

BEN-GURION UNIVERSITY OF THE NEGEV  
THE FACULTY OF NATURAL SCIENCES  
DEPARTMENT OF COMPUTER SCIENCE

# Bacterial Pathogenicity Classification Via Sparse SVM

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS  
FOR THE MASTER OF SCIENCES DEGREE

WRITTEN BY: Eran Barash

UNDER THE SUPERVISION OF: Prof. Michal Ziv-Ukelson and Dr. Sivan Sabato

September 2017

BEN-GURION UNIVERSITY OF THE NEGEV  
THE FACULTY OF NATURAL SCIENCES  
DEPARTMENT OF COMPUTER SCIENCE

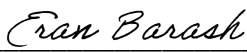
# Bacterial Pathogenicity Classification Via Sparse SVM

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS  
FOR THE MASTER OF SCIENCES DEGREE

WRITTEN BY: Eran Barash

UNDER THE SUPERVISION OF: Prof. Michal Ziv-Ukelson and Dr. Sivan Sabato

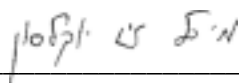
Signatures:

  
\_\_\_\_\_

Author: Eran Barash

12/9/17  
\_\_\_\_\_

Date

  
\_\_\_\_\_

Advisor: Prof. Michal Ziv-Ukelson

12/9/17  
\_\_\_\_\_

Date

  
\_\_\_\_\_

Advisor: Dr. Sivan Sabato

12/9/17  
\_\_\_\_\_

Date

\_\_\_\_\_

Dept. Committee Chairperson

\_\_\_\_\_

Date

September 2017

## Abstract

**Motivation:** Bacterial infections are a major cause of illness worldwide. However, many bacterial strains pose no threat to human health, and some are even beneficial to it. Thus, effective bioinformatic tools, differentiating pathogenic from commensal bacteria, are of great importance for understanding bacterial infections and for rapid diagnosis. Continuous development of gene sequencing technologies leads to increasing availability of sequenced bacterial genomes and motivates tools for classifying the pathogenicity of a human hosted bacteria based on its genome.

**Results:** We propose a machine-learning based approach for classifying a given human-hosted bacterial genome as pathogenic to human or not. Our approach is based on Sparse SVM, which efficiently selects a small set of genes that are related to bacterial pathogenicity without manual intervention. We implement our approach as a tool, denoted “Bacterial Pathogenicity Classification via Sparse SVM” (BACPACSS). Our tool is fully automated and our approach scales up to a dataset significantly larger than those used by previous approaches to related problems.

To design a tool clinically relevant to human, we created a dataset consisting of only human-hosted bacteria. Using this data for training, testing and model generation, BACPACSS shows a high accuracy in distinguishing pathogenic from non-pathogenic human-hosted bacteria. Moreover, using this tool we were able to detect genes that are important for bacterial pathogenicity, some of which are with unknown function.

**Availability:** The annotated dataset used, the BACPACSS code, the resulting model and additional supplementary materials are available in:

<https://www.cs.bgu.ac.il/~barashe/>

## Table of Contents

Abstract .....	1
Table of Contents .....	1
1 Introduction.....	2
2 Related work .....	6
3 Our Contribution .....	9
4 Methods .....	12

4.1	Extracting features .....	12
4.2	Training the Classification Model .....	13
4.3	Scoring .....	15
5	Dataset .....	16
6	Results and Discussion .....	17
6.1	Quantitative results .....	18
6.2	Biological results .....	21
7	Conclusions and future work .....	24
8	Supplementary material .....	25
9	Bibliography .....	25
Table 1 - Summary of ML Accuracy results with different measures. ....		18
Table 2 - Comparison of different sizes of gene lists used to train our model. ....		20
Table 3 - Top 25 PFs and their functional categories, sorted in decreasing weight order. Hosts were determined using all members of each PF. Some PFs were species specific, and some broader. The weight column in the table denotes the weights assigned to the corresponding feature by the classifier; the higher the weight, the more meaningful the feature. ....		23
Figure 1 - Classification workflow. Training steps are outlined in red, and prediction steps are outlined in black. Input cells are green and output cell is orange. ....		15
Figure 2 - F1-macro score vs number of trained features. a. Each fold's F1-macro score. b. Overall mean of all 10 folds. ....		19

# 1 Introduction

Infectious diseases are one of the biggest threats to public health, despite the advance of modern medicine in the post-genomic era. For example, in 2011, the CDC estimated that each year, 1 out of 6 Americans (48 million people) contract foodborne infections,

among them 128,000 are hospitalized, and 3,000 die<sup>1</sup>. In the same report, it was shown that only 20% of infections are caused by known pathogens, whereas 80% are caused by unidentified pathogens. Out of all foodborne related deaths, a majority of cases (56%) are due to an unidentified pathogen.

Not all bacteria are pathogenic to humans and many of them are innocuous or even beneficial to human health. These include thousands of different microbial species that reside in a healthy human gut, and are important for nutrition, development, and regulation of the immune response<sup>2,3</sup>. As the core microbiota of humans is largely diverse, the determination of whether a specific bacterial strain is commensal or pathogenic is challenging. In addition, commensal human bacteria could evolve into pathogenic bacteria by acquisition of novel genes via horizontal gene transfer (HGT) mechanism<sup>4,5</sup>. This complicates clinical diagnosis done using traditional methods and encourages using complete genome sequencing to reveal new genetic traits. It also motivates the need to identify human-pathogenic (HP) strains, and to understand their virulence mechanisms; such studies could facilitate the identification of contaminated food, increase the chances of correct infection diagnosis, provide better patient treatment, and lead to better development of targeted drugs and vaccines.

In current clinical practices, determination of an infection agent is based on Koch's postulates, established in the 19<sup>th</sup> century. These require animal models and proper methods to isolate bacterial strains and culture them<sup>6,7</sup>. However, animal models are not always available and many pathogens are human-specific and therefore cannot be transformed to animal. These requirements make the identification of infection agents very challenging. This is further emphasized by the discovery of polymicrobial diseases<sup>8</sup>, recent findings regarding the role of the microbiota in chronic diseases<sup>9</sup>, and the discovery of HGT responsible for the transport of genetic material between bacteria<sup>10</sup>.

Recent advances in next-generation sequencing (NGS) technology have made microbial sequencing highly accessible: bacterial sequences are now regularly collected during routine monitoring and in clinical studies. Many new NGS databases are forming, thus increasing the amount of available bacteria sequences<sup>11–14</sup>. To date, complete genome sequences of almost all major bacterial pathogens have been determined, providing significant insights into microbial pathogenesis and the mechanism of drug resistance. In addition, several repositories that collect virulence factors and annotate their structures, functions and mechanisms, are available<sup>15,16</sup>.

Furthermore, sequences of non-human pathogenic (NHP) bacteria, such as microbiome data, are also collected and deposited in sequence databases<sup>17–19</sup>.

Altogether, the available number of sequenced bacterial genomes is at the range of hundreds of thousands and is growing rapidly<sup>12,14</sup>.

In spite of the rising availability of sequences of bacterial genomes, until recently it was not possible to sequence bacterial samples in clinically relevant time, especially in developing countries. However, new technologies, such as the portable sequencer MinION<sup>20</sup>, allow quicker gene sequencing; a previous study sequenced and assembled an 860Mbp genome in two days using MinION<sup>21</sup>. In addition, MinION was recently used to retrieve reliable clinical information of a hospital *Salmonella* outbreak within half a day<sup>22</sup>. Such progress in sequencing technologies motivates a “Future Clinic” vision, in which patient diagnosis for infectious diseases will be based on quick real-time bacterial sequencing and bioinformatic analysis of inpatient samples. The bioinformatic analysis should provide a complete assessment of known and potential pathogenic strains.

To utilize the vast sequencing information obtained by whole genome sequencing, bioinformatic tools should be put into place to allow effective determination of pathogenic strains. Such tools have been developed for this purpose in the last decade<sup>23–27</sup> and are discussed in Section 2. The vast diversity of bacteria as well as the quick bacterial mutation rate, suggest that novel bacterial HP will continuously evolve, sometimes having genomes closely related to their commensal counterparts. Thus, to correctly identify a bacterial HP, a bioinformatic approach which can detect relatively small changes in bacterial genomes is required. Here we introduce a Machine-Learning based approach that overcomes genetic divergence in predicting bacterial pathogenicity, by training on a wide range of species with known human-pathogenicity phenotype.

Recent microbiological studies for bacterial outbreaks addressed the transition of a microbial genome from “friend”(NHP) to “foe”(HP) as a process involving either the acquisition (mainly via HGT), or the mutation of a small set of genes that are known to be involved in pathogenicity and antimicrobial resistance pathways<sup>28</sup>. Therefore, tools designed for HP/NHP classification should not be limited to the narrow set of genes that are currently known to be related to virulence. To this end, we develop a classifier to differentiate potentially HP and NHP based on complete microbial genomes.

Our proposed classifier is genome-based, thus all genes encoded by microbial genomes can potentially be included in the classification model. Therefore, the number of potential features (six million genes), greatly exceeds our available training set size (tens of thousands of genomes). In such circumstances, there is a risk of overfitting the model to the training data, which would cause the model to perform poorly on other, yet unseen, genomes. To tackle this challenge, we employ a “Sparse Support Vector Machine (SVM)” learning method, which generates a genomic model of pathogenicity that uses a relatively small set of genes. This method exploits the fact that a linear SVM with L1-norm regularization inherently performs feature selection by assigning weights equal to zero to all but a small set of features<sup>29</sup>. This property both guards against overfitting, thus improving classification accuracy on unseen data. In addition, it allows better human understanding of the model’s key features (i.e. the genes whose acquisition turns an NHP genome to HP). This will likely reveal novel genes that are linked to bacterial virulence. We show some potential examples of this in our discussion of the results in section 6.2. Another benefit of this approach is that it does not require manual selection of meaningful features, as done in some of the previously proposed pathogenicity classifiers<sup>24,26</sup>, making our method fully automatic and reproducible. Another computational challenge that we face involves scaling up of the training process to efficiently handle the number of available genomes, in our training data, which is larger than the training data used in previous works, as we specify below. The classification process requires each organism to be represented by a set of features which is comparable to those of the other organisms. Since each organism has its unique set of genes, using the genes directly as features will generate a representation, in which organisms rarely share features, and thus cannot be compared. To solve this problem, previous studies<sup>24,26</sup> clustered similar genes into protein families (PFs). However, these studies (reviewed in the section 2) cannot practically scale up, in terms of CPU time required for model training, to the fast-growing amount of training data that is becoming available. For example, in a previous study it took four weeks to cluster genes from 885 organisms<sup>26</sup>. This subject is further discussed in Section 2. In this work, we use a larger data set, of 21,155 organisms, extracted from the PATRIC dataset<sup>30</sup>. Assuming a linear time dependency, using the same approach and the same type of computers, our dataset would require 22 months to cluster. Thus, clustering time is a bottleneck to the efficiency of training, and render the scaling-up of previous approaches to this problem impractical. To solve this problem, we propose a scalable

model, which uses data more effectively, thus reducing clustering time substantially. The method we suggest here allows us to use faster clustering approaches, without compromising the accuracy of the model's predictions. This is explained thoroughly in Section 4.

We review previous related work in section 2. In section 3, we describe our novel contribution, which includes the method, the tool and the dataset. This is followed by a detailed description of our methods, in section 4, and a full explanation of how our dataset was created, in section 5. Our final results are discussed in section 6, which includes quantitative as well as biological results. We conclude in section 7, presenting possible future directions.

## 2 Related work

A number of sequence-based methods have been developed for microbial pathogenicity prediction. One such method, proposed by Garg et al.<sup>23</sup>, uses a cascade SVM classifier<sup>31</sup> to predict whether a given bacterial protein is related to bacterial virulence or not. To train the cascade SVM classifier, two sets of features are used; the first set measures levels of amino acid compositions; the second set measures sequence similarity to proteins found in the database. The similarity is measured using PSI-blast<sup>32</sup>, a method for comparing distant protein sequences. Our approach uses a broader view, and analyzes whole bacterial genomes as potentially pathogenic or not, rather than specific proteins. This provides a wider genome context consideration: a specific gene could contribute to bacterial pathogenicity in the context of one genome, while playing an NHP role in the context of another genome. With a clinical perspective in mind, we claim that all genes should be taken into consideration as well as their genetic context. This will allow better prediction of whether a given bacteria is HP or not.

Another method, developed by Iraola et al.<sup>25</sup>, proposed an SVM model to predict bacterial virulence, using known orthologous gene groups. The researchers created a presence/absence table, showing which orthologous groups (genes) were present or absent in the tested organisms. Based on this table, they performed feature selection using an iterative algorithm. Eventually, 120 out of the 814 original orthologous groups were used as the model's features. Both Garg et al. and Iraola et al.'s methods rely on pre-established virulence databases, which annotate virulence at the gene level<sup>23,25</sup>. One downside of this approach, is that many unannotated genes, whose sequences



are available, possibly possess virulence (or anti-virulence) roles, and this information is lost when only characterized virulence genes are being considered. Our approach utilizes the complete genetic information, thus allowing novel unannotated proteins that contribute to bacterial virulence to be discovered, and to be used for accuracy.

Other research groups attempted to develop pathogen prediction tools without using pre-established protein families, but rather by creating protein families and annotating them, based on their appearance frequency in pathogenic or non-pathogenic organisms<sup>24,26</sup>. Both studies required clustering the proteins of all organisms in the training set as an initial step for creating protein families (PFs). Then, PFs significantly enriched in either HP or NHP were assigned a weight value, depending on the degree of the enrichment. Families that were considered not significantly enriched were discarded. Then, given a new example (bacterial genome), its protein sequences were aligned against the PFs, and a score was computed. According to the score, the strain's pathogenicity was determined. The first method, by Andreatta *et al.*<sup>24</sup>, used a small set of 24  $\gamma$ -Proteobacteria genomes. Since their dataset was very small, they used all vs. all BLAST to cluster the genes to PFs. Pathogenfinder, by Cosentino *et al.*<sup>26</sup>, extended the former approach to a variety of bacterial species. They used CD-HIT<sup>33</sup> (reviewed in Section 4.1) to cluster the genes, since an all-vs.-all BLAST would have been too demanding in terms of CPU time. In addition, they identified genes that were predicted to be the most significantly associated with (or important for) pathogenicity or non-pathogenicity. While these methods were novel in not relying on previous PF, they had a few limitations: the thresholds to determine whether a PF is significantly enriched were manually selected, therefore they cannot be used on a different dataset. In addition, both methods were trained on datasets that included bacterial species that were never detected in a human host. This is likely to weaken the relevance of the method and its accuracy estimates to clinical samples. Finally, the methods were designed based on the genomic data available at the time, but unfortunately cannot be scaled up to the increasing volume of genomic data that are quickly becoming available. This is because protein clustering imposes a major computational bottleneck. We designed our approach with these issues in mind, and therefore created a fully automated method, trained it only on clinically relevant data (commensal microbiota and human pathogens), and designed a clustering approach that is more effective, and reduces computational time to a feasible value. This is further discussed in section 3.

Another interesting study, done by Barbosa *et al.*<sup>34</sup> investigated whether organisms can be separated according to the range of hosts they are pathogenic to; these included human-pathogens, broad-spectrum pathogens, opportunistic pathogens, and non-pathogens. The study included 240 organisms as a training set, limited to the actinobacteria family. All bacterial genes were aligned against each other using BLAST, creating a similarity matrix, which was used to cluster the genes to similarity groups. Influential groups were selected by applying a random forest technique to the entire training set. Then the data was split to training and validation sets. The selected features were used to train decision trees on the training set to identify the different pathogenicity groups. Their results show that broad-spectrum pathogens and human pathogens can be separated from non-pathogens. However, since the initial bacterial dataset was very small and the features were selected based on the entire dataset (without a proper distinction between training and validation) this model cannot be applied to broad bacterial genomes. In contrast, our classification model was trained on a large dataset, including bacterial genomes from diverse bacterial phyla. This ensures the generality of our proposed approach, and its applicability for broad clinical samples. Moreover, we validated our method on data that was excluded from the training process entirely; therefore, our validation results better represent the model's accuracy on future unseen data.

A more recent tool, PaPrBaG<sup>27</sup> used short genomic reads as raw input. The researchers in this study noted that the vast majority of available microbial genomes belong to pathogens and therefore, to compensate for the data imbalance, a single, randomly selected, genome per species was considered. They used two types of features: amino acid features and DNA features. To extract protein features, a reading frame was heuristically selected for each read, based on minimizing the number of stop codons, and amino acid features (AA frequency, AAindex<sup>35</sup>) were extracted. For the DNA features, nucleic acid k-mer compositions were used. To reduce the number of features, important features were then selected using permutation tests and Gini index values (a coefficient closely related to the area under the ROC curve<sup>36</sup>), on the entire dataset. Using these selected features, the model was evaluated by training a random forest classifier with a 5-fold cross validation process. As PaPrBaG method uses raw reads, its contribution to modeling infectious mechanisms is limited. In addition, this method, similarly to the one by Barbosa *et al.*<sup>34</sup>, selects features using the entire dataset, thus the cross-validation results may not truly represent the model's success

on an independent unseen dataset. As mentioned above, our approach predicts pathogenicity for unseen bacterial genomes, and thus it does not use any details from the validation set for training the model or for selecting its features.

All the above studies dealt with a significant data imbalance. Bacterial samples are often collected and sequenced, in clinics, from diseased patients. Thus, most sequenced bacteria in most databases, including ours, are of HP bacteria. In our data we have a ratio of 1:5.45 of NHP:HP bacteria. Our training method and analysis take this imbalance into account, as we further discuss in subsequent sections.

### 3 Our Contribution

We propose a new methodology for automatically building a model for pathogenicity identification, based on a training set of organisms, which includes, for each organism, its genome and its PATRIC annotations. The model can then process the proteome of an unseen organism (not included in the training set), and predict whether this organism is HP or NHP. Our method has several advantages over previous available methods:

1. **Dataset size:** We use a much larger dataset than previously used, consisting of 21,155 bacterial genomes. We extract this data from one of the main publicly available databases for microbial genomes, the Pathosystems Resource Integration Center (PATRIC) [<http://www.patricbrc.org>]. This database provides researchers with an online resource that stores and integrates a variety of data types (e.g. genomics, transcriptomics, protein–protein interactions (PPIs), three-dimensional protein structures and sequence typing data) and their associated metadata. As of July 27<sup>th</sup> 2017, PATRIC, contained 106,260 sequenced bacterial genomes<sup>30</sup>. We use genomes that are marked as WGS.
2. **Dataset relevance:** We focus exclusively on bacteria found in human samples (pathogenic or commensal), which are more relevant to clinical settings. A sample taken from a patient will naturally only contain genomes of human-hosted bacteria. Therefore, a human-pathogenicity classifier should be trained on human-hosted bacteria, so that it can obtain a high accuracy in identifying pathogenicity among those bacteria. Our dataset therefore includes only human-hosted bacteria. From a statistical perspective, identifying which bacteria are pathogenic in a dataset that includes only human-hosted pathogens is a more difficult problem than the one addressed by previous pathogenicity

classifiers<sup>24–26</sup>, since there are more common genes within human-hosted bacteria than between human-hosted and, e.g., plant-hosted bacteria. A model trained on a dataset that includes plant-hosted bacteria can associate plant-specific genes with an NHP classification, since plant-hosted bacteria are usually not human-pathogenic. Such a bias improves the model’s accuracy when measured on a population of bacteria that includes plant-hosted ones. However, this high accuracy does not represent the model’s success on the more relevant human-hosted bacteria. To conclude, training and testing the model exclusively on genomes from human-hosted bacteria results in a more relevant classification problem than the ones previously addressed, yet this problem is also statistically more challenging.

3. **Annotated data.** Annotations that clearly state HP or NHP are needed for this study and for future studies focusing on bacterial pathogenicity. We developed a method for automatic annotation of organisms as NP/NHP based on their PATRIC annotations. We include the resulting HP/NHP annotations for 21,155 bacterial genomes in the supplementary materials section. The method that we used to annotate the genomes is detailed in section 5.
4. **Automatic construction of the sparse model.** We use a sparse SVM approach, which has an inherent preference to models that use fewer features (in our case, genes). Using a small number of meaningful features improves the model in two aspects: first, it yields a less complex, more human-interpretable biological infection model, which can then be further studied. Second, by reducing the dimensionality of the model, the model becomes less dataset-specific, which improves its expected prediction accuracy on unseen bacterial genome samples. Importantly, unlike the previous methods discussed above, the sparse SVM approach does not select each feature individually. Instead, it optimizes for the best set of features that altogether achieve a high prediction accuracy. This takes into consideration that some features are more predictive in the presence of additional features, while some features are redundant when some other features exist in the model. These issues are not addressed when features are individually selected, as done in the methods of Iraola *et al.* and Cosentino *et al.* The automatic feature selection we present, does not require manual intervention, and can be applied out-of-the-box on other datasets.

5. **Scalability.** Clustering the proteins into PFs is the most computationally intensive procedure during training, making it the computational bottleneck of the training stage. Since our data set is much larger than previous data sets, it is crucial to speed up this process in order to make the training stage feasible. We achieve a speed up in the clustering process using two methods. First, following the approach of Weizhong *et al.*<sup>33</sup>, we hypothesize that the longer protein sequences contain more sequential features (i.e. domains) than shorter protein sequences. Also, we observe that during the clustering computation, CD-HIT processes input protein sequences sorted by length (longest to shortest). Therefore, it was natural to select a subset of clusters generated from the longest input sequences, and to stop the CD-HIT process once these sequences are clustered. Therefore, we only use the longest proteins in the training set (top 10% of the total number of proteins) to generate PFs. This results in a significant reduction in the time needed for model training. We demonstrate in section 6.1 that, when employing our classification approach, this does not have a significant effect on the resulting model's accuracy. The model's high accuracy (described in section 6.1) supports the validity of our hypothesis regarding the importance of longer protein sequences. We discuss this hypothesis further in section 6.1. The second method we apply for speeding up the clustering process is running the CD-HIT clustering package using its "fast mode"<sup>33</sup>. We explain in section 4.1 why in our training approach, the use of "fast mode" does not affect the final result.

Our method is fully automated, and does not require any manual hand-tuning of parameters. It can thus be easily used to train a pathogenicity prediction model using an updated dataset or a completely different one: this is likely to be useful in the future, since the current dataset of sequenced bacterial strains is rapidly growing and more data is becoming available. We implemented our approach in a pipeline which we term "BACPACSS" (Bacterial Pathogenicity Classification via Sparse SVM). The full code for BACPACSS is included in the supplementary material, in addition to the annotated data set and the trained model.

## 4 Methods

The workflow of our classification approach is illustrated in Figure 1.

### 4.1 Extracting features

Like Cosentino *et al.*<sup>26</sup>, our method uses CD-HIT to extract PFs, which are then used as features from the protein sequences. CD-HIT is a greedy incremental clustering algorithm. The basic CD-HIT algorithm sorts the input sequences in order of decreasing length, and processes them sequentially in this order. Each protein is classified by CD-HIT either as redundant (similar to an existing representative) or as a new representative, defining a new cluster. The first sequence is automatically classified as the first cluster representative sequence. Then each of the remaining sequences is compared to the representative sequences found before it, and is classified as redundant or representative, based on whether it is similar to one of the existing representative sequences or not.

In our approach, as in Pathogenfinder<sup>26</sup>, the PF representatives which are generated using CD-HIT are used as possible features for the classifier, and the sequence-similarity threshold is set to 40% (similarly to the parameter-setting used by Cosentino *et al.* in Pathogenfinder). However, in our implementation we introduce a key difference: we use only the longest gene sequences in the training set as input for CD-HIT. We set the threshold at selecting 10% of the total number of proteins, since this threshold yields practical clustering time and proved robust on a smaller dataset used to test this approach (this is discussed in section 6.1). This reduces the clustering time by a factor of 20, as shown in detail in section 6.1.

To extract an individual feature vector for each organism, we use CD-HIT-2D<sup>33</sup>, which is a variant of CD-HIT that compares query sequences to database sequences. Here, the queries are each organism’s protein sequences, and the database is the list of PF representatives generated earlier by CD-HIT. The output indicates which PF representatives have matches in each organism. Again, we set the threshold to 40% sequence similarity. Here we use CD-HIT’s “accurate mode”, which selects the most similar representative for each of the organism’s proteins. This takes longer, but this stage can easily be distributed between different computers, since running organisms against the PF representatives can be done independently. For each organism, a binary feature vector is created, based on the identification of matched PFs, as follows: The vector has a coordinate for every PF, and the *i*’th coordinate is 1 if the organism

has a protein matching the PF representative indexed  $i$ , and 0 if the organism has no match for this PF.

We make another change in our use of CD-HIT to improve the running time of our clustering step, and use its “fast mode”. In CD-HIT’s “fast mode”, a query is attached to the first representative that it is similar to, without comparing it to other representatives. In “accurate mode”, a query is compared to all representatives, and attached to the most similar one. An important observation is that choosing “fast mode” over “accurate mode” only affects the redundant proteins (i.e. a protein that is not a representative, but rather a member of an existing cluster), by determining to which PF they belong. In contrast, it does not affect the resulting representatives at all. Therefore, since our method only uses PF representatives (unlike Cosentino *et al.*, who used the PF members as well), we apply CD-HIT using the “fast mode” setting. This saves many comparisons, without affecting the resulting features, which are the PF representatives, yielding an additional 20 percent improvement in running time. The running time measurements are detailed in Section 6.1.

## 4.2 Training the Classification Model

We train an SVM classifier using the vectors representing the organisms in the training set as input. Each such training vector is provided to the training procedure with a binary label, which identifies its known pathogenicity status based on our methodology which extracts this label from the PATRC annotations, detailed in section 5. We used SVM with a linear kernel, so that the output model is represented by a vector  $\vec{w}$ . This vector  $\vec{w}$  can then be used for prediction. To predict whether an organism with a feature vector  $\vec{x}_i$  is pathogenic, the following formula is used:

$$predict(x_i) = \begin{cases} HP, & sign(\vec{w} \cdot \vec{x}_i) > 0 \\ NHP, & otherwise \end{cases}$$

We use a version of SVM training which minimizes the following objective function:

$$\frac{C}{n} \sum_{i=1}^n max[0, 1 - y_i(\vec{w} \cdot \vec{x}_i - b)] + \|\vec{w}\|_1$$

Here,  $n$  is the number of training samples,  $x_i$  is the  $i$ ’th sample feature vector,  $y_i$  is the  $i$ ’th sample’s label (which can be 1 or -1),  $b$  is a bias term, and  $C$  is a regularization parameter, which we discuss below.  $\|\vec{w}\|_1 = \sum_{i=1}^n |w_i|$  is the L1 norm of vector  $\vec{w}$ .

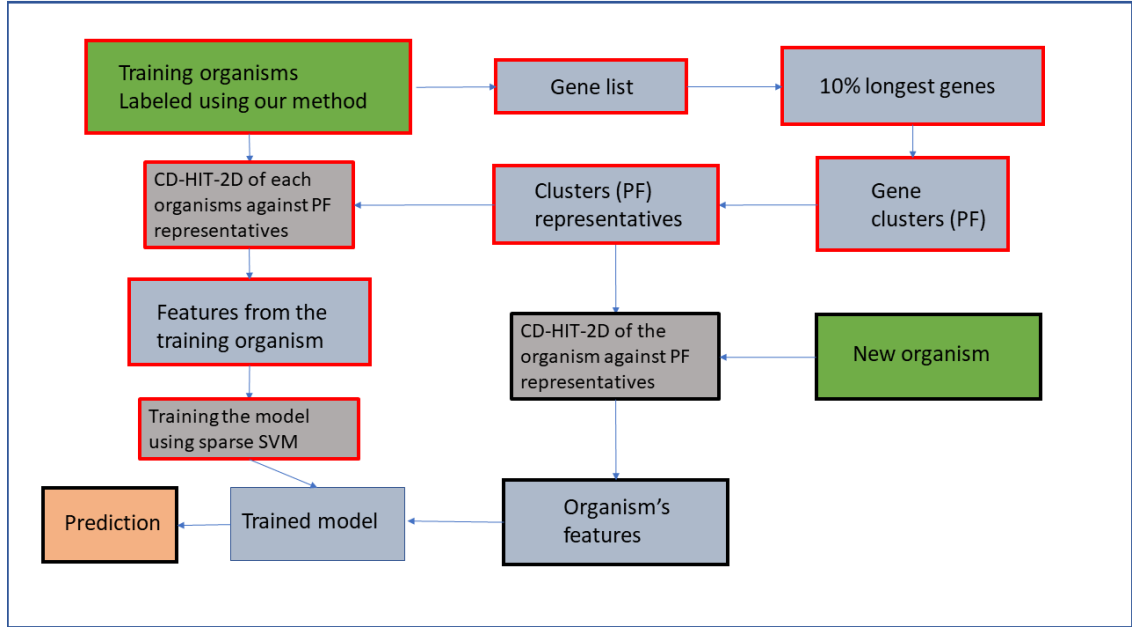
The formula above is different from the more classical SVM training objective. In the classical SVM training objective, the last term is  $\|\vec{w}\|_2^2 = \sum_{i=1}^n w_i^2$ . However, this classical formulation does not prefer sparse solutions, and so might cause the model to



use too many features, leading to less robust and less interpretable results. Our choice to use the L1 penalty is motivated by our goal of obtaining a sparse model, which uses less features. It has been shown that the L1 penalty is a good surrogate for directly minimizing the number of features in the model, a task which is computationally infeasible<sup>37</sup>. We implemented the training procedure using the python package Scikit-learn<sup>38</sup>, which provides an L1-SVM training module.

The linear SVM objective has a parameter 'C', which is a positive number that needs to be tuned for the classifier to produce optimal results. This parameter indirectly controls the number of features that the final model will have. A larger number of features will lead to a better accuracy on the training set, at the expense of a less sparse model, which could result in worse accuracy on unseen organisms. A smaller number of features could lead to a poor accuracy on the training set, which can also result in a poor accuracy on unseen organisms. Thus, the value of 'C' needs to be tuned to obtain the best expected accuracy on unseen organisms. To tune the 'C' parameter, our goal is to obtain a relatively balanced accuracy on both positive and negative examples. This is done via cross-validation on the training set alone. The 'C' value that produces the highest average F1-macro score (defined in Section 4.3) is selected. Due to the skewness of the data, using the training set as-is for training would result in a higher accuracy on pathogenic examples and a low accuracy on non-pathogenic ones. To overcome this issue, we oversample the non-human pathogen (NHP) examples. This is equivalent to counting each NHP example as if it appeared N times, where the ratio of HP to NHP in our dataset is N:1. Thus, the resulting input to the SVM training procedure is balanced in terms of HP vs. NHP frequencies.





**Figure 1 - Classification workflow.** Training steps are outlined in red, and prediction steps are outlined in black. Input cells are green and output cell is orange.

### 4.3 Scoring

We tune the value of the ‘C’ parameter using standard 10-fold cross-validation<sup>39</sup>. In order to evaluate the resulting model for various values of ‘C’ during the cross-validation process, we need a scoring procedure that assigns a value to each such model, representing its accuracy on the validation set. Due to the skewness of the data (HP-annotated genomes appearing approximately five times more than NHP genomes in the data set), regular accuracy (the proportion of correct predictions out of the validation set) would result in misleading scores. For example, assigning every object to the larger set (i.e. HP bacteria) achieves a high proportion of correct predictions, even though in this case every object in the smaller set (i.e. NHP bacteria) is labeled incorrectly. Therefore, we use instead an averaged F1-score measure. F1-score is the harmonic mean of precision ( $\frac{TP}{TP+FP}$ ) and sensitivity ( $\frac{TP}{TP+FN}$ ). By definition, both these scores relate to the positive label of the data set; in this case, HP. We use F1-macro, an unweighted average of F1-positive (the F1-score defined above) and F1-negative, for which precision and sensitivity values are computed using the negative label. The “negative precision” ( $\frac{TN}{TN+FN}$ ) is termed Negative Predictive Value (NPV) and “negative sensitivity” ( $\frac{TN}{TN+FP}$ ) is termed specificity, or True Negative Rate (TNR). Using an

unweighted average balances the significance of both labels (HP and NHP) when evaluating the classifier, since in such case both F1-positive and F1-negative have equal weights. Thus:

$$F1\text{-positive} = \frac{2 * precision * sensitivity}{precision + sensitivity}$$

$$F1\text{-negative} = \frac{2 * NPV * TNR}{NPV + TNR}$$

$$F1\text{-macro} = \frac{F1\text{-positive} + F1\text{-negative}}{2}$$

## 5 Dataset

Due to the considerations described in section 3, we focused only on human-hosted bacterial genomes. Out of the 106,260 bacterial genomes in the PATRIC database<sup>30</sup>, we used a subset of 40,297 human-hosted bacteria, identified by having host name ‘Homo sapiens’ or ‘Humans, Homo sapiens’ in PATRIC. Unfortunately, the PATRIC database does not include pathogenicity annotations (nor does any other database we are familiar with). Thus, we created an annotation-based pathogenicity classification method. This method was used to associate a pathogenicity label with each organism in our dataset. We labeled 17,881 organisms as human pathogens (HP), 3,274 as non-human pathogens (NHP), and 19,412 as inconclusive, using the process described below.

We downloaded (on March 15<sup>th</sup>, 2017), all ‘WGS’ and ‘Complete’ genomes available in PATRIC. The ‘Complete’ genomes list was downloaded in order to validate our annotation scheme using Pathogenfinder’s<sup>26</sup> self-annotated list of NHP/HP bacterial genomes. We used Pathogenfinder’s annotation of the ‘Complete’ genomes as a reference to control the annotation scheme we developed for the ‘WGS’ genomes. Their data was annotated mostly manually, using references from the literature, making it highly reliable.

Based on PATRIC’s annotations, we labeled genomes as HP if they satisfied at least one of the following criteria:

- The ‘Disease’ field is not empty, and does not contain a commensal term, as defined below.

- One of the fields 'Isolation Source', 'Host Health', or 'Comments' includes an HP term. In addition, the same fields cannot include any of the NHP terms (the terms used for HP and NHP are detailed below).
- A genome was manually verified as HP, by reviewing it in the literature.

Excluding the generated HP list, we labeled genomes as NHP if they satisfied at least one of the following criteria:

- One of the fields 'Isolation Source', 'Host Health', or 'Comments' includes an NHP term.
- One of the fields 'Isolation Source', 'Host Health', or 'Comments' includes a weaker NHP term.

This resulted in 17,881 HPs and 3,274 NHPs, making a ratio of 5.45:1 HP: NHP in the dataset. The criteria above were selected iteratively, using as reference the shorter list of the 'Complete' genomes. This list includes Pathogenfinder's genomes, and thus, for each iteration, the HP/NHP annotations were compared to theirs, and the criteria were optimized to minimize discrepancies.

The following term lists were used for the criteria above:

- HP terms: virulence, disease, superbug, patient, diarrhea, waterborne, foodborne, toxin, clinical, intensive, outbreak, infection, pathogen, water borne, food borne.
- NHP terms: healthy, probiotic, commensal.
- Weaker NHP terms: 'comparative', 'reference'.
- Commensal terms: 'healthy', 'Healthy', 'Commensal', 'Commensal (plant)', 'Periodontally healthy'.

To clarify, we term the weaker NHP terms as weaker, since they are not sufficient to reject HP, but when no HP indications are available, an annotation containing these terms is considered NHP.

## 6 Results and Discussion

The data was split to 10 stratified folds (each fold with the same HP/NHP ratio). For each fold, the proteins of the training set were clustered to create PFs as described in Section 4.1. Using the PF features, a stratified 10-fold cross validation was performed within the training set, to optimize the classifier's 'C' parameter (see section 4.2 above). The classifier was then trained on the training set, using the optimized parameter from

the cross-validation process. Lastly, the classifier was evaluated on the unseen test set (using the PF features, which were created solely on the training set). Note that in this process we made sure that the test set is not used in any way during the feature construction or training process, so that accuracy measures on the test set are representative of new genomes never seen during the training procedure.

## 6.1 Quantitative results

The accuracy results for each split into training set and test set are given in Table 1.

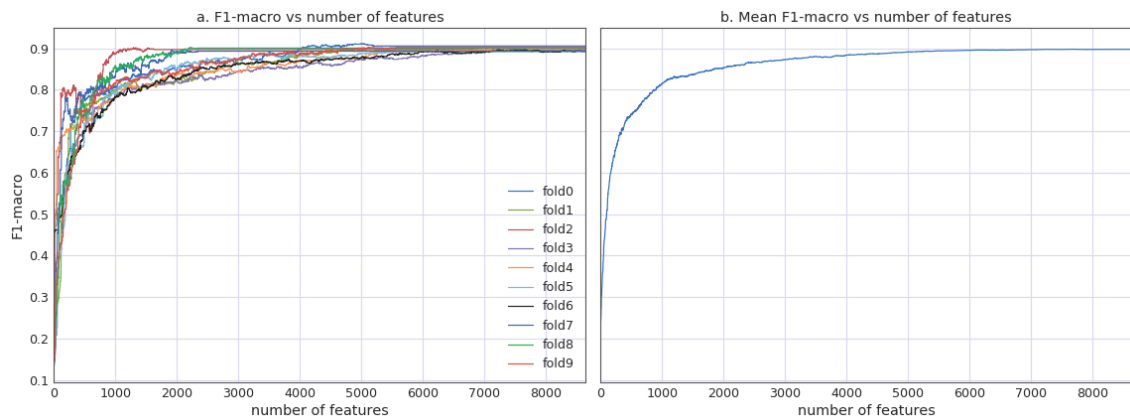
Split	F1-macro	PR-AUC	ROC-AUC	sensitivity	specificity	MCC
0	0.905	0.994	0.972	0.978	0.811	0.811
1	0.896	0.993	0.970	0.965	0.835	0.793
2	0.897	0.994	0.973	0.950	0.899	0.798
3	0.896	0.991	0.961	0.971	0.808	0.791
4	0.899	0.991	0.967	0.961	0.860	0.799
5	0.891	0.993	0.968	0.969	0.805	0.783
6	0.893	0.991	0.968	0.967	0.817	0.785
7	0.893	0.992	0.967	0.962	0.838	0.786
8	0.900	0.992	0.963	0.968	0.835	0.801
9	0.902	0.995	0.975	0.968	0.845	0.804
Avg	0.897	0.992	0.968	0.966	0.835	0.795
Std	0.0045	0.0014	0.0044	0.0072	0.0287	0.0091

**Table 1 - Summary of ML Accuracy results with different measures.**

The columns in the table are as follows: F1-macro, sensitivity and specificity are defined in section 4.3. PR-AUC is the unweighted averaged area under the precision-recall curve, for both labels. The negative label's (NHP), PR-curve is actually the NPV vs specificity (see section 4.3) curve. ROC-AUC is the area under the ROC curve. Matthews Correlation Coefficient<sup>40</sup> (MCC) is a measure of the quality of a binary confusion matrix, ranging from -1 (complete disagreement between predictions and observations) and 1 (perfect prediction). For reference, Pathogenfinder's<sup>26</sup> classifier scored an MCC of 0.758 on a different dataset, which also contained HP and NHP bacteria. However, their database (both training and testing data) included bacteria hosted by animals and plants as well, while ours included only human-hosted bacteria. Genomes of animals and plants are likely easier to identify as NHP, since the classifier can use animal-specific or plant-specific genes as an NHP bias.

To examine how many features are truly needed by our models for accurate prediction, Figure 2.a plots F1-macro scores against the number of features that are used by the model. To generate this plot, for each fold and each value of  $n$  on the  $X$  axis, we

generated a model with  $n$  features, by taking the model that the classifier learned on this fold, and setting to zero the weights of all the features in this model, except for the  $n$  features with the highest absolute weight in the classifier vector  $\vec{w}$ . The plot shows the F1-macro score of each model. Figure 2.b shows the average F1-macro scores of the models generated for each of the folds. It can be seen that there are differences between the maximal numbers of features used by the trained model in each fold, from as low as 1724 for fold 2 to as high as 8649 for fold 3. Indeed, the SVM 'C' parameter selected by the cross-validation process was 0.1 for fold 2 and 100 for fold 3. These differences likely result from differences in the content of the training set in each fold. However, the overall test accuracy of each of the models on their respective test set is similar, as the standard deviation scores in Table 1 show. It is likely that features that are present only in a small number of models out of the 10 models generated for the 10 different folds are not statistically significant for pathogenicity prediction, for the type of organisms in this dataset. The model trained on the entire data set (provided in our Github directory) uses 9,469 features. Its 'C' parameter was set by the cross-validation procedure to 300. This larger number of features (corresponding to this high 'C' value) might be due to the larger training set, which reduces the risk for overfitting, thus allowing the model to use more features. Alternatively, it could be a result of statistical variation.



**Figure 2 - F1-macro score vs number of trained features. a. Each fold's F1-macro score. b. Overall mean of all 10 folds.**

Comparing our results to existing studies turned out to be difficult: PaPrBaG<sup>27</sup> used raw reads, from a different set of organisms, and thus, it is impossible to compare to.

Pathogenfinder<sup>26</sup> used data in a similar format to ours, but due to issues in the Pathogenfinder interface, we found it technically impossible to test its classification accuracy on more than a handful of organisms. However, our average MCC score of

0.795 is higher than Pathogenfinder's MCC score of 0.758. Next, we validate our hypothesis that clustering only the 10% longest genes speeds up model training substantially and does not have a significant effect on the model's accuracy. Since our training set is large, and clustering it fully would be highly computationally demanding – rendering it impractical, we tested this hypothesis on a smaller data set. For this purpose, the data set described by Cosentino et al.<sup>26</sup> was used, which includes 885 training organisms and 449 validation organisms. As done with the dataset that we used for generating our model, a validation set was set apart for the entire training process, and was only used to evaluate the accuracy of the generated models. In Table 2, we show the results of this comparative analysis. We compare both the model's training time and its accuracy as measured by the F1 score and by the prediction accuracy on the validation set. Here we used regular accuracy and F1-positive measures since the dataset is relatively balanced. Our measurements show that the training time is more than 20 times faster when clustering only 10% of the longest proteins, and that this speedup does not come at the expense of accuracy and F1 score on this dataset. We observe a decline in performance only when the threshold is set to select only 2.5% of the longest genes for clustering.

Moreover, we observe that the accuracy measures for different thresholds between 5% and 100% is not monotonic in the threshold, thus we believe that the differences result from statistical variation and not from any inherent deficiency of using a threshold in the range 5%-100%.

We selected a threshold of 10% for clustering our dataset since this was computationally feasible, and according to the table below, it is also sufficient.

Assuming that on our dataset, clustering 100% of the proteins would also have taken almost 20 times more CPU time than clustering the 10% longest proteins. For us, using a computer with 32 CPUs, clustering 10% of the longest proteins of the training set (90% of the whole set) took 12 days. Thus, clustering the entire data could take

$$12days \times \frac{100}{9} = 133days, \text{ for each of our ten folds.}$$

% genes used	CPU time (days)	Accuracy	F1 score
100	395	87.44	0.83
20	53	86.55	0.82
15	35	87.67	0.83
10	20	86.77	0.83
5	7	87.00	0.82
2.5	2	84.98	0.80

**Table 2 - Comparison of different sizes of gene lists used to train our model.**

## 6.2 Biological results

We study PF representatives that received the highest positive weights in our classification model, based on the final model (available in our supplementary material) which was trained on the entire dataset. In general, such features are expected to have a strong positive correlation between their presence in the genome of a bacterial strain and the involvement of this specific strain in human diseases. We found that many of these features have broad functional activities (Table 3). Since we made no *a priori* assumptions regarding the genes that are expected to receive high positive weight, unlike previous studies<sup>23,25</sup>, we found genes that were never reported to be related to bacterial virulence alongside genes that are known virulence factors. 4,885 PF representatives received positive weights in our classification model, linking them to an HP lifestyle, and 4,584 representatives received negative weights, indicating that their presence in the bacterial genomes is not correlated with human diseases.

Table 3 summarizes the characteristics of 25 PFs, which received the highest positive weights. The number of HP bacteria (HP) and NHP bacteria (NHP) that have the PF in their genomes and the normalized HP/NHP ratios (taking into account the data's imbalance - 1:5.45 NHP to HP bacteria) are presented. The bacterial spectrum is also presented in the form of number of genera that contained the PF. This information indicates if the PF feature is spread among the bacterial population or mostly limited to a specific genus. (Additional information, such as the details regarding the specific genera contributing to the spectrum per bacteria, are included in the supplementary material).

Among the 25 top-scoring virulence-related genes we found genes that encode antitoxin proteins (genes 1, 9-10, 14 in the table), phage tail fiber proteins (genes 2 and 3), mobile elements (genes 8 and 20), secretion/transporter systems (genes 4 and 18), and biofilm associated proteins (genes 13 and 25). Since many virulence factors are suspected to spread by HGT (acquisition of free DNA, phage transduction or by bacterial conjugation), our finding that genes encoding mobile elements and phage proteins have high positive weights is not surprising. In addition, many secretion systems were previously reported to be involved in antibiotic resistance, immune system modulation and virulence mechanisms that allow pathogenic bacteria to survive within the host environment and fight host-immune mechanisms. Biofilm production allows pathogens to protect the bacterial community by forming multi-cellular structure.

Antitoxin production is related to the ability of a bacterial strain to produce a potent toxin against the host cell or the commensal microflora without affecting itself.

We found several metabolic genes that received high scores (genes 6, 7, 16, 19, 21). These were involved in diverse metabolic pathways such as amino acid, nitrate and iron metabolism. While iron metabolism was previously suggested to be important for bacterial virulence, as it allows acquisition of the limited iron elements, the other metabolic pathways were not previously reported to be directly related to bacterial pathogenesis. This finding demonstrates the advantage of analyzing a combination of multiple genes that collectively can predict the nature of bacterial strains according to their genome.

Surprisingly, many of the top genes on the list are uncharacterized and their function is unknown (genes 5, 10-12, 15, 17, 22-24). We believe that studying these genes might reveal novel virulence mechanisms and will be of great importance to the field of infectious diseases. To acquire initial information on the role of these uncharacterized genes, we examined whether any of the proteins assigned to the PF of the representative protein contain a known function. Unfortunately, all proteins within the PFs of genes 10, 11, 17, and 23 had hypothetical proteins with unknown functions. To examine whether they contain conserved domains of characterized proteins we analyzed them thorough NCBI Conserved Domain Search<sup>41</sup>. Gene 10 in Table 3 was found to have a conserved region, at amino acid positions 44-603, that appeared in many bacterial genes. This region was found to be 100% identical to a putative protein conjugal transfer protein, called Tral. This might indicate that gene 10 encodes a protein that can be translocated to other bacterial strains or is involved in the bacterial conjugation process, which allows bacteria to transfer DNA horizontally. These abilities are likely related to the pathogenicity of bacterial strain. Gene 23 in Table 3 was found to include a protein domain, termed LXG, at positions 432-570 that is found in a group of polymorphic bacterial toxins. Such toxins are predicted to use the Type VII secretion pathway to mediate their export. We found that Gene 11 in Table 3, contains cell surface protein domains, suggesting it localizes on the bacterial membrane, where many virulence factors are found. Gene 17 at the list, was found to have a Histidine kinase-like ATPase domain, which is not necessarily related to bacterial virulence.



#	protein id (PATRIC)	Function	Genus broadness	weight	HP	NHP	P - ratio
1	fig 57678.3.peg.3911	Antitoxin/ABC transporter/Methionyl-tRNA synthetase	13	1.121	165	11	2.75
2	fig 562.5116.peg.303	Phage tail fiber protein	9	1.041	9	1	1.65
3	fig 1035839.4.peg.497	Phage tail fiber	9	1.011	18	4	0.82
4	fig 104628.18.peg.787	ABC transport system/drug resistance efflux pump	24	0.954	64	7	1.67
5	fig 1638939.5.peg.6205	Secreted protein - Unknown function	7	0.885	11	2	1.01
6	fig 1896987.3.peg.84	Permease/amino acid metabolism	9	0.879	26	7	0.68
7	fig 546275.3.peg.221	Iron-metabolism	8	0.814	28	1	5.13
8	fig 908937.4.peg.2347	Mobile element	6	0.810	23	2	2.11
9	fig 518636.5.peg.5056	Antitoxin	9	0.808	7	7	0.18
10	fig 777.21.peg.2668	Unknown function	5	0.757	258	1	47.24
11	fig 873513.3.peg.40	Unknown function	3	0.749	6	3	0.37
12	fig 160453.3.peg.3556	–Putative lipoprotein – Unknown function	7	0.734	256	0	inf
13	fig 29385.56.peg.1991	Biofilm-associated protein	10	0.732	41	1	7.51
14	fig 1637974.4.peg.3466	Antitoxin	5	0.729	6	0	inf
15	fig 38289.22.peg.1916	Unknown function	5	0.726	53	0	inf
16	fig 48296.46.peg.4105	Nitrate metabolism	9	0.719	44	3	2.69
17	fig 1276.5.peg.1875	Unknown function	8	0.718	10	3	0.61
18	fig 53378.3.peg.5629	Type IV secretory pathway	8	0.713	13	3	0.79
19	fig 1008457.3.peg.262	Metabolism of amino groups	11	0.708	18	1	3.30
20	fig 108486.3.peg.1334	Mobile element	12	0.699	42	1	7.69
21	fig 866774.4.peg.1228	Aminopeptidase	7	0.693	12	2	1.10
22	fig 29303.4.peg.53	Unknown function	7	0.678	15	1	2.75
23	fig 1408286.3.peg.775	Unknown function	11	0.678	122	8	2.79
24	fig 742821.3.peg.2226	Unknown function	15	0.673	42	15	0.51
25	fig 525372.3.peg.2084	Fimbria adhesion/Two-component system sensor/Acetylornithine deacetylase	9	0.670	27	15	0.33

**Table 3 - Top 25 PFs and their functional categories, sorted in decreasing weight order. Hosts were determined using all members of each PF. Some PFs were species specific, and some broader. The weight column in the table denotes the weights assigned to the corresponding feature by the classifier; the higher the weight, the more meaningful the feature.**

An interesting example of our model's success is nicely demonstrated in the *Acinetobacter* genus. Our dataset included a balanced bacterial population of this genus, with 689 HP and 627 NHP organisms. Out of these bacterial strains, 92.3% of the HP genomes were correctly classified as HP (sensitivity), and 93.0% of the NHP genomes were correctly classified as NHP (specificity).

At this point, we could not fully comprehend the biological correlation between the functions of the PF that obtained the highest negative weights and the non-pathogenic bacterial life-style. It is possible that the genes required for NHP life-style are operating as multi-factorial components and therefore are harder to correlate with specific function/activity.

## **7 Conclusions and future work**

In this work, we investigated whether pathogenicity can be successfully predicted using a sparse set of informative PFs. To that end, we developed a machine learning tool, "BACPACSS". BACPACSS uses an L1-norm linear SVM classifier, which naturally performs feature selection during classification. For BACPACSS to scale well with large datasets, only 10% of the longest proteins in our dataset were clustered to create PFs. Considering the growing size of available bacterial genome sequences, scalability is of key importance. Strikingly, this resulted in a high accuracy prediction tool, which is completely automated, thus requires no manual configurations.

Pathogenicity classification training is highly dependent on available pathogenicity annotations. To that purpose, we created a protocol for pathogenicity annotation inference based on phenotypes which are readily available to download. This protocol can be further extended and tuned for a larger database in the future.

Aside from being a prediction tool, BACPACSS can be used to reveal previously unknown virulence genes, as was done in this paper. In the future, we are hoping to determine, based on our tool, a minimal set of genes, specific for a known pathogenic genus, that can differentiate quickly and effectively between pathogenic and non-pathogenic bacterial strains. This will assist medical researchers and future clinic practitioners to develop high quality kits based on these genes.

Our approach should not be limited to pathogenicity alone. It can be used as a general pipeline to predict different phenotypes (e.g. antibiotic resistance).

## 8 Supplementary material

The complete BACPACSS code, and the following files, can be found in

<https://www.cs.bgu.ac.il/barashe/>:

1. **genomes.xlsx** – the full list of the genomes we used, divided to HP, NHP and inconclusive.
2. **genus\_phylum.xlsx** – the full list of the different genera and phyla of the organisms we used in this work.

## 9 Bibliography

1. Prevention, C. for D. C. and. CDC Estimates of Foodborne Illness in the United States CDC 2011 Estimates. *Centers Dis. Control Prev.* **68**, 3–4 (2011).
2. Qin, J. *et al.* A human gut microbial gene catalogue established by metagenomic sequencing. *Nature* **464**, 59–65 (2010).
3. Hooper, L. V & Gordon, J. I. Commensal host-bacterial relationships in the gut. *Science (80-. )*. **292**, 1115–1118 (2001).
4. Soucy, S. M., Huang, J. & Gogarten, J. P. Horizontal gene transfer: building the web of life. *Nat. Rev. Genet.* **16**, 472–482 (2015).
5. Kelly, B. G., Vespermann, A. & Bolton, D. J. The role of horizontal gene transfer in the evolution of selected foodborne bacterial pathogens. *Food Chem. Toxicol.* **47**, 951–968 (2009).
6. Sassetti, C. M., Boyd, D. H. & Rubin, E. J. Genes required for mycobacterial growth defined by high density mutagenesis. *Mol. Microbiol.* **48**, 77–84 (2003).
7. Young, R. A. *et al.* Genes for the major protein antigens of the leprosy parasite *Mycobacterium leprae*. *Nature* **316**, 450–452 (1984).
8. Brogden, K. A., Guthmiller, J. M. & Taylor, C. E. Human polymicrobial infections. *Lancet* **365**, 253–255 (2005).
9. DuPont, A. W. & DuPont, H. L. The intestinal microbiota and chronic disorders of the gut. *Nat. Rev. Gastroenterol. Hepatol.* **8**, 523–531 (2011).
10. Gogarten, J. P. & Townsend, J. P. Horizontal gene transfer, genome innovation and evolution. *Nat. Rev. Microbiol.* **3**, 679–687 (2005).
11. Benson, D. A. *et al.* GenBank. *Nucleic Acids Res.* **43**, D30-5 (2015).
12. O’Leary, N. A. *et al.* Reference sequence (RefSeq) database at NCBI: Current status, taxonomic expansion, and functional annotation. *Nucleic Acids Res.* **44**, D733–D745 (2016).
13. Kulikova, T. *et al.* EMBL Nucleotide Sequence Database in 2006. *Nucleic Acids Res.* **35**, (2007).
14. Mashima, J. *et al.* DNA data bank of Japan (DDBJ) progress report. *Nucleic Acids Res.* **44**, D51–D57 (2016).
15. Chen, L. *et al.* VFDB: A reference database for bacterial virulence factors. *Nucleic Acids Res.* **33**, (2005).
16. Zhou, C. E. *et al.* MvirDB - A microbial database of protein toxins, virulence factors and antibiotic resistance genes for bio-defence applications. *Nucleic Acids Res.* **35**, (2007).
17. Gevers, D. *et al.* The Human Microbiome Project: A Community Resource for the Healthy Human Microbiome. *PLoS Biol.* **10**, (2012).

18. Chen, T. *et al.* The Human Oral Microbiome Database: a web accessible resource for investigating oral microbe taxonomic and genomic information. *Database (Oxford)*. **2010**, baq013 (2010).
19. Chen, I. M. A. *et al.* IMG/M: Integrated genome and metagenome comparative data analysis system. *Nucleic Acids Res.* **45**, D507–D516 (2017).
20. Gent, I. P., Jefferson, C. & Miguel, I. Minion: A Fast Scalable Constraint Solver. *17th Eur. Conf. Artif. Intell.* **141**, 98–102 (2006).
21. Jansen, H. J. *et al.* Rapid de novo assembly of the European eel genome from nanopore sequencing reads. *bioRxiv* 101907 (2017). doi:10.1101/101907
22. Quick, J. *et al.* Rapid draft sequencing and real-time nanopore sequencing in a hospital outbreak of Salmonella. *Genome Biol.* **16**, 114 (2015).
23. Garg, A. & Gupta, D. VirulentPred: a SVM based prediction method for virulent proteins in bacterial pathogens. *BMC Bioinformatics* **9**, 62 (2008).
24. Andreatta, M., Nielsen, M., Møller Aarestrup, F. & Lund, O. In silico prediction of human pathogenicity in the  $\gamma$ -proteobacteria. *PLoS One* **5**, e13680 (2010).
25. Iraola, G., Vazquez, G., Spangenberg, L. & Naya, H. Reduced set of virulence genes allows high accuracy prediction of bacterial pathogenicity in humans. *PLoS One* **7**, e42144 (2012).
26. Cosentino, S., Voldby Larsen, M., Møller Aarestrup, F. & Lund, O. PathogenFinder - Distinguishing Friend from Foe Using Bacterial Whole Genome Sequence Data. *PLoS One* **8**, e77302 (2013).
27. Deneke, C. *et al.* PaPrBaG: A machine learning approach for the detection of novel pathogens from NGS data. *Sci. Rep.* **7**, 39194 (2017).
28. Schmidt, H. & Hensel, M. Pathogenicity Islands in Bacterial Pathogenesis. *Society* **17**, 14–56 (2004).
29. Bi, J., Bennett, K., Embrechts, M., Breneman, C. & Song, M. Dimensionality reduction via sparse support vector machines. *J. Mach. Learn. Res.* **3**, 1229–1243 (2003).
30. Wattam, A. R. *et al.* Improvements to PATRIC, the all-bacterial Bioinformatics Database and Analysis Resource Center. *Nucleic Acids Res.* **45**, D535–D542 (2017).
31. Graf, H. P., Cosatto, E., Bottou, L., Dourdanovic, I. & Vapnik, V. Parallel support vector machines: The cascade svm. in *Advances in neural information processing systems* 521–528 (2004).
32. Altschul, S. F. *et al.* Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.* **25**, 3389–3402 (1997).
33. Li, W. & Godzik, A. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics* **22**, 1658–1659 (2006).
34. Barbosa, E., Röttger, R., Hauschild, A. C., Azevedo, V. & Baumbach, J. On the limits of computational functional genomics for bacterial lifestyle prediction. *Brief. Funct. Genomics* **13**, 398–408 (2014).
35. Kawashima, S. *et al.* AAindex: Amino acid index database, progress report 2008. *Nucleic Acids Res.* **36**, (2008).
36. Hand, D. J. & Till, R. J. A Simple Generalisation of the Area Under the ROC Curve for Multiple Class Classification Problems. *Mach. Learn.* **45**, 171–186 (2001).
37. Zhu, J., Rosset, S., Hastie, T. & Tibshirani, R. 1 -norm Support Vector Machines.
38. Pedregosa, F. *et al.* Scikit-learn: Machine Learning in Python. **12**, 2825–2830 (2012).
39. Kohavi, R. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. *Int. Jt. Conf. Artif. Intell.* **14**, 1137–1143 (1995).

40. Matthews, B. W. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *BBA - Protein Struct.* **405**, 442–451 (1975).
41. Marchler-Bauer, A. *et al.* CDD/SPARCLE: Functional classification of proteins via subfamily domain architectures. *Nucleic Acids Res.* **45**, D200–D203 (2017).

## תקציר

זיהום חיידקי הוא גורם עיקרי למחלות מסביב לעולם. עם זאת, זני חיידקים רבים אינם מהווים סכנה בריאותית לאדם, וחלקם אף מועילים לו. לכן, על מנת לשפר את הבנת מודלי ההדבקה של חיידקים הגורמים למחלות, ולטובת אבחנות מדויקות ומהירות יותר, יש צורך בכלים ביודאינפורמטיים יעילים, בעלי יכולת סיווג חיידקים, לכאלו המזיקים לאדם (חיידקים פתוגנים), ולכאלו שאינם מזיקים לאדם. בעקבות השיפור בשנים האחרונות בטכנולוגיות ריצוף גנים, עולה הזמינות של רצפים גנטיים של חיידקים. זהו אך טבעי שכלי סיווג החיידקים ישתמשו ברצפים אלו, על מנת לסווג חיידקים למזיקים לאדם ולאלו שאינם.

בעבודה זו, אני מציגים גישה מבוססת למידת-מכונה לסיווג חיידקים החיים בגוף האדם לאלו הגורמים למחלות, ולאלו שאינם. השיטה שלנו מבוססת על Sparse SVM, מסווג (classifier) הבוחר מספר מועט של גנים, ולפיהם מבצע את סיווג החיידקים. בחירה זו נעשית באופן אוטומטי, ללא התערבות אדם. אנו מממשים את השיטה ככלי, שאותו אנו מכנים "BACPACSS" – Bacterial Pathogenicity Classification via Sparse SVM. BACPACSS הוא כלי אוטומטי לחלוטין, המסוגל להתמודד עם כמות נתונים (גנומים מרוצפים של חיידקים) הגדולה משמעותית מכמות הנתונים שאיתה התמודדו הכלים הקיימים עד כה.

על מנת לייצר כלי רלוונטי לאבחון זיהומים חיידקיים באדם, השתמשנו אך ורק בגנומים של חיידקים הידועים כבעלי יכולת להתקיים בגוף האדם. בשימוש בגנומים הללו, BACPACSS מציג יכולת סיווג בעלת דיוק גבוה, בין חיידקים פתוגנים לכאלו שאינם. יתרה מזאת, השימוש בכלי מאפשר זיהוי של גנים כגנים המעניקים לחיידקים תכונות פתוגניות, למרות שעד כה מעורבותם ביכולת לחולל מחלות באדם לא הייתה ידועה.

הקוד המלא של BACPACSS, המודל המאומן המוכן, וחומרי קריאה נוספים זמינים ב <https://www.cs.bgu.ac.il/~barashe>. כמו כן, בכתובת זו נמצא גם תיאור הגנומים בהם השתמשנו, המחולק לחיידקים פתוגנים ולא פתוגנים.

אוניברסיטת בן-גוריון בנגב

הפקולטה למדעי הטבע

המחלקה למדעי המחשב

## סיווג פתוגניות של חיידקים

## באמצעות Sparse SVM

חיבור זה מהווה חלק מהדרישות לקבלת התואר "מוסמך למדעי טבע" (M.Sc.)

מאת: ערן ברש

בהנחיית: פרופ' מיכל זיו-יוקלסון וד"ר סיון סבתו

תאריך: 12/9/17	<u>Eran Barash</u>	חתימת הסטודנט:
תאריך: 12/9/17	<u>מיכל זיו-יוקלסון</u>	חתימת המנחה:
תאריך: 12/9/17	<u></u>	חתימת המנחה:
תאריך: _____	_____	חתימת יו"ר הועדה המחלקתית:

אוניברסיטת בן-גוריון בנגב

הפקולטה למדעי הטבע

המחלקה למדעי המחשב

סיווג פתוגניות של חיידקים

באמצעות Sparse SVM

חיבור זה מהווה חלק מהדרישות לקבלת התואר "מוסמך למדעי טבע" (M.Sc.)

מאת: ערן ברש

בהנחיית: פרופ' מיכל זיו-יוקלסון וד"ר סיון סבתו