ”_SecondaryStrD1025”, ”_ChargeD1075”, ”_ChargeD2001”, ”_ChargeD2025”, ”_ChargeD3025”, ”_ChargeD3050”, ”_PolarityD1075”, ”_PolarityD3025”, ”_NormalizedVDWVD1001”, ”_NormalizedVDWVD3050”, ”_HydrophobicityD2001”, ”DG”, ”DT”, ”GD”
d5748	”DNA-A”, ”DNA-T”, ”DNA-G”, ”DNA-GC”, ”AA_Len”, ”G”, ”A”, ”S”, ”T”, ”N”, ”Turn”, ”Sheet”, ”_PolarizabilityC1”, ”_PolarizabilityC3”, ”_SecondaryStrC1”, ”_SecondaryStrC2”, ”_SecondaryStrC3”, ”_ChargeC1”, ”_ChargeC2”, ”_ChargeC3”, ”_NormalizedVDWVC1”, ”_NormalizedVDWVC3”, ”_HydrophobicityC2”, ”_HydrophobicityC3”, ”_SolventAccessibilityT12”, ”_SolventAccessibilityT13”, ”_SecondaryStrT23”, ”_NormalizedVDWVT23”, ”_HydrophobicityT12”, ”_PolarizabilityD1001”, ”_SolventAccessibilityD1001”, ”_SolventAccessibilityD2001”, ”_SolventAccessibilityD3001”, ”_SecondaryStrD1001”, ”_SecondaryStrD1025”, ”_ChargeD1025”, ”_ChargeD1075”, ”_ChargeD2001”, ”_ChargeD2025”, ”_ChargeD3025”, ”_ChargeD3050”, ”_PolarityD1001”, ”_PolarityD1050”, ”_PolarityD1075”, ”_PolarityD3025”, ”_NormalizedVDWVD1001”, ”_NormalizedVDWVD3001”, ”_HydrophobicityD1001”, ”_HydrophobicityD2001”, ”NG”, ”DG”, ”DT”, ”GD”, ”GT”
d7185	”DNA-GC”, ”AA_Len”, ”Aromaticity”, ”IsoelectricPoint”, ”G”, ”A”, ”L”, ”V”, ”I”, ”P”, ”F”, ”S”, ”T”, ”C”, ”Y”, ”N”, ”Q”, ”D”, ”E”, ”R”, ”K”, ”H”, ”W”, ”M”, ”Turn”, ”Sheet”, ”_PolarizabilityC1”, ”_PolarizabilityC2”, ”_PolarizabilityC3”, ”_SolventAccessibilityC1”, ”_SolventAccessibilityC2”, ”_SecondaryStrC1”, ”_SecondaryStrC3”, ”_ChargeC1”, ”_ChargeC2”, ”_ChargeC3”, ”_PolarityC2”, ”_NormalizedVDWVC2”, ”_NormalizedVDWVC3”, ”_HydrophobicityC1”, ”_HydrophobicityC2”, ”_SecondaryStrT13”, ”_SecondaryStrT23”, ”_ChargeT12”, ”_ChargeT13”, ”_HydrophobicityT12”, ”_PolarizabilityD1001”, ”_PolarizabilityD1025”, ”_PolarizabilityD1050”, ”_PolarizabilityD2001”, ”_PolarizabilityD3025”, ”_PolarizabilityD3050”, ”_PolarizabilityD3075”, ”_SolventAccessibilityD1050”, ”_SolventAccessibilityD2001”, ”_SolventAccessibilityD2025”, ”_SolventAccessibilityD2050”, ”_SolventAccessibilityD3025”, ”_SolventAccessibilityD3050”, ”_SolventAccessibilityD3100”, ”_SecondaryStrD1025”, ”_SecondaryStrD1050”, ”_SecondaryStrD1075”, ”_SecondaryStrD2001”, ”_SecondaryStrD2050”, ”_SecondaryStrD2075”, ”_ChargeD1050”, ”_ChargeD1075”, ”_ChargeD1100”, ”_ChargeD2025”, ”_ChargeD3025”, ”_ChargeD3050”, ”_PolarityD2050”, ”_PolarityD3050”, ”_NormalizedVDWVD1001”, ”_NormalizedVDWVD1050”, ”_NormalizedVDWVD2001”, ”_NormalizedVDWVD2025”, ”_HydrophobicityD3001”, ”_HydrophobicityD3075”, ”AD”, ”AW”, ”AY”, ”RC”, ”RT”, ”NA”, ”NE”, ”NG”, ”NP”, ”DE”, ”DQ”, ”DG”, ”DT”, ”DY”, ”CG”, ”CL”, ”CY”, ”CV”, ”EN”, ”QA”, ”QR”, ”QE”, ”QI”, ”GA”, ”GR”, ”GD”, ”GQ”, ”GG”, ”GH”, ”GL”, ”GF”, ”GP”, ”GT”, ”GY”, ”HA”, ”HC”, ”HI”, ”HK”, ”HP”, ”IC”, ”IG”, ”IS”, ”IT”, ”IW”, ”LA”, ”LR”, ”LH”, ”LI”, ”LK”, ”LP”, ”KQ”, ”KH”, ”KS”, ”KT”, ”MQ”, ”MG”, ”MI”, ”FA”, ”FR”, ”FS”, ”FY”, ”PC”, ”PE”,

Table S6: DPO positive proteins with accession number and the corresponding phage name

Protein Accession number	Phage name
YP_008060136.1	*Acinetobacter phage AB3*
QMP19165.1	*Acinetobacter phage Ab124*
QOV07748.1	*Acinetobacter phage vB_AbaA_fBenAci001*
QOV07848.1	*Acinetobacter phage vB_AbaA_fBenAci003*
YP_009288671.1	*Acinetobacter phage phiAB6*
YP_009006536.1	*Acinetobacter phage Petty*
YP_009289769.1	*Acinetobacter phage vB_AbaS_TRS1*
YP_009189830.1	*Acinetobacter phage vB_AbaP_PD-AB9*
QQO97001.1	*Acinetobacter phage vB_AbaP_APK26*
QFG06960.1	*Acinetobacter phage vB_AbaP_APK48*
AZU99395.1	*Acinetobacter phage vB_AbaP_APK32*
QGH71569.1	*Acinetobacter phage vB_AbaP_APK48-3*
QGK90394.1	*Acinetobacter phage vB_AbaP_APK89*
QIW86364.1	*Acinetobacter virus vB_AbaP_AGC01*
QQM15083.1	*Acinetobacter phage Paty*
AZU99292.1	*Acinetobacter phage vB_AbaP_APK2-2*
QGK90444.1	*Acinetobacter phage vB_AbaP_APK44*
ALJ99087.1	*Acinetobacter phage Ab105-1phi*
YP_009610536.1	*Acinetobacter phage vB_ApiP_P2*
YP_009189380.1	*Acinetobacter phage phiAB1*
YP_009190472.1	*Acinetobacter phage vB_AbaP_PD-6A3*
YP_009949058.1	*Acinetobacter phage SWH-Ab-1*
YP_009599281.1	*Acinetobacter phage vB_AbaP_AS11*
QHS01530.1	*Acinetobacter phage vB_AbaP_APK116*
AYR04394.1	*Acinetobacter phage vB_AbaP_APK14*
YP_009203055.1	*Acinetobacter phage Fri1*
QGF20174.1	*Acinetobacter phage vB_AbaP_PMK34*
QOV07800.1	*Acinetobacter phage vB_AbaA_fBenAci002*
QQO92973.1	*Acinetobacter phage Pipo*
YP_009814060.1	*Acinetobacter phage vB_AbaP_B09_Aci08*
QNO11418.1	*Acinetobacter phage vB_AbaP_APK81*

Table S6 continued from previous page

Protein Accession number	Phage name
YP_009813438.1	*Acinetobacter phage vB_AbaP_46-62_Aci07*
QGK90498.1	*Acinetobacter phage vB_AbaP_APK87*
YP_009103257.1	*Acinetobacter phage vB_AbaP_Acibel007*
QAU04146.1	*Acinetobacter phage AbTJ*
YP_009216837.1	*Acinetobacter phage phiAC-1*
AUG85465.1	*Acinetobacter phage SH-Ab 15497*
QOI69765.1	*Acinetobacter phage DMU1*
QNO11465.1	*Acinetobacter phage Aristophanes*
AYP68982.1	*Acinetobacter phage vB_AbaM_IME284*
QGH74055.1	*Acinetobacter phage Bphi-R2919*
QGH74134.1	*Acinetobacter phage Bphi-R1888*
QEA11050.1	*Acinetobacter phage Abp9*
YP_009291902.1	*Acinetobacter phage LZ35*
YP_009613841.1	*Acinetobacter phage AB1*
YP_009609870.1	*Acinetobacter phage AbP2*
YP_009604496.1	*Acinetobacter phage WCHABP1*
YP_006383804.1	*Acinetobacter phage AP22*
AYP69084.1	*Acinetobacter phage vB_AbaM_IME512*
YP_009146765.1	*Acinetobacter phage YMC13/03/R2096*
YP_003347555.1	*Klebsiella phage KP32*
YP_003347556.1	*Klebsiella phage KP32*
AWN07125.1	*Klebsiella phage KP32_isolate 194*
AWN07126.1	*Klebsiella phage KP32_isolate 194*
AOT28172.1	*Klebsiella phage vB_KpnP_KpV763*
AOT28173.1	*Klebsiella phage vB_KpnP_KpV763*
AWN07083.1	*Klebsiella phage KP32_isolate 192*
AWN07084.1	*Klebsiella phage KP32_isolate 192*
APZ82804.1	*Klebsiella phage K5-2*
APZ82805.1	*Klebsiella phage K5-2*
AWN07213.1	*Klebsiella phage KP32_isolate 196*
AWN07214.1	*Klebsiella phage KP32_isolate 196*

Table S6 continued from previous page

Protein Accession number	Phage name
AOZ65569.1	*Klebsiella phage vB_KpnP_KpV766*
YP_009215499.1	*Klebsiella phage vB_KpnP_KpV289*
YP_009215498.1	*Klebsiella phage vB_KpnP_KpV289*
ALT58498.1	*Klebsiella phage vB_KpnP_IME205*
APZ82847.1	*Klebsiella phage K5-4*
APZ82848.1	*Klebsiella phage K5-4*
YP_009280720.1	*Klebsiella phage KpV475*
YP_009302756.1	*Klebsiella phage KpV71*
YP_009098385.1	*Klebsiella phage NTUH-K2044-K1-1*
YP_009188797.1	*Klebsiella phage KpV41*
BBF66844.1	*Klebsiella phage KN1-1*
BBF66888.1	*Klebsiella phage KN4-1*
AZS06408.1	*Klebsiella phage Henu1*
QAU05545.1	*Klebsiella phage Kund-ULIP47*
QAU05505.1	*Klebsiella phage K1-ULIP33*
QBG78385.1	*Klebsiella phage Kund-ULIP54*
QGZ00758.1	*Klebsiella phage VLC1*
QGZ00819.1	*Klebsiella phage VLC2*
QGZ00875.1	*Klebsiella phage VLC3*
QGZ00936.1	*Klebsiella phage VLC4*
QIW86419.1	*Klebsiella phage VLC5*
QIW86428.1	*Klebsiella phage VLC5*
QJI52623.1	*Klebsiella phage VLC6*
QJI52632.1	*Klebsiella phage VLC6*
QMP82097.1	*Klebsiella virus KpV2883*
QMP82089.1	*Klebsiella virus KpV2883*
QOI68577.1	*Klebsiella phage vB_KpnP_Dlv622*
QOI68629.1	*Klebsiella phage vB_KpnM_Seu621*
AOZ65386.1	*Klebsiella phage vB_KpnM_KpV52*
YP_009153196.1	*Klebsiella phage K64-1*
YP_009153203.1	*Klebsiella phage K64-1*

Table S6 continued from previous page

Protein Accession number	Phage name
QIW88225.1	*Klebsiella phage KpS8*
YP_002003831.1	*Klebsiella phage K11*
YP_009198668.1	*Klebsiella phage K5*
YP_009198669.1	*Klebsiella phage K5*
AUV61507.1	*Klebsiella phage SH-Kp 152410*
ARB12452.1	*Klebsiella phage vB_KpnP_IL33*
ARB12406.1	*Klebsiella phage vB_KpnP_PRA33*
ARB12500.1	*Klebsiella phage vB_KpnP_BIS33*
AXE28435.1	*Klebsiella phage vB_KpnP_IME321*
ASZ78307.1	*Klebsiella phage 2044-307w*
YP_003347651.1	*Klebsiella phage KP34*
YP_009199937.1	*Klebsiella phage vB_KpnP_SU503*
ASV44946.1	*Klebsiella phage AltoGao*
YP_009204835.1	*Klebsiella phage vB_KpnP_SU552A*
AWK24039.1	*Klebsiella phage phiKpS2*
ARM70347.1	*Klebsiella phage KOX1*
AVI03134.1	*Klebsiella phage JY917*
ASV44964.1	*Klebsiella phage MezzoGao*
AXF39389.1	*Klebsiella phage NJS1*
AUE22051.1	*Klebsiella phage GML-KpCol1*
ASW27458.1	*Klebsiella phage KPN N141*
YP_009197879.1	*Klebsiella phage 1513*
AUV59228.1	*Klebsiella phage vB_KpnM_KpS110*
AUV59229.1	*Klebsiella phage vB_KpnM_KpS110*
AUV59230.1	*Klebsiella phage vB_KpnM_KpS110*
AUV59234.1	*Klebsiella phage vB_KpnM_KpS110*
YP_009796379.1	*Klebsiella phage Menlow*
AUG87748.1	*Klebsiella phage Menlow*
AUG87751.1	*Klebsiella phage Menlow*
AUG87753.1	*Klebsiella phage Menlow*
AUG87958.1	*Klebsiella phage May*

Table S6 continued from previous page

Protein Accession number	Phage name
AUG87959.1	*Klebsiella phage May*
AUG87960.1	*Klebsiella phage May*
AUG87962.1	*Klebsiella phage May*
YP_007007253.1	*Klebsiella phage vB_KleM_RaK2*
YP_007007685.1	*Klebsiella phage vB_KleM_RaK2*
YP_007007686.1	*Klebsiella phage vB_KleM_RaK2*
YP_007007687.1	*Klebsiella phage vB_KleM_RaK2*
AWN07172.1	*Klebsiella phage KP32_isolate 195*
YP_008532046.1	*Klebsiella virus 0507KN21*
YP_008532047.1	*Klebsiella virus 0507KN21*
YP_008532048.1	*Klebsiella virus 0507KN21*
YP_008532049.1	*Klebsiella virus 0507KN21*
YP_008532051.1	*Klebsiella virus 0507KN21*
YP_008532050.1	*Klebsiella virus 0507KN21*
AZF89844.1	*Klebsiella phage 13*
QEQ50396.1	*Klebsiella phage vB_KpnP_IME337*
QKY78353.1	*Klebsiella phage P509*
YP_654147.1	*Escherichia virus K1-5*
YP_004678762.1	*Escherichia phage K30*
BAW85696.1	*Klebsiella phage K64-1*
BAW85697.1	*Klebsiella phage K64-1*
BAQ02780.1	*Klebsiella phage K64-1*
BAW85692.1	*Klebsiella phage K64-1*
BAW85695.1	*Klebsiella phage K64-1*
APW79830.1	*Acinetobacter phage vB_AbaP_AS12*
NP_112090.1	*Enterobacteria phage HK620*
NP_059644.1	*Salmonella virus P22*
NP_853609.1	*Salmonella virus SP6*
NP_853610.1	*Salmonella virus SP6*
AIB07058.1	*Salmonella phage 9NA*
YP_009140380.1	*Salmonella phage Det7*

Table S6 continued from previous page

Protein Accession number	Phage name
NP_848228.1	*Salmonella phage epsilon15*
YP_008126828.1	*Vibrio phage JA-1*

Table S7: DPO negative proteins with accession number and the corresponding phage name. All proteins were obtained from *Escherichia virus T4*.

Protein accession number	Phage name
NP_049616.1	*Escherichia virus T4*
NP_049617.1	*Escherichia virus T4*
NP_049618.1	*Escherichia virus T4*
NP_049619.1	*Escherichia virus T4*
NP_049620.1	*Escherichia virus T4*
NP_049621.1	*Escherichia virus T4*
NP_049622.1	*Escherichia virus T4*
NP_049623.1	*Escherichia virus T4*
NP_049624.1	*Escherichia virus T4*
NP_049625.1	*Escherichia virus T4*
NP_049626.1	*Escherichia virus T4*
NP_049627.1	*Escherichia virus T4*
NP_049628.1	*Escherichia virus T4*
NP_049629.1	*Escherichia virus T4*
NP_049630.1	*Escherichia virus T4*
NP_049631.1	*Escherichia virus T4*
NP_049632.1	*Escherichia virus T4*
NP_049633.1	*Escherichia virus T4*
NP_049634.1	*Escherichia virus T4*
NP_049635.1	*Escherichia virus T4*
NP_049636.1	*Escherichia virus T4*
NP_049638.1	*Escherichia virus T4*
NP_049639.1	*Escherichia virus T4*

Table S7 continued from previous page

Accession number	Phage name
NP_049640.1	*Escherichia virus T4*
NP_049641.1	*Escherichia virus T4*
NP_049642.1	*Escherichia virus T4*
NP_049643.1	*Escherichia virus T4*
NP_049644.1	*Escherichia virus T4*
NP_049645.1	*Escherichia virus T4*
NP_049646.1	*Escherichia virus T4*
NP_049647.1	*Escherichia virus T4*
NP_049648.1	*Escherichia virus T4*
NP_049649.1	*Escherichia virus T4*
NP_049650.1	*Escherichia virus T4*
NP_049651.1	*Escherichia virus T4*
NP_049652.1	*Escherichia virus T4*
NP_049653.1	*Escherichia virus T4*
NP_049654.1	*Escherichia virus T4*
NP_049655.1	*Escherichia virus T4*
NP_049656.2	*Escherichia virus T4*
NP_049657.1	*Escherichia virus T4*
NP_049658.1	*Escherichia virus T4*
NP_049659.1	*Escherichia virus T4*
NP_049660.1	*Escherichia virus T4*
NP_049661.1	*Escherichia virus T4*
NP_049662.1	*Escherichia virus T4*
NP_049663.1	*Escherichia virus T4*
NP_049664.1	*Escherichia virus T4*
NP_049665.1	*Escherichia virus T4*
NP_049666.1	*Escherichia virus T4*
NP_049667.1	*Escherichia virus T4*
NP_049668.1	*Escherichia virus T4*
NP_049669.1	*Escherichia virus T4*
NP_049670.1	*Escherichia virus T4*

Table S7 continued from previous page

Accession number	Phage name
NP_049671.1	*Escherichia virus T4*
NP_049672.1	*Escherichia virus T4*
NP_813808.1	*Escherichia virus T4*
NP_049673.1	*Escherichia virus T4*
NP_049674.1	*Escherichia virus T4*
NP_049675.1	*Escherichia virus T4*
NP_049676.1	*Escherichia virus T4*
NP_049677.1	*Escherichia virus T4*
NP_049678.1	*Escherichia virus T4*
NP_049679.1	*Escherichia virus T4*
NP_049680.1	*Escherichia virus T4*
NP_049681.1	*Escherichia virus T4*
NP_049682.1	*Escherichia virus T4*
NP_049683.1	*Escherichia virus T4*
NP_049684.1	*Escherichia virus T4*
NP_049685.1	*Escherichia virus T4*
NP_049686.1	*Escherichia virus T4*
NP_049687.1	*Escherichia virus T4*
NP_049688.1	*Escherichia virus T4*
NP_049689.1	*Escherichia virus T4*
NP_049691.2	*Escherichia virus T4*
NP_049690.1	*Escherichia virus T4*
NP_049692.1	*Escherichia virus T4*
NP_049693.2	*Escherichia virus T4*
NP_049694.1	*Escherichia virus T4*
NP_049695.1	*Escherichia virus T4*
NP_049696.1	*Escherichia virus T4*
NP_049697.1	*Escherichia virus T4*
NP_049698.1	*Escherichia virus T4*
NP_049699.1	*Escherichia virus T4*
NP_049700.1	*Escherichia virus T4*

Table S7 continued from previous page

Accession number	Phage name
NP_049701.1	*Escherichia virus T4*
NP_049702.1	*Escherichia virus T4*
NP_049703.1	*Escherichia virus T4*
NP_049704.1	*Escherichia virus T4*
NP_049705.1	*Escherichia virus T4*
NP_049706.1	*Escherichia virus T4*
NP_049707.1	*Escherichia virus T4*
NP_049710.1	*Escherichia virus T4*
NP_049711.1	*Escherichia virus T4*
NP_049712.1	*Escherichia virus T4*
NP_813809.1	*Escherichia virus T4*
NP_049713.1	*Escherichia virus T4*
NP_049714.1	*Escherichia virus T4*
NP_049715.1	*Escherichia virus T4*
NP_049716.1	*Escherichia virus T4*
NP_049717.1	*Escherichia virus T4*
NP_049718.1	*Escherichia virus T4*
NP_049719.1	*Escherichia virus T4*
NP_049721.1	*Escherichia virus T4*
NP_049722.1	*Escherichia virus T4*
NP_049723.1	*Escherichia virus T4*
NP_049724.1	*Escherichia virus T4*
NP_049725.1	*Escherichia virus T4*
NP_049726.1	*Escherichia virus T4*
NP_049727.1	*Escherichia virus T4*
NP_049728.1	*Escherichia virus T4*
NP_049729.1	*Escherichia virus T4*
NP_049730.1	*Escherichia virus T4*
NP_049731.1	*Escherichia virus T4*
NP_049732.1	*Escherichia virus T4*
NP_049733.1	*Escherichia virus T4*

Table S7 continued from previous page

Accession number	Phage name
NP_049734.1	*Escherichia virus T4*
NP_049735.1	*Escherichia virus T4*
NP_049736.1	*Escherichia virus T4*
NP_049737.1	*Escherichia virus T4*
NP_049738.1	*Escherichia virus T4*
NP_049739.1	*Escherichia virus T4*
NP_049740.1	*Escherichia virus T4*
NP_049741.1	*Escherichia virus T4*
NP_049742.1	*Escherichia virus T4*
NP_049743.1	*Escherichia virus T4*
NP_049744.1	*Escherichia virus T4*
NP_049745.2	*Escherichia virus T4*
NP_049746.1	*Escherichia virus T4*
NP_049747.1	*Escherichia virus T4*
NP_049748.1	*Escherichia virus T4*
NP_049749.1	*Escherichia virus T4*
NP_049750.1	*Escherichia virus T4*
NP_049751.1	*Escherichia virus T4*
NP_049752.1	*Escherichia virus T4*
NP_049753.1	*Escherichia virus T4*
NP_049754.1	*Escherichia virus T4*
NP_049755.1	*Escherichia virus T4*
NP_049756.1	*Escherichia virus T4*
NP_049757.1	*Escherichia virus T4*
NP_049758.1	*Escherichia virus T4*
NP_049759.1	*Escherichia virus T4*
NP_049760.1	*Escherichia virus T4*
NP_049761.1	*Escherichia virus T4*
NP_049762.1	*Escherichia virus T4*
NP_049763.1	*Escherichia virus T4*
NP_049764.1	*Escherichia virus T4*

Table S7 continued from previous page

Accession number	Phage name
NP_049765.1	*Escherichia virus T4*
NP_049766.1	*Escherichia virus T4*
NP_049767.1	*Escherichia virus T4*
NP_049771.1	*Escherichia virus T4*
NP_049775.1	*Escherichia virus T4*
NP_049783.1	*Escherichia virus T4*
NP_049860.1	*Escherichia virus T4*
NP_049861.1	*Escherichia virus T4*
NP_049862.1	*Escherichia virus T4*
NP_049863.1	*Escherichia virus T4*

Table S8: DPO positive proteins with accession number and the corresponding DPO percentage for the models SVM and ANN.

Protein Accession number	SVM	ANN
YP_008060136.1	99.0	100.0
QMP19165.1	99.0	100.0
QOV07748.1	98.0	100.0
QOV07848.1	100.0	100.0
YP_009288671.1	100.0	100.0
YP_009006536.1	98.0	100.0
YP_009289769.1	100.0	100.0
YP_009189830.1	98.0	100.0
QQO97001.1	100.0	100.0
QFG06960.1	100.0	100.0
AZU99395.1	99.0	100.0
QGH71569.1	97.0	100.0
QGK90394.1	98.0	100.0
QIW86364.1	100.0	100.0
QQM15083.1	99.0	100.0

Table S8 continued from previous page

Protein Accession number	SVM	ANN
AZU99292.1	99.0	100.0
QGK90444.1	98.0	100.0
ALJ99087.1	99.0	100.0
YP_009610536.1	100.0	100.0
YP_009189380.1	99.0	100.0
YP_009190472.1	69.0	100.0
YP_009949058.1	58.0	100.0
YP_009599281.1	98.0	100.0
QHS01530.1	99.0	100.0
AYR04394.1	90.0	98.0
YP_009203055.1	96.0	100.0
QGF20174.1	100.0	100.0
QOV07800.1	100.0	100.0
QQO92973.1	99.0	100.0
YP_009814060.1	94.0	100.0
QNO11418.1	98.0	100.0
YP_009813438.1	98.0	88.0
QGK90498.1	100.0	100.0
YP_009103257.1	100.0	100.0
QAU04146.1	100.0	99.0
YP_009216837.1	99.0	97.0
AUG85465.1	99.0	100.0
QOI69765.1	99.0	100.0
QNO11465.1	82.0	100.0
AYP68982.1	100.0	95.0
QGH74055.1	99.0	100.0
QGH74134.1	99.0	100.0
QEA11050.1	100.0	99.0
YP_009291902.1	100.0	100.0
YP_009613841.1	100.0	100.0
YP_009609870.1	100.0	100.0

Table S8 continued from previous page

Protein Accession number	SVM	ANN
YP_009604496.1	100.0	99.0
YP_006383804.1	100.0	100.0
AYP69084.1	100.0	100.0
YP_009146765.1	100.0	100.0
YP_003347555.1	99.0	100.0
YP_003347556.1	100.0	100.0
AWN07125.1	100.0	100.0
AWN07126.1	100.0	100.0
AOT28172.1	100.0	100.0
AOT28173.1	100.0	100.0
AWN07083.1	100.0	100.0
AWN07084.1	99.0	100.0
APZ82804.1	100.0	100.0
APZ82805.1	100.0	100.0
AWN07213.1	99.0	100.0
AWN07214.1	100.0	100.0
AOZ65569.1	99.0	100.0
YP_009215499.1	97.0	100.0
YP_009215498.1	100.0	100.0
ALT58498.1	99.0	100.0
APZ82847.1	98.0	100.0
APZ82848.1	100.0	100.0
YP_009280720.1	100.0	98.0
YP_009302756.1	99.0	74.0
YP_009098385.1	99.0	98.0
YP_009188797.1	99.0	91.0
BBF66844.1	100.0	100.0
BBF66888.1	95.0	100.0
AZS06408.1	99.0	99.0
QAU05545.1	100.0	100.0
QAU05505.1	100.0	100.0

Table S8 continued from previous page

Protein Accession number	SVM	ANN
QBG78385.1	100.0	100.0
QGZ00758.1	99.0	100.0
QGZ00819.1	98.0	100.0
QGZ00875.1	98.0	100.0
QGZ00936.1	99.0	100.0
QIW86419.1	99.0	100.0
QIW86428.1	100.0	100.0
QJI52623.1	99.0	100.0
QJI52632.1	99.0	100.0
QMP82097.1	99.0	90.0
QMP82089.1	100.0	100.0
QOI68577.1	100.0	100.0
QOI68629.1	100.0	100.0
AOZ65386.1	100.0	100.0
YP_009153196.1	100.0	100.0
YP_009153203.1	100.0	100.0
QIW88225.1	100.0	100.0
YP_002003831.1	100.0	100.0
YP_009198668.1	99.0	100.0
YP_009198669.1	100.0	100.0
AUV61507.1	100.0	100.0
ARB12452.1	99.0	100.0
ARB12406.1	99.0	100.0
ARB12500.1	99.0	100.0
AXE28435.1	100.0	100.0
ASZ78307.1	36.0	0.0
YP_003347651.1	100.0	100.0
YP_009199937.1	100.0	100.0
ASV44946.1	100.0	100.0
YP_009204835.1	100.0	100.0
AWK24039.1	100.0	100.0

Table S8 continued from previous page

Protein Accession number	SVM	ANN
ARM70347.1	100.0	100.0
AVI03134.1	100.0	100.0
ASV44964.1	68.0	100.0
AXF39389.1	73.0	100.0
AUE22051.1	60.0	100.0
ASW27458.1	49.0	100.0
YP_009197879.1	100.0	100.0
AUV59228.1	80.0	78.0
AUV59229.1	100.0	100.0
AUV59230.1	100.0	100.0
AUV59234.1	100.0	100.0
YP_009796379.1	100.0	100.0
AUG87748.1	98.0	100.0
AUG87751.1	99.0	100.0
AUG87753.1	96.0	100.0
AUG87958.1	97.0	100.0
AUG87959.1	99.0	100.0
AUG87960.1	99.0	99.0
AUG87962.1	99.0	100.0
YP_007007253.1	82.0	96.0
YP_007007685.1	98.0	100.0
YP_007007686.1	96.0	93.0
YP_007007687.1	100.0	100.0
AWN07172.1	100.0	100.0
YP_008532046.1	94.0	100.0
YP_008532047.1	99.0	100.0
YP_008532048.1	100.0	100.0
YP_008532049.1	99.0	99.0
YP_008532051.1	99.0	37.0
YP_008532050.1	100.0	100.0
AZF89844.1	100.0	100.0

Table S8 continued from previous page

Protein Accession number	SVM	ANN
QEQ50396.1	98.0	100.0
QKY78353.1	100.0	100.0
YP_654147.1	39.0	100.0
YP_004678762.1	100.0	100.0
BAW85696.1	99.0	100.0
BAW85697.1	99.0	100.0
BAQ02780.1	39.0	100.0
BAW85692.1	100.0	100.0
BAW85695.1	3.0	92.0
APW79830.1	98.0	100.0
NP_112090.1	100.0	100.0
NP_059644.1	100.0	100.0
NP_853609.1	99.0	97.0
NP_853610.1	99.0	100.0
AIB07058.1	100.0	100.0
YP_009140380.1	100.0	100.0
NP_848228.1	98.0	100.0
YP_008126828.1	99.0	100.0

Table S9: DPO negative proteins with accession number and the corresponding DPO percentage for the models SVM and ANN.

Protein Accession number	SVM	ANN
NP_049616.1	1.0	0.0
NP_049617.1	4.0	0.0
NP_049618.1	1.0	0.0
NP_049619.1	0.0	0.0
NP_049620.1	6.0	0.0
NP_049621.1	5.0	0.0
NP_049622.1	5.0	0.0
NP_049623.1	5.0	0.0
NP_049624.1	3.0	0.0

Table S9 continued from previous page

Protein Accession number	SVM	ANN
NP_049625.1	3.0	0.0
NP_049626.1	1.0	0.0
NP_049627.1	1.0	0.0
NP_049628.1	0.0	0.0
NP_049629.1	0.0	0.0
NP_049630.1	1.0	0.0
NP_049631.1	6.0	0.0
NP_049632.1	1.0	0.0
NP_049633.1	2.0	0.0
NP_049634.1	1.0	0.0
NP_049635.1	0.0	0.0
NP_049636.1	1.0	0.0
NP_049638.1	1.0	0.0
NP_049639.1	6.0	0.0
NP_049640.1	5.0	0.0
NP_049641.1	0.0	0.0
NP_049642.1	6.0	0.0
NP_049643.1	3.0	0.0
NP_049644.1	1.0	0.0
NP_049645.1	1.0	0.0
NP_049646.1	1.0	0.0
NP_049647.1	1.0	0.0
NP_049648.1	3.0	0.0
NP_049649.1	6.0	0.0
NP_049650.1	1.0	0.0
NP_049651.1	1.0	0.0
NP_049652.1	3.0	0.0
NP_049653.1	5.0	0.0
NP_049654.1	2.0	0.0
NP_049655.1	4.0	0.0
NP_049656.2	0.0	0.0

Table S9 continued from previous page

Protein Accession number	SVM	ANN
NP_049657.1	1.0	0.0
NP_049658.1	1.0	0.0
NP_049659.1	1.0	0.0
NP_049660.1	5.0	0.0
NP_049661.1	2.0	0.0
NP_049662.1	0.0	0.0
NP_049663.1	1.0	0.0
NP_049664.1	0.0	0.0
NP_049665.1	0.0	0.0
NP_049666.1	1.0	0.0
NP_049667.1	1.0	0.0
NP_049668.1	2.0	0.0
NP_049669.1	4.0	0.0
NP_049670.1	6.0	0.0
NP_049671.1	3.0	0.0
NP_049672.1	2.0	0.0
NP_813808.1	3.0	0.0
NP_049673.1	0.0	0.0
NP_049674.1	1.0	0.0
NP_049675.1	2.0	0.0
NP_049676.1	3.0	0.0
NP_049677.1	2.0	0.0
NP_049678.1	3.0	0.0
NP_049679.1	0.0	0.0
NP_049680.1	6.0	0.0
NP_049681.1	4.0	0.0
NP_049682.1	3.0	0.0
NP_049683.1	3.0	0.0
NP_049684.1	5.0	0.0
NP_049685.1	4.0	0.0
NP_049686.1	6.0	0.0

Table S9 continued from previous page

Protein Accession number	SVM	ANN
NP_049687.1	4.0	0.0
NP_049688.1	2.0	0.0
NP_049689.1	2.0	0.0
NP_049691.2	1.0	0.0
NP_049690.1	0.0	0.0
NP_049692.1	1.0	0.0
NP_049693.2	6.0	0.0
NP_049694.1	1.0	0.0
NP_049695.1	6.0	0.0
NP_049696.1	2.0	0.0
NP_049697.1	4.0	0.0
NP_049698.1	1.0	0.0
NP_049699.1	4.0	0.0
NP_049700.1	2.0	0.0
NP_049701.1	1.0	0.0
NP_049702.1	3.0	0.0
NP_049703.1	0.0	0.0
NP_049704.1	0.0	0.0
NP_049705.1	3.0	0.0
NP_049706.1	2.0	0.0
NP_049707.1	2.0	0.0
NP_049710.1	0.0	0.0
NP_049711.1	3.0	0.0
NP_049712.1	6.0	0.0
NP_813809.1	6.0	0.0
NP_049713.1	3.0	0.0
NP_049714.1	3.0	0.0
NP_049715.1	3.0	0.0
NP_049716.1	0.0	0.0
NP_049717.1	2.0	0.0
NP_049718.1	6.0	0.0

Table S9 continued from previous page

Protein Accession number	SVM	ANN
NP_049719.1	2.0	0.0
NP_049721.1	5.0	0.0
NP_049722.1	4.0	0.0
NP_049723.1	3.0	0.0
NP_049724.1	4.0	0.0
NP_049725.1	3.0	0.0
NP_049726.1	1.0	0.0
NP_049727.1	3.0	0.0
NP_049728.1	2.0	0.0
NP_049729.1	6.0	0.0
NP_049730.1	1.0	0.0
NP_049731.1	1.0	0.0
NP_049732.1	1.0	0.0
NP_049733.1	1.0	0.0
NP_049734.1	5.0	0.0
NP_049735.1	5.0	0.0
NP_049736.1	1.0	0.0
NP_049737.1	0.0	0.0
NP_049738.1	3.0	0.0
NP_049739.1	4.0	0.0
NP_049740.1	3.0	0.0
NP_049741.1	0.0	0.0
NP_049742.1	2.0	0.0
NP_049743.1	2.0	0.0
NP_049744.1	0.0	0.0
NP_049745.2	1.0	0.0
NP_049746.1	0.0	0.0
NP_049747.1	3.0	0.0
NP_049748.1	3.0	0.0
NP_049749.1	3.0	0.0
NP_049750.1	0.0	0.0

Table S9 continued from previous page

Protein Accession number	SVM	ANN
NP_049751.1	4.0	0.0
NP_049752.1	0.0	0.0
NP_049753.1	47.0	0.0
NP_049754.1	2.0	0.0
NP_049755.1	1.0	0.0
NP_049756.1	1.0	0.0
NP_049757.1	5.0	0.0
NP_049758.1	6.0	0.0
NP_049759.1	6.0	0.0
NP_049760.1	20.0	0.0
NP_049761.1	1.0	0.0
NP_049762.1	6.0	0.0
NP_049763.1	3.0	0.0
NP_049764.1	3.0	0.0
NP_049765.1	2.0	0.0
NP_049766.1	0.0	0.0
NP_049767.1	4.0	0.0
NP_049771.1	5.0	6.0
NP_049775.1	0.0	0.0
NP_049783.1	4.0	0.0
NP_049860.1	7.0	1.0
NP_049861.1	6.0	0.0
NP_049862.1	6.0	1.0
NP_049863.1	16.0	97.0

Table S10: CDS list from *Acinetobacter phage vB_Api_3043-K38*, as obtained from NCBI, and the predicted probability of each CDS being a DPO. Includes the predictions of the SVM model and ANN model.

Protein Accession number	SVM	ANN
QYC50686.1	6.0	0.0
QYC50685.1	5.0	0.0

Table S10 continued from previous page

Protein Accession number	SVM	ANN
QYC50678.1	5.0	0.0
QYC50677.1	4.0	0.0
QYC50660.1	5.0	0.0
QYC50671.1	7.0	0.0
QYC50665.1	2.0	0.0
QYC50668.1	5.0	0.0
QYC50657.1	2.0	0.0
QYC50687.1	6.0	0.0
QYC50664.1	8.0	0.0
QYC50661.1	4.0	0.0
QYC50663.1	1.0	0.0
QYC50688.1	6.0	47.0
QYC50682.1	6.0	0.0
QYC50675.1	6.0	0.0
QYC50676.1	6.0	0.0
QYC50652.1	0.0	0.0
QYC50669.1	10.0	0.0
QYC50673.1	5.0	0.0
QYC50645.1	0.0	0.0
QYC50653.1	9.0	53.0
QYC50647.1	1.0	0.0
QYC50640.1	6.0	0.0
QYC50674.1	4.0	0.0
QYC50650.1	2.0	0.0
QYC50666.1	1.0	0.0
QYC50649.1	0.0	0.0
QYC50658.1	1.0	0.0
QYC50662.1	4.0	0.0
QYC50648.1	0.0	0.0
QYC50654.1	0.0	0.0
QYC50639.1	0.0	0.0

Table S10 continued from previous page

Protein Accession number	SVM	ANN
QYC50681.1	3.0	0.0
QYC50672.1	6.0	0.0
QYC50644.1	3.0	0.0
QYC50651.1	25.0	0.0
QYC50646.1	45.0	10.0
QYC50684.1	6.0	0.0
QYC50679.1	6.0	0.0
QYC50656.1	0.0	0.0
QYC50641.1	10.0	2.0
QYC50655.1	67.0	0.0
QYC50638.1	17.0	1.0
QYC50637.1	8.0	1.0
QYC50642.1	99.0	100.0
QYC50667.1	4.0	0.0
QYC50659.1	4.0	6.0
QYC50670.1	12.0	0.0
QYC50643.1	2.0	16.0
QYC50689.1	6.0	0.0
QYC50680.1	3.0	0.0
QYC50683.1	6.0	0.0

Table S11: CDS list from *Klebsiella phage RAD2*, as obtained from NCBI, and the predicted probability of each CDS being a DPO. Includes the predictions of the SVM model and ANN model.

Protein Accession number	SVM	ANN
YP_010115728.1	81.0	78.0
YP_010115729.1	99.0	100.0
YP_010115730.1	4.0	0.0
YP_010115731.1	1.0	0.0

Table S11 continued from previous page

Protein Accession number	SVM	ANN
YP_010115732.1	6.0	0.0
YP_010115733.1	2.0	0.0
YP_010115734.1	5.0	0.0
YP_010115735.1	1.0	12.0
YP_010115736.1	6.0	0.0
YP_010115737.1	1.0	0.0
YP_010115738.1	2.0	0.0
YP_010115739.1	3.0	0.0
YP_010115740.1	7.0	0.0
YP_010115741.1	2.0	0.0
YP_010115742.1	5.0	0.0
YP_010115743.1	2.0	33.0
YP_010115744.1	1.0	0.0
YP_010115745.1	5.0	20.0
YP_010115746.1	18.0	0.0
YP_010115747.1	8.0	0.0
YP_010115748.1	3.0	0.0
YP_010115749.1	2.0	0.0
YP_010115750.1	8.0	0.0
YP_010115751.1	3.0	0.0
YP_010115752.1	6.0	0.0
YP_010115753.1	0.0	0.0
YP_010115754.1	6.0	0.0
YP_010115755.1	1.0	0.0
YP_010115756.1	5.0	0.0
YP_010115757.1	4.0	0.0
YP_010115758.1	2.0	0.0
YP_010115759.1	6.0	0.0
YP_010115760.1	4.0	0.0
YP_010115761.1	5.0	0.0
YP_010115762.1	7.0	0.0

Table S11 continued from previous page

Protein Accession number	SVM	ANN
YP_010115763.1	6.0	0.0
YP_010115764.1	3.0	0.0
YP_010115765.1	3.0	0.0
YP_010115766.1	6.0	0.0
YP_010115767.1	2.0	0.0
YP_010115768.1	3.0	0.0
YP_010115769.1	5.0	0.0
YP_010115770.1	7.0	0.0
YP_010115771.1	5.0	0.0
YP_010115772.1	1.0	0.0
YP_010115773.1	6.0	0.0
YP_010115774.1	5.0	0.0
YP_010115775.1	3.0	0.0
YP_010115776.1	4.0	0.0
YP_010115777.1	6.0	0.0
YP_010115778.1	6.0	0.0
YP_010115779.1	6.0	7.0
YP_010115780.1	19.0	0.0
YP_010115781.1	1.0	0.0
YP_010115782.1	5.0	0.0
YP_010115783.1	6.0	0.0
YP_010115784.1	6.0	0.0
YP_010115785.1	2.0	0.0
YP_010115786.1	2.0	0.0
YP_010115787.1	7.0	0.0
YP_010115788.1	3.0	0.0
YP_010115789.1	1.0	0.0
YP_010115790.1	59.0	8.0
YP_010115791.1	6.0	0.0
YP_010115792.1	2.0	0.0
YP_010115793.1	13.0	0.0

Table S11 continued from previous page

Protein Accession number	SVM	ANN
YP_010115794.1	5.0	0.0
YP_010115795.1	47.0	0.0
YP_010115796.1	1.0	0.0
YP_010115797.1	25.0	49.0
YP_010115798.1	7.0	0.0
YP_010115799.1	7.0	1.0
YP_010115800.1	12.0	0.0
YP_010115801.1	78.0	4.0
YP_010115802.1	2.0	0.0
YP_010115803.1	5.0	0.0

Table S12: CDS list from *Pseudomonas Phage LUZ19*, as obtained from NCBI, and the predicted probability of each CDS being a DPO. Includes both predictions of the SVM model and ANN model.

Protein Accession number	SVM	ANN
YP_001671942.1	6.0	0.0
YP_001671943.1	4.0	0.0
YP_001671944.1	5.0	0.0
YP_001671945.1	2.0	3.0
YP_001671946.1	18.0	0.0
YP_001671947.1	3.0	0.0
YP_001671948.1	6.0	0.0
YP_001671949.1	9.0	0.0
YP_001671950.1	6.0	0.0
YP_001671951.1	4.0	0.0
YP_001671952.1	3.0	1.0
YP_001671953.1	9.0	0.0
YP_001671954.1	3.0	0.0
YP_001671955.1	2.0	0.0

Table S12 continued from previous page

Protein Accession number	SVM	ANN
YP_001671956.1	6.0	0.0
YP_001671957.1	6.0	0.0
YP_001671958.1	4.0	0.0
YP_001671959.1	9.0	0.0
YP_001671960.1	3.0	0.0
YP_001671961.1	0.0	0.0
YP_001671962.1	4.0	0.0
YP_001671963.1	2.0	0.0
YP_001671964.1	6.0	0.0
YP_001671965.1	50.0	2.0
YP_001671966.1	6.0	0.0
YP_001671967.1	11.0	0.0
YP_001671968.1	1.0	0.0
YP_001671969.1	11.0	0.0
YP_001671970.1	6.0	0.0
YP_001671971.1	14.0	0.0
YP_001671972.1	4.0	0.0
YP_001671973.1	2.0	0.0
YP_001671974.1	6.0	0.0
YP_001671975.1	10.0	0.0
YP_001671976.1	12.0	0.0
YP_001671977.1	17.0	0.0
YP_001671978.1	17.0	0.0
YP_001671979.1	1.0	28.0
YP_001671980.1	7.0	0.0
YP_001671981.1	4.0	0.0
YP_001671982.1	6.0	0.0
YP_001671983.1	10.0	0.0
YP_001671984.1	6.0	0.0
YP_001671985.1	96.0	16.0
YP_001671986.1	2.0	0.0

Table S12 continued from previous page

Protein Accession number	SVM	ANN
YP_001671987.1	4.0	0.0
YP_001671988.1	33.0	2.0
YP_001671989.1	6.0	0.0
YP_001671990.1	9.0	3.0
YP_001671991.1	2.0	0.0
YP_001671992.1	6.0	0.0
YP_001671993.1	34.0	0.0
YP_001671994.1	6.0	0.0
YP_001671995.1	3.0	0.0

Table S13: CDS list from *Escherichia phage vB_EcoP_G7C*, as obtained from NCBI, and the predicted probability of each CDS being a DPO. Includes both predictions of the SVM model and ANN model.

Protein Accession number	SVM	ANN
YP_004782125.1	4.0	0.0
YP_004782126.1	6.0	0.0
YP_004782127.1	1.0	0.0
YP_004782128.1	1.0	17.0
YP_004782129.1	0.0	0.0
YP_004782130.1	5.0	0.0
YP_004782131.1	6.0	0.0
YP_004782132.1	6.0	0.0
YP_004782133.1	6.0	0.0
YP_004782134.1	6.0	0.0
YP_004782135.1	6.0	0.0
YP_004782136.1	6.0	0.0
YP_004782137.1	6.0	0.0
YP_004782138.1	4.0	0.0
YP_004782139.1	7.0	0.0

Table S13 continued from previous page

Protein Accession number	SVM	ANN
YP_004782140.1	6.0	0.0
YP_004782141.1	0.0	0.0
YP_004782142.1	1.0	0.0
YP_004782143.1	2.0	90.0
YP_004782144.1	6.0	0.0
YP_004782145.1	62.0	11.0
YP_004782146.1	5.0	0.0
YP_004782147.1	6.0	0.0
YP_004782148.1	16.0	0.0
YP_004782149.1	2.0	0.0
YP_004782150.1	1.0	0.0
YP_004782151.1	0.0	0.0
YP_004782152.1	1.0	0.0
YP_004782153.1	7.0	0.0
YP_004782154.1	6.0	0.0
YP_004782155.1	2.0	0.0
YP_004782156.1	2.0	0.0
YP_004782157.1	6.0	0.0
YP_004782158.1	6.0	0.0
YP_004782159.1	7.0	0.0
YP_004782160.1	6.0	0.0
YP_004782161.1	6.0	0.0
YP_004782162.1	0.0	0.0
YP_004782163.1	3.0	0.0
YP_004782164.1	2.0	0.0
YP_004782165.1	7.0	0.0
YP_004782166.1	14.0	18.0
YP_004782167.1	1.0	0.0
YP_004782168.1	1.0	0.0
YP_004782169.1	5.0	0.0
YP_004782170.1	2.0	0.0

Table S13 continued from previous page

Protein Accession number	SVM	ANN
YP_004782171.1	1.0	0.0
YP_004782172.1	2.0	0.0
YP_004782173.1	0.0	1.0
YP_004782174.1	0.0	0.0
YP_004782175.1	1.0	3.0
YP_004782176.1	5.0	0.0
YP_004782177.1	6.0	0.0
YP_004782178.1	6.0	0.0
YP_004782179.1	4.0	0.0
YP_004782180.1	5.0	0.0
YP_004782181.1	31.0	17.0
YP_004782182.1	6.0	0.0
YP_004782183.1	67.0	0.0
YP_004782184.1	13.0	0.0
YP_004782185.1	0.0	0.0
YP_004782186.1	28.0	1.0
YP_004782187.1	8.0	0.0
YP_004782188.1	12.0	0.0
YP_004782189.1	15.0	0.0
YP_004782190.1	4.0	0.0
YP_004782191.1	8.0	0.0
YP_004782192.1	2.0	0.0
YP_004782193.1	5.0	0.0
YP_004782194.1	1.0	0.0
YP_004782195.1	94.0	92.0
YP_004782196.1	100.0	100.0
YP_004782197.1	3.0	0.0
YP_004782198.1	0.0	0.0
YP_004782199.1	12.0	0.0
YP_004782200.1	2.0	0.0
YP_004782201.1	1.0	0.0

Table S13 continued from previous page

Protein Accession number	SVM	ANN
YP_004782202.1	5.0	0.0
YP_004782203.1	5.0	0.0

Table S14: Prophage protein list from *Acinetobacter baumannii strain A85*, within region 3477508-3510350, as obtained from PHASTER, and the predicted probability of each protein being a DPO. Includes the predictions of the SVM model and ANN model.

Protein identifier and Description	SVM	ANN
ASF78650.1_VirulencesensorproteinBvgSprecursor	0.0	0.0
ASF78651.1_Phageportalprotein	0.0	0.0
ASF78652.1_Terminase-likefamilyprotein	8.0	0.0
ASF78653.1_Phagecapsidscaffoldingprotein(GPO)serinepeptidase	1.0	0.0
ASF78654.1_Phagemajorcapsidprotein,P2family	18.0	12.0
ASF78655.1_Phagesmallterminasesubunit	1.0	0.0
ASF78656.1_Phageheadcompletionprotein(GPL)	10.0	0.0
ASF78657.1_PhageTailProteinX	4.0	0.0
ASF78658.1_hypotheticalprotein	7.0	0.0
ASF78659.1_hypotheticalprotein	6.0	0.0
ASF78660.1_Putativepeptidoglycanbindingdomainprotein	1.0	0.0
ASF78661.1_P2phagetailcompletionproteinR(GpR)	0.0	0.0
ASF78662.1_Phagevirionmorphogenesisfamilyprotein	3.0	0.0
ASF78663.1_Phage-relatedbaseplateassemblyprotein	23.0	67.0
ASF78664.1_Gene25-likelysozyme	5.0	0.0
ASF78665.1_BaseplateJ-likeprotein	23.0	0.0
ASF78666.1_Phagetailprotein(Tail_P2_I)	2.0	0.0
ASF78667.1_hypotheticalprotein	99.0	97.0
ASF78668.1_Phagetailsheathprotein	0.0	0.0
ASF78669.1_PhagetailtubeproteinFII	11.0	0.0
ASF78670.1_PhagetailproteinE	4.0	0.0
ASF78671.1_Phage-relatedminortailprotein	5.0	0.0

Table S14 continued from previous page

Protein identifier and Description	SVM	ANN
ASF78672.1_PhageP2GpU	3.0	0.0
ASF78673.1_PhagelatecontrolgeneDprotein(GPD)	42.0	85.0
ASF78674.1_Ogr/Delta-likezincfinger	7.0	0.0
ASF78675.1_hypotheticalprotein	3.0	0.0
ASF78676.1_hypotheticalprotein	4.0	0.0
ASF78677.1_DNApolymeraseIIIsubunitepsilon	0.0	0.0
ASF78678.1_hypotheticalprotein	3.0	0.0
ASF78679.1_PhageregulatoryproteinRha(Phage_pRha)	19.0	0.0
ASF78680.1_hypotheticalprotein	3.0	0.0
ASF78681.1_Helix-turn-helixdomainprotein	1.0	0.0
ASF78682.1_hypotheticalprotein	6.0	0.0
ASF78683.1_hypotheticalprotein	7.0	1.0
ASF78684.1_hypotheticalprotein	1.0	1.0
ASF78685.1_hypotheticalprotein	0.0	1.0
ASF78686.1_hypotheticalprotein	5.0	0.0
ASF78687.1_hypotheticalprotein	8.0	0.0
ASF78688.1_Single-strandedDNA-bindingprotein	2.0	0.0
ASF78689.1_Exodeoxyribonuclease10	16.0	0.0
ASF78690.1_hypotheticalprotein	2.0	0.0
ASF78691.1_hypotheticalprotein	2.0	0.0

Table S15: Prophage protein list from *Acinetobacter baumannii ATCC 19606*, within region 78042-120394, as obtained from PHASTER, and the predicted probability of each protein being a DPO. Includes the predictions of the SVM model and ANN model.

Protein identifier and Description	SVM	ANN
ENW74124.1_hypotheticalprotein	1.0	0.0
ENW74125.1_hypotheticalprotein	1.0	0.0
ENW74126.1_hypotheticalprotein	0.0	0.0
ENW74127.1_hypotheticalprotein	2.0	0.0

Table S15 continued from previous page

Protein identifier and Description	SVM	ANN
ENW74128.1_hypotheticalprotein	5.0	0.0
ENW74129.1_hypotheticalprotein	6.0	0.0
ENW74130.1_hypotheticalprotein	5.0	0.0
ENW74131.1_hypotheticalprotein	62.0	76.0
ENW74132.1_hypotheticalprotein	5.0	0.0
ENW74133.1_hypotheticalprotein	24.0	99.0
ENW74134.1_hypotheticalprotein	49.0	99.0
ENW74135.1_hypotheticalprotein	16.0	1.0
ENW74136.1_hypotheticalprotein	17.0	0.0
ENW74137.1_hypotheticalprotein	3.0	0.0
ENW74138.1_hypotheticalprotein	15.0	0.0
ENW74139.1_hypotheticalprotein	5.0	0.0
ENW74140.1_hypotheticalprotein	3.0	0.0
ENW74141.1_hypotheticalprotein	53.0	0.0
ENW74142.1_hypotheticalprotein	4.0	0.0
ENW74143.1_hypotheticalprotein	7.0	0.0
ENW74144.1_hypotheticalprotein	2.0	0.0
ENW74145.1_hypotheticalprotein	1.0	0.0
ENW74146.1_hypotheticalprotein	4.0	0.0
ENW74147.1_hypotheticalprotein	4.0	2.0
ENW74148.1_hypotheticalprotein	14.0	85.0
ENW74149.1_hypotheticalprotein	4.0	100.0
ENW74150.1_hypotheticalprotein	2.0	0.0
ENW74151.1_hypotheticalprotein	6.0	0.0
ENW74152.1_hypotheticalprotein	2.0	0.0
ENW74153.1_hypotheticalprotein	5.0	0.0
ENW74154.1_HI1409familyphage-associatedprotein	0.0	0.0
ENW74155.1_hypotheticalprotein	0.0	0.0
ENW74156.1_hypotheticalprotein	1.0	0.0
ENW74157.1_hypotheticalprotein	1.0	0.0
ENW74158.1_hypotheticalprotein	1.0	0.0

Table S15 continued from previous page

Protein identifier and Description	SVM	ANN
ENW74159.1_hypotheticalprotein	2.0	0.0
ENW74160.1_hypotheticalprotein	6.0	2.0
ENW74161.1_hypotheticalprotein	5.0	0.0
ENW74162.1_hypotheticalprotein	1.0	0.0
ENW74163.1_hypotheticalprotein	1.0	0.0
ENW74164.1_hypotheticalprotein	5.0	0.0
ENW74165.1_hypotheticalprotein	3.0	0.0
ENW74166.1_hypotheticalprotein	2.0	0.0
ENW74167.1_hypotheticalprotein	2.0	0.0
ENW74168.1_hypotheticalprotein	6.0	0.0
ENW74169.1_hypotheticalprotein	21.0	2.0
ENW74170.1_hypotheticalprotein	1.0	0.0
ENW74171.1_hypotheticalprotein	3.0	4.0
ENW74172.1_hypotheticalprotein	3.0	0.0
ENW74173.1_hypotheticalprotein	1.0	0.0
ENW74174.1_hypotheticalprotein	5.0	0.0
ENW74175.1_hypotheticalprotein	6.0	0.0
ENW74176.1_hypotheticalprotein	3.0	0.0
ENW74177.1_hypotheticalprotein	1.0	0.0
ENW74178.1_hypotheticalprotein	5.0	0.0
ENW74179.1_hypotheticalprotein	4.0	0.0
ENW74180.1_hypotheticalprotein	6.0	0.0
ENW74181.1_hypotheticalprotein	3.0	0.0
ENW74182.1_hypotheticalprotein	1.0	0.0
ENW74183.1_hypotheticalprotein	6.0	0.0

Table S16: Prophage protein list from *Acinetobacter baumannii ATCC 19606*, within region 274341-319584, as obtained from PHASTER, and the predicted probability of each protein being a DPO. Includes the predictions of the SVM model and ANN model.

Protein identifier and Description	SVM	ANN
ENW74322.1_hypotheticalprotein	6.0	0.0
ENW74323.1_hypotheticalprotein	5.0	0.0
ENW74324.1_hypotheticalprotein	91.0	1.0
ENW74325.1_hypotheticalprotein	2.0	0.0
ENW74326.1_hypotheticalprotein	1.0	0.0
ENW74327.1_hypotheticalprotein	5.0	0.0
ENW74328.1_hypotheticalprotein	2.0	0.0
ENW74329.1_hypotheticalprotein	4.0	0.0
ENW74330.1_hypotheticalprotein	8.0	0.0
ENW74331.1_hypotheticalprotein	19.0	5.0
ENW74332.1_hypotheticalprotein	2.0	0.0
ENW74333.1_hypotheticalprotein	5.0	0.0
ENW74334.1_hypotheticalprotein	4.0	0.0
ENW74335.1_hypotheticalprotein	57.0	83.0
ENW74336.1_hypotheticalprotein	88.0	8.0
ENW74337.1_hypotheticalprotein	67.0	62.0
ENW74338.1_hypotheticalprotein	2.0	0.0
ENW74339.1_hypotheticalprotein	4.0	0.0
ENW74340.1_hypotheticalprotein	18.0	1.0
ENW74341.1_hypotheticalprotein	20.0	0.0
ENW74342.1_hypotheticalprotein	2.0	0.0
ENW74343.1_hypotheticalprotein	2.0	0.0
ENW74344.1_hypotheticalprotein	4.0	0.0
ENW74345.1_hypotheticalprotein	7.0	0.0
ENW74346.1_hypotheticalprotein	3.0	0.0
ENW74347.1_hypotheticalprotein	3.0	0.0
ENW74348.1_hypotheticalprotein	10.0	10.0
ENW74349.1_hypotheticalprotein	2.0	0.0
ENW74350.1_hypotheticalprotein	3.0	0.0

Table S16 continued from previous page

Protein identifier and Description	SVM	ANN
ENW74351.1_hypotheticalprotein	6.0	0.0
ENW74352.1_hypotheticalprotein	4.0	0.0
ENW74353.1_hypotheticalprotein	1.0	0.0
ENW74354.1_hypotheticalprotein	1.0	0.0
ENW74355.1_hypotheticalprotein	6.0	0.0
ENW74356.1_hypotheticalprotein	3.0	0.0
ENW74357.1_hypotheticalprotein	0.0	0.0
ENW74358.1_hypotheticalprotein	2.0	0.0
ENW74359.1_hypotheticalprotein	0.0	0.0
ENW74360.1_hypotheticalprotein	3.0	0.0
ENW74361.1_hypotheticalprotein	6.0	0.0
ENW74362.1_hypotheticalprotein	3.0	0.0
ENW74363.1_hypotheticalprotein	4.0	5.0
ENW74364.1_hypotheticalprotein	3.0	0.0
ENW74365.1_hypotheticalprotein	4.0	0.0
ENW74366.1_hypotheticalprotein	1.0	0.0
ENW74367.1_hypotheticalprotein	6.0	0.0
ENW74368.1_hypotheticalprotein	2.0	0.0
ENW74369.1_hypotheticalprotein	1.0	0.0
ENW74370.1_hypotheticalprotein	1.0	0.0
ENW74371.1_hypotheticalprotein	22.0	11.0
ENW74372.1_hypotheticalprotein	6.0	0.0
ENW74373.1_hypotheticalprotein	3.0	0.0
ENW74374.1_hypotheticalprotein	1.0	0.0
ENW74375.1_hypotheticalprotein	9.0	23.0
ENW74376.1_hypotheticalprotein	5.0	0.0
ENW74377.1_hypotheticalprotein	10.0	0.0
ENW74378.1_hypotheticalprotein	4.0	0.0
ENW74379.1_hypotheticalprotein	5.0	0.0
ENW74380.1_hypotheticalprotein	3.0	0.0
ENW74381.1_hypotheticalprotein	25.0	0.0

Table S16 continued from previous page

Protein identifier and Description	SVM	ANN
ENW74382.1_hypotheticalprotein	3.0	0.0
ENW74383.1_hypotheticalprotein	5.0	0.0
ENW74384.1_hypotheticalprotein	1.0	0.0
ENW74385.1_hypotheticalprotein	1.0	0.0
ENW74386.1_hypotheticalprotein	6.0	0.0
ENW74387.1_hypotheticalprotein	39.0	94.0
ENW74388.1_hypotheticalprotein	1.0	0.0
ENW74389.1_hypotheticalprotein	3.0	0.0

A.2 Supplementary figures

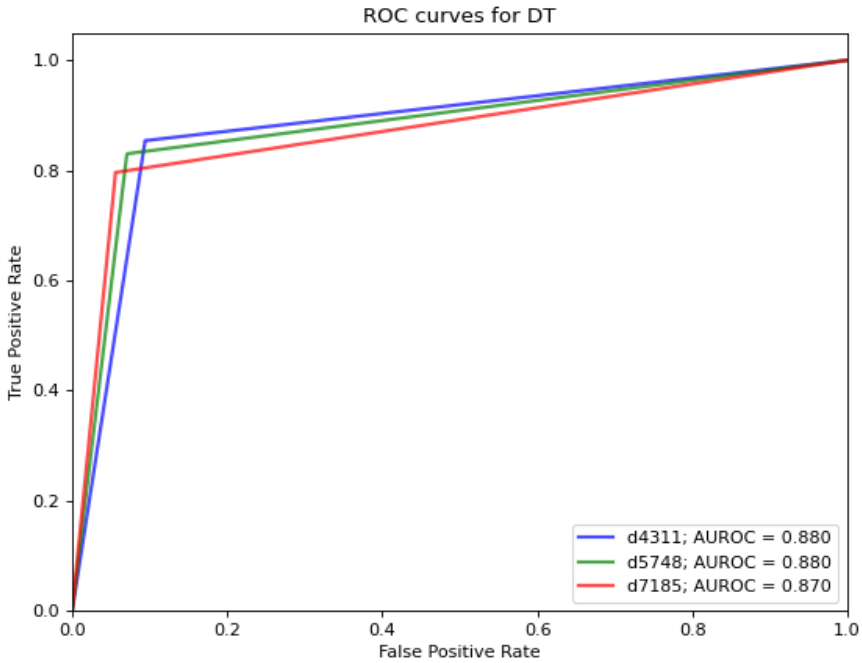


Figure S1: ROC curves representing the DT model performance for all three datasets with corresponding AUROC value.

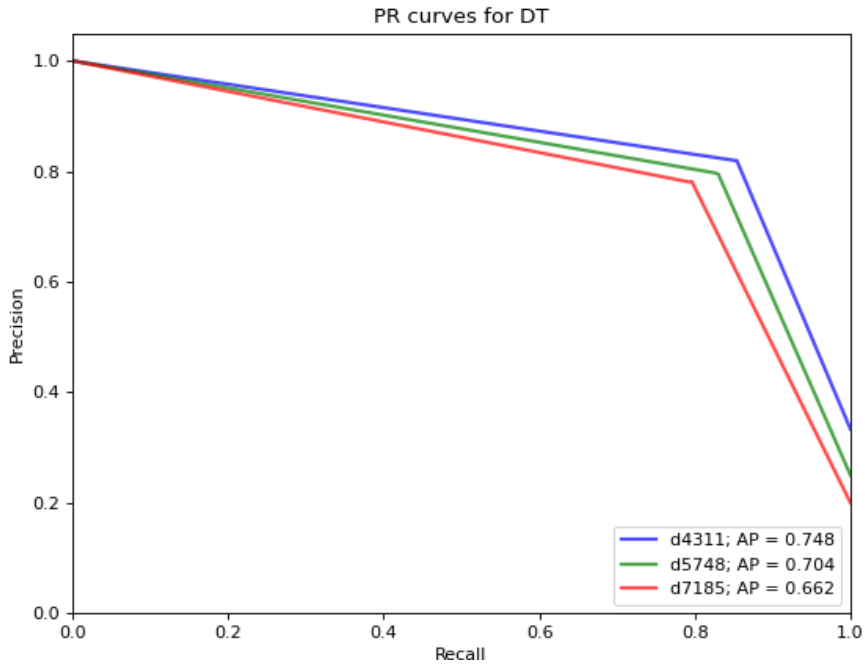


Figure S2: PR curves representing the DT model performance for all three datasets with corresponding AP value.

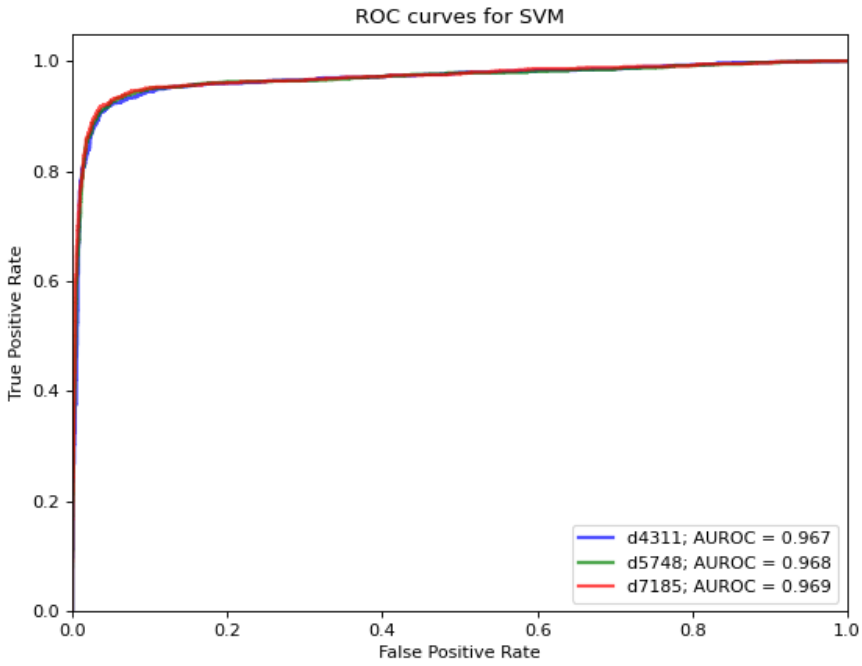


Figure S3: ROC curves representing the SVM model performance for all three datasets with corresponding AUROC value.

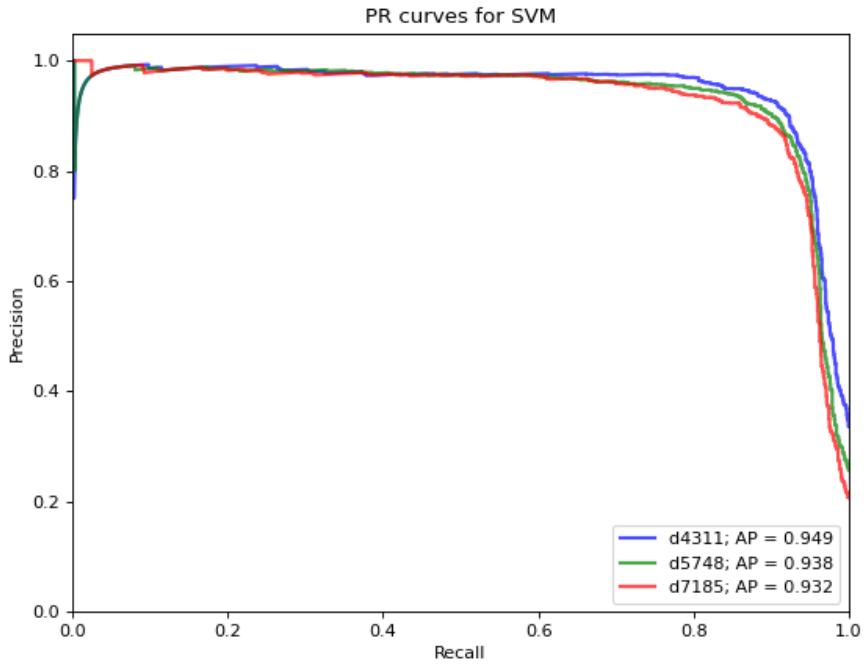


Figure S4: PR curves representing the SVM model performance for all three datasets with corresponding AP value.

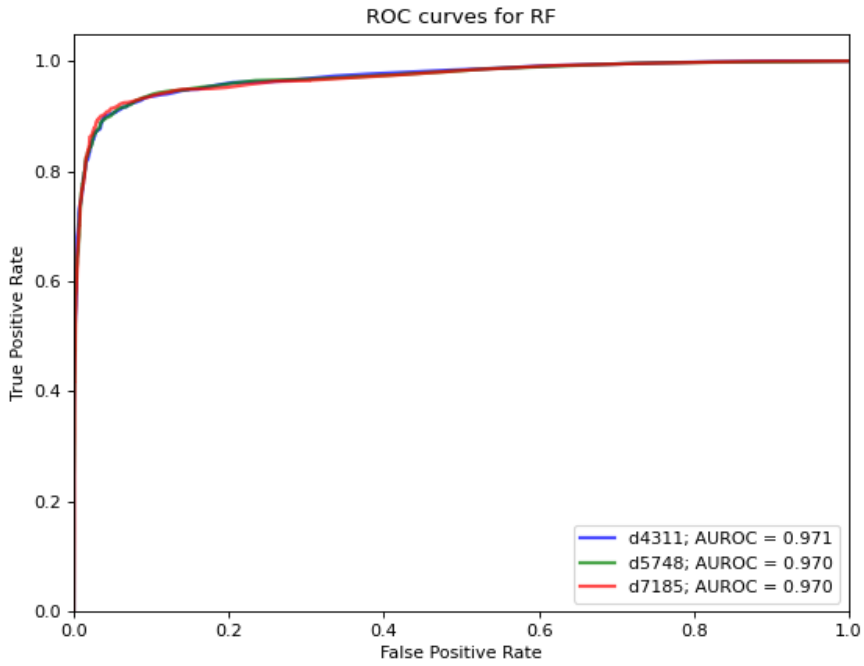


Figure S5: ROC curves representing the RF model performance for all three datasets with corresponding AUROC value.

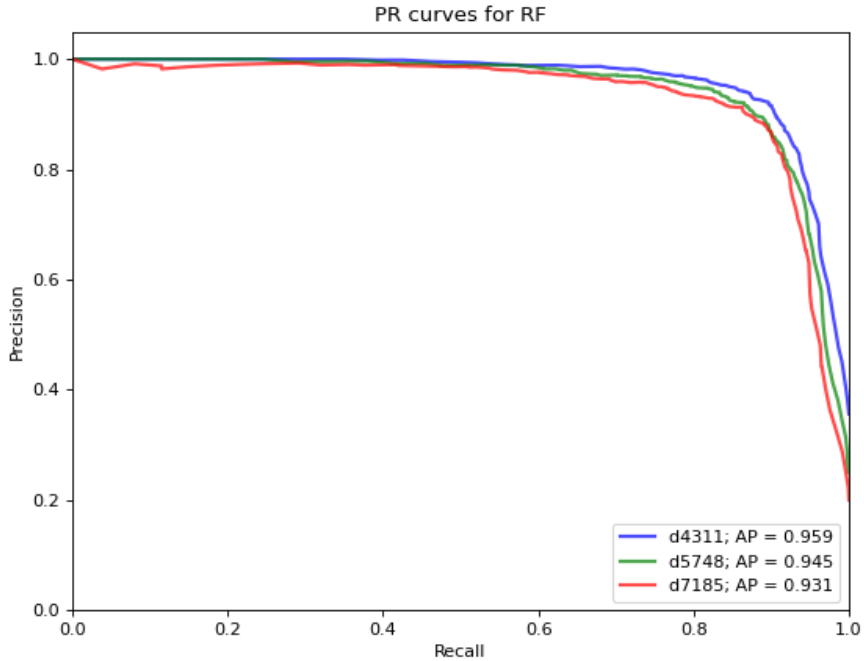


Figure S6: PR curves representing the RF model performance for all three datasets with corresponding AP value.