PaPrBaG: A random forest approach for the detection of novel pathogens from NGS data

Carlus Deneke, Robert Rentzsch, and Bernhard Y. Renard

Research Group Bioinformatics (NG4), Robert Koch Institute, 13353 Berlin, Germany.

ABSTRACT

The reliable detection of novel bacterial pathogens from next generation sequencing data is a key challenge for microbial diagnostics. Current computational tools usually rely on sequence similarity and often fail to detect novel species when closely related genomes are unavailable or missing from the reference database used.
Here, we present the machine learning based approach PaPrBaG (Pathogenicity Prediction for Bacterial Genomes). PaPrBaG overcomes genetic divergence by training on a wide range of species with known pathogenicity phenotype. To that end we compiled a comprehensive list of pathogenic and non-pathogenic bacteria, using a rule-based protocol to annotate pathogenicity based on genome metadata. A detailed comparative study reveals that PaPrBaG has several advantages over sequence similarity approaches. Most importantly, it always provides a prediction whereas other approaches discard a large number of sequencing reads that are far away from currently known reference genomes. Furthermore, PaPrBaG remains reliable even at very low genomic coverages. Combining PaPrBaG with existing approaches further improves prediction results.

Keywords: Pathogenicity prediction, Analysis of NGS data, Applied machine learning

1 INTRODUCTION

The vast amount and diversity of bacteria on Earth, together with ever increasing human exposure Vouga and Greub (2016), suggests that we will be continuously confronted with novel bacterial pathogens, too. The identification of novel strains or even species with pathogenic potential directly from NGS sequencing data has so far been problematic when no closely related genomes are known or when these are missing from the used reference database. The availability of high-throughput sequencing technology and increasingly comprehensive microbial genome databases makes it possible to detect putative novel pathogens solely based on sequence. This is true even given that pathogenesis is ultimately governed by the interplay of host (state) and pathogen and therefore one may better speak of *pathogenic potential* when referring to e.g. a specific bacterium. Existing methods amenable to pathogenicity prediction broadly fall into two classes: protein content based and whole-genome based.

Protein content methods Where assembled genomes are available, the presence/absence pattern of certain protein families can be expected to correlate with organism phenotypes like pathogenicity. This is primarily based on the presence of virulence factors (VFs) - often acquired through horizontal gene transfer Juhas (2015) - or the absence of more common genes (functions) that become dispensable when, e.g., host-specific pathogens evolve from commensal ancestors Merhej et al. (2013). Three recent studies rely on these considerations.

The BacFier method by Iraola et al. Iraola et al. (2012) was the first to apply the described approach on a large scale. The authors defined eight VF categories and obtained 814 related VF protein families from KEGG Kanehisa et al. (2014). They further used a set of 848 human-pathogenic (HP) and non-pathogenic (NP) genomes broadly covering bacterial taxonomy. Using a support vector machine (SVM) based approach, they subsequently selected the most discriminative subset of all 814 features (presence/absence of a given family in a given genome) through cross-validation.

For PathogenFinder, Cosentino and coworkers Cosentino et al. (2013) compiled a list of 1,334 genomes with available pathogenicity information. They clustered all proteins from those 885 genomes that were published before November 2010 into tens of thousands of families using CD-HIT Fu et al. (2012). Those significantly enriched in either HPs or non-HPs were assigned a signed weight value depending on the degree of enrichment. The 449 genomes published later formed the test set. Their phenotype was predicted by assigning the encoded proteins to the previously generated families and summing up the associated weights, respectively.

In a more focused and qualitative study, Barbosa and colleagues Barbosa et al. (2014) used a manually labelled set of 240 actinobacterial genomes and identified just under 30,000 protein families using their own Transitivity Clustering method Röttger et al. (2013). The authors further distinguished between HPs, broad-spectrum animal pathogens, opportunistic human pathogens and NPs.

Whole-genome methods Pathogenicity may also be predicted using a range of established tools that were originally developed for mapping NGS reads to reference genomes and/or classifying them taxonomically. In doing so, the likelihood of pathogenicity increases with proximity to a pathogenic reference. While the challenge of detecting known pathogens in mixed (e.g. clinical) samples is very much related to general metagenomic analysis workflows Miller et al. (2013); Mande et al. (2012); Lindner and Renard (2015), these methods have rarely been used for predicting the presence of novel pathogens. While classic alignment tools like BLAST Altschul et al. (1990) may be used for mapping with high sensitivity but relatively low throughput, the opposite applies to dedicated read mappers such as Bowtie2 Langmead and Salzberg (2012) and BWA Li and Durbin (2009), whose performance deteriorates quickly for highly divergent query strains or even novel species. Still, the latter are widely used in non-predictive NGS analysis pipelines. PathoScope Francis et al. (2013); Hong et al. (2014), for example, provides a

statistical filtering scheme to resolve mapping conflicts, i.e. reads mapping to different references. Clinical PathoScope Byrd et al. (2014) is particularly useful to remove large numbers of contaminant reads, e.g. human ones in the context of clinical samples. Instead of using existing mappers, the SURPI pipeline Naccache et al. (2014) relies on two custom-built tools for nucleotide and BLASTX-like translated alignment. Both are shown to scale better with data set size compared to their conventional competitors whilst maintaining similar performance. The translated alignment step with *de-novo* assembled contigs is reported to increase sensitivity particularly in the viral domain.

Composition-based methods compare the distributions of different compositional features (usually k-mer occurrence or frequencies) in the query and reference sequences. Kraken Wood and Salzberg (2014), for example, tries to match 31-mers found in the query to a precomputed database, which maps those sequences to the lowest common ancestor taxon of all reference genomes they occur in, respectively. Somewhat related, MetaPhlAn Segata et al. (2012) first creates a compact database of clade-specific marker genes (with clades ranging from strain to phylum level), which it then uses for prediction. NBC Rosen et al. (2008, 2011) calculates k-mer frequency profiles of all references and uses them to train a naïve Bayesian classifier. Other machine learning approaches use kernelised nearest neighbour Diaz et al. (2009) or hierarchical structured-output SVMs McHardy et al. (2007); Patil et al. (2012). Hogan et al. Hogan et al. (2013) trained binary classifiers on two reference groups (e.g. two phyla or one vs. all others). In this case, a database of 'competing' classifiers must be built for wide taxonomic coverage.

Motivation and aims The existing approaches exemplarily outlined above have, like any method, different strengths and weaknesses depending on the specific usage scenario. In brief, ours is the fast, robust and user-friendly estimation of pathogenic potential based on raw NGS data from newly discovered bacterial strains or species with potentially large sequence divergence. Note that the latter is not an uncommon event: e.g., Schlaberg and colleagues Schlaberg et al. (2012) identified 673 isolates that belong to 'as-yet-undescribed' species. More recently, sequencing of bacterial isolates from patients in an intensive care unit led to the discovery of 428 potential novel species within a single year Roach et al. (2015).

While protein content based methods show great potential for not only prediction but also qualitative analyses (e.g. pinpointing clade-specific VFs), and even for identifying yet uncharacterised VFs (as illustrated in the above-cited publications), their primary, shared drawback is the dependence on genome assembly and annotation. These steps are both time-consuming and, particularly in novel-species and/or low-coverage scenarios, error-prone. Further, these methods neglect the signal potentially found outside of protein-coding genes. While dedicated read mappers do not share these problems, they may still struggle with highly divergent strains or (even) novel species; in turn, this impacts frameworks like PathoScope. The same applies to methods depending on long, gapless k-mer matches, like Kraken and NBC. BLAST, on the other hand, is generally considered too slow for large-scale read mapping. Finally, we are not aware of any previous studies using a read-based machine learning approach for pathogenicity prediction, in conjunction with a comprehensive evaluation.

Given our above-stated scenario, however, more fundamental differences exist between these genome-based methods and what we were aiming for. They are (i) heavily influenced by the taxonomic coverage of the underlying data sets, (ii) make taxonomic instead of phenotypical predictions, and (iii) are not designed to make predictions per se (but rather identify already known organisms). In summary, while all these methods are highly useful in different contexts, they do not necessarily fit the task at hand. Therefore, we developed PaPrBaG: Pathogenicity Prediction for Bacterial Genomes. In an important preliminary step, we compiled a comprehensive set of genomic data and metadata. Based on the latter, we then established a system of rules to automatically identify human-pathogenic and human-non-pathogenic bacteria. For the prediction task, we introduced several new compositional features and used them for training as well as querying a binary random forest classifier. These are known to be fast, error-tolerant and capable of dealing with a large number of features. Finally, we provide a solid evaluation, also comparing against other types of methods. This may serve as a guideline for users to select the most appropriate method for a given task, e.g. in clinical settings. A user-friendly R package is additionally provided at https://github.com/crarlus/paprbag.

2 DATA

No comprehensive standard resource listing bacterial strains with or without pathogenic potential in human is publicly available. However, the Integrated Microbial Genomes (IMG) system collects a wide range of metadata on microbial genome projects Markowitz et al. (2014). We accessed the IMG web site on 04/06/2015 and downloaded a table containing all available data. In a first step we pre-filtered the IMG data for the key *Bacteria* in the field *Domain*, for *Finished* or *Permanent draft* in *Status* and for *Genome Analysis* in *Project Type*; the latter serves to exclude metagenomic studies. We furthermore excluded any genomes for species marked as *unclassified*.

In the following, we describe a set of rules to infer reliable labels for HPs and non-HPs. To infer the pathogen label we search for entries that contain the term *Pathogen* in the fields *Phenotype* or *Relevance*. Additionally, all genomes that contain an entry in the field *Diseases* are labelled as pathogenic. We inferred non-pathogens by searching for the keyword *Non-pathogen* in the field *Phenotype*. Note that no further field clearly designates non-pathogens. In particular, from a missing entry in the field *Disease*, it does not follow that the organism is not a pathogen. The same holds for (reference) genomes that were sequenced as part of the Human Microbiome Project (HMP), since those include both pathogens and commensals Human Microbiome Project Consortium (2012a,b). If contradicting rules were met for an entry, e.g. non-pathogenic phenotype yet still an annotated disease, the label 'unknown' was assigned and the genome excluded from further analysis.

For the present study, and with a clinical setting in mind, we were interested in human pathogens and non-pathogens only. Therefore, including e.g. plant pathogens or non-pathogenic soil bacteria could result in misleading conclusions (those could be used in analogous studies for other habitats and hosts, though). Bacteria with human host were identified using the following set of rules: either the entries *human* or *Homo Sapiens* are found in the fields *Host Name*, *Ecosystem Category* or *Habitat*; or the field *Study Name* contains the entry *HMP*. For further analysis we kept all entries with human host and either pathogenic or non-pathogenic phenotype.

We finally obtained labels for 2,836 bacterial strains (177 non-pathogens and 2,659 pathogens). These belong to 422 different species. On the species level, we found 363 pure pathogens and 53 pure non-pathogens. For 6 species, we found that labels were mixed between strains. Most strikingly, *Escherichia coli* comprises 172 pathogenic and 93 non-pathogenic strains. For five other

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2379v1 | CC BY 4.0 Open Access | rec: 19 Aug 2016, publ: 19 Aug 2016

species, *Campylobacter jejuni*, *Listeria monocytogenes*, *Staphylococcus epidermidis*, *Peptoclostridium difficile* and *Clostridium botulinum*, we found one or two non-pathogenic strains vs. many pathogenic strains. However, these species are commonly known as pathogens and therefore the non-pathogenic entries were excluded from further analysis. The resulting table of label data is provided in the supplied R package.

3 METHODS

Approach Figure 1 summarises the individual steps of PaPrBaG. The supervised machine learning setup consists of a training and a prediction workflow. The entire set of HP and non-HP bacterial species is divided into non-overlapping training and test sets. Subsequently, selected genomes from all species are fragmented into reads (see section 3.1), from which a range of sequence features are extracted (section 3.2). The training sequence features together with the associated phenotype labels compose the training database, on which the random forest algorithm trains a pathogenicity classifier (section 3.3). In turn, this classifier predicts the pathogenic potential for each read in the test set. Based on these raw results various analysis steps can be performed. This section further provides a summary of the different benchmark approaches (3.4) and evaluation strategies (3.5) used in the Results section.

3.1 Training and test data

This work is based on the analysis of the newly generated collection of pathogenic and non-pathogenic bacterial strains (see section 2). For all labeled genomes, FASTA files were obtained from the NCBI Entrez Benson et al. (2013) database via queries using the NCBI *Bioproject accessions* (on August 24, 2015).

To evaluate the classifier on independent test data, we performed a randomised five fold cross-validation study. Therefore, we randomly distributed unique pathogenic and non-pathogenic species into five non-overlapping parts, preserving the original label distribution. The number of strains per species varies from 1 to more than 200, particularly pathogens have been studied at much greater breadth. This imbalance would translate into a largely skewed training database. Therefore, we kept only one random strain per species for further analysis. Apart from markedly reducing the label imbalance (the ratio decreases from 16 to 7), this also reduces the training data size. This approach also reflects the scope of this article, which is predicting phenotypes on the species level. To evaluate possible effects of this sub-sampling strategy, we included all strains of each training species in a separate benchmark study (see Results). *E. coli* played a unique role in that it possesses a large number of pathogenic and non-pathogenic strains. For the current analysis, we considered the pathogenic strains of *E. coli* only. Since our aim was to provide species-level predictions, the rationale was to be more sensitive towards pathogens.

For both the training and test data sets, we simulated 250 bp long Illumina reads. To that end we used the Mason read simulator with the default Illumina error model Holtgrewe (2010). The number of reads sampled for each genome differs for the training and test sets. The ratio of pathogens and non-pathogens in each training fold is about 7:1. However, for binary classification tasks it is advantageous to show the learner an equal number of examples for both classes. Therefore, we decided to sample the same total number of reads per class. For the present study we chose 10^6 reads per class, which represent a trade-off between genome coverage and training data size. An increase of the training size to 10^7 did not substantially improve prediction results. The number of reads per genome in each class was chosen such that each genome has the same coverage, i.e. proportional to the size of the genome. Conversely, for the test data sets, we chose to sample up to a coverage of approximately 1 for each genome. The read simulation was repeated for each fold.

3.2 Features

For the machine learning task, a set of informative features must be extracted from the read sequences. We implemented a number of different feature types to capture the information content present in a sequencing read.

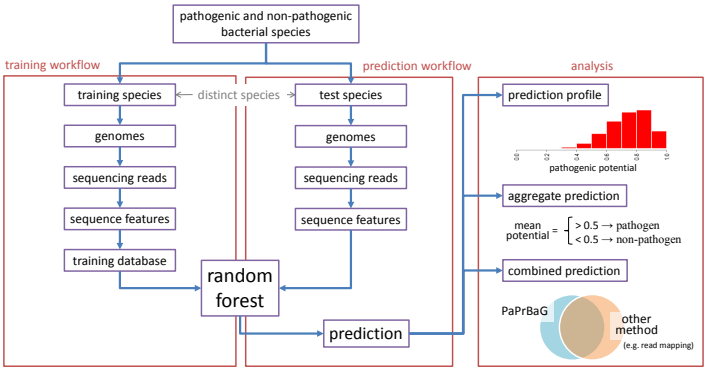

Figure 1. Overview of PaPrBaG workflow. Reads are simulated from genomes in both the training (left) and prediction workflow (center), from which features are extracted. The training sequence features together with the associated phenotype labels, compose the training database, on which the random forest algorithm trains a pathogenicity classifier. This classifier predicts the pathogenic potential for each read in the test set. From these raw results, the prediction profile, the genome aggregate prediction and a combined prediction can be generated (right).

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2379v1 | CC BY 4.0 Open Access | rec: 19 Aug 2016, publ: 19 Aug 2016

Genomic features Different features can be extracted from DNA sequences. They are all based on sequences based on k-mer occurrence patterns. Since we analyze read data, strand information is not available. Therefore, we cast all features symmetrically so that the occurrence of a word and its reverse-complement considered jointly, a common strategy in related methods Melsted and Pritchard (2011); Marçais and Kingsford (2011).

A first features type is the relative k-mer frequency. We found that including monomers, dimers, trimers and tetramers led to good results, but higher values of k did not lead to further improvement. The occurrences of longer k-mers are less likely to overlap among highly divergent sequences. Conversely, the consideration of a large number of uninformative long k-mers can compromise prediction performance. However, as focussing on selected longer sequence motifs can still be beneficial for classification, we also recorded the frequencies of the 100 most abundant 8-mers in an independent set of bacterial genomes. More precisely, we scanned both possible sequence strands and allowed for one mismatch. *Spaced words* were introduced for the alignment of dissimilar sequences Ma et al. (2002); Leimeister et al. (2014). Thus, their incorporation is useful in the context of novel species discovery. Spaced words denote the occurrence of all k-mers interrupted by *(l-k)* spacers in a word of length *l*. For this analysis we searched for all symmetric 4-mers in a spaced word of length 6.

Protein features Bacterial genomes are known to be densely packed with proteins Patthy (1999). Since protein sequences are evolutionarily more conserved than DNA sequences, peptide features can provide additional valuable information. A read might (partially) cover a protein sequence, but the correct reading frame is unknown. However, longer DNA sequences tend to contain frequent stop codons in the anti-sense frames by chance. Therefore, as a simple heuristic, we generally used the frame and strand with the fewest number of stop codons. This frame was translated into a peptide sequence and several types of features were extracted: codon frequencies, relative monopeptide and dipeptide frequencies, amino acid properties and Amino Acid Index (AAIndex) Nakai et al. (1988); Tomii and Kanehisa (1996); Kawashima and Kanehisa (2000) statistics. The amino acid property features consist of the relative frequencies of tiny, small, aliphatic, aromatic, non-polar, polar, charged, basic and acidic residues Creighton (1993). Finally, the AAIndex assigns scores for diverse properties (often based on peptide secondary structure) to each residue. From 544 indices, we selected the 32 with the lowest pairwise correlation. Features were obtained by computing the product of the amino acid frequencies and their associated index scores. In total, we included 948 features in our classification workflow.

Feature importance As measured by both the permutation and Gini tests, the most important features come from the DNA monomer, dimer and trimer feature groups. Among the 100 most important features, the tetramer, codon frequency, AAIndex and spaced words groups are also prevalent. We estimated the importance of the different groups by searching for the highest scoring member of each group. The resulting order of group importance was trimer, monomer, dimer, tetramer, spaced words, AAIndex score, codon frequencies, monopeptides, DNA motifs, amino acid properties and dipeptides. Particularly the last 5 groups were of minor importance for the classification task.

3.3 Machine Learning

A random forest classifier Breiman (2001) was trained using the above-described features and (genome) pathogenicity labels for each read in the training data set. We chose this classifier type because it combines high accuracy, fast prediction speed and the capability to deal with noisy data Folleco et al. (2008b,a). Among the different implementations of the random forest algorithm available, we opted for *ranger* Malley et al. (2012); Wright and Ziegler (2015) since it is one of the fastest and can handle large data sets. We used probability forests, which return the fraction of votes for each class. This can also be interpreted as the prediction probability. We refer to the prediction probability of the pathogenic class as the *pathogenic potential* of a read. Another advantage of random forest is that it has only few tunable parameters. We found that it is sufficient to train 100 per forest and that more trees do not lead to better predictions. We further adjusted the minimum size for terminal nodes. High numbers can result in impure terminal nodes and smaller trees. Changing it from 1 to 10 had no effect, while sizes above 1000 led to overfitting. Changing other parameters had no substantial effect. The trained random forest objects are available on github.

3.4 Benchmark configuration

We compared the performance of PaPrBaG with a range of other tools, most of which were originally developed for taxonomic classification. We used Bowtie2 Langmead and Salzberg (2012) as one of the commonly used read mappers that combines speed and accuracy Hatem et al. (2013). Furthermore, we considered Pathoscope2 Hong et al. (2014) as a dedicated pipeline for pathogen identification. More sensitive mapping is expected from BLAST Altschul et al. (1990), which is still widely used in NGS pipelines. As a candidate for composition-based methods, we chose Kraken, which has emerged as one of the primary taxonomic classification tools Wood and Salzberg (2014). Finally, we considered NBC as a composition-based machine learning method Rosen et al. (2008, 2011). It is advantageous over similar approaches in that it allows the construction of a custom training database. We evaluated the performance of these tools using the PaPrBaG training genomes and test sets, again using five fold cross-validation.

Bowtie2 For read mapping, we used Bowtie2 (v2.2.4) in the *very-sensitive* configuration, which is highly tolerant towards mismatches and gaps. We obtained the 50 top alignments of each read. Parsing the resulting SAM file, we matched the best-scoring mapping for each read against the label database. When more than one alignment shared the best score, we chose a match to a pathogen over a match to a non-pathogen. For unmapped reads, no prediction could be made. Additionally, we repeated this mapping workflow for a larger reference genome set that included all strains of the training species.

Pathoscope2 Pathoscope2 (v2.0.6) works as a post-mapping filter. Hence, we ran the Pathoscope2 ID Module on the SAM file produced by our Bowtie2 read mapping. The resulting filtered SAM file was analysed as above to obtain label predictions. Also, the Pathoscope2 workflow was repeated with the larger reference data set containing all strains.

Kraken We provided Kraken (v0.10.5) with the training genome sequences, from which it builds a database based on 31-mers and generates a taxonomic tree. Based on this, the tool classifies each read taxonomically and returns an NCBI taxonomy id, which can be translated into the corresponding name using the translation module. The predicted label can now either be inferred

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2379v1 | CC BY 4.0 Open Access | rec: 19 Aug 2016, publ: 19 Aug 2016

from the NCBI taxonomy id or by matching the classified species to the label database. In case Kraken's prediction was not at species resolution, no prediction was made. Since matching 31-mers to divergent sequences might be a difficult challenge, we also repeated the entire analysis using 16-mers (Kraken-16).

BLAST We ran NCBI BLAST (v2.2.28) with the option '-task dc-megaBLAST', which is tailored for inter-species comparisons. Additionally, we chose an E-value cutoff of 10. From the resulting BLAST output, the highest-scoring target was matched to the reference label database.

Naïve Bayes Classifier We created a set of NBC (v1.0) training databases with word length 15 and then scored all test read sets against all training databases. For each read, we selected the highest-scoring hit and matched the species name to the label reference database. Since classification with NBC took very long, we had to use parallel threads.

3.5 Evaluation metrics

Majority prediction rule All tested methods return a prediction for each read, but ultimately we are interested in one integrated prediction for each genome. A single read matching to a pathogen is not by itself deemed significant, given that also non-pathogen genomes may contain stretches showing similarity to pathogen genomes. Therefore, a straight-forward integration scheme weighs the evidence for the presence of a pathogen versus a non-pathogen. In PaPrBaG we average over all read-based prediction probabilities. If this value exceeds 0.5, the organism is classified as pathogenic. Likewise, for the other methods, if the number of reads mapped to a pathogen exceeds the number of reads mapped to a non-pathogen, the organism is classified as pathogenic. This evaluation metric will henceforth be referred to as *majority prediction rule*.

Minimum detection threshold The *majority prediction rule* allows for a simple estimation of the pathogenic potential of a sample; however, it completely ignores uncertainty due to missing predictions. Therefore, we also use a complementary metric, the *minimum detection threshold*. Here, a user can define the minimum fraction of reads that should be required for a confident phenotype prediction. As before, for a given test genome we collect the read-based evidence. If the number of reads supporting a phenotype exceeds the minimum detection threshold, a prediction is made accordingly. If both phenotypes are supported, that with higher support determines the prediction.

For both phenotypes, we assess the fraction of correct predictions, which corresponds to the true positive rate (TPR) and true negative rate (TNR), respectively. We then summarise the performance using *informedness*, also known as Youden's J statistic, which is a joined measure of specificity and sensitivity Youden (1950). Formally, informedness is defined as $I = \text{TPR} + \text{TNR} - 1$ and ranges from -1 (only wrong predictions) to 1 (only correct predictions).

The optimal value of the *minimum detection threshold* is not known a-priori and depends on the particular experimental settings. Therefore, we vary the threshold from 0 to 1.

Consensus filter Individual approaches may yield heterogeneous predictions, which makes it attractive to combine them to enhance prediction confidence. We therefore define a *consensus filter* as follows: In a first step, we evaluate which predictions coincide between two methods. We then keep only the consensus subset for further performance evaluation.

Prediction certainty Each prediction made by the *majority prediction rule* is associated with uncertainty. We define the prediction certainty as $|\mu - 0.5| \times 2$, where μ denotes the majority prediction as discussed above. We further normalise the certainty of each predictor by the highest certainty it reports for any genome. The result is a relative certainty value that always ranges from 0 (maximally uncertain) to 1 (maximally certain). Note that this value does not reflect the type of prediction (pathogen or non-pathogen). Normalisation is not a necessary step but aids visualisation in Figure 4.

4 RESULTS

In the following, we discuss the results of a five-fold cross validation study on the entire data set of pathogenic and non-pathogenic species and compare the performance of PaPrBaG with that of the other methods tested. Training the PaPrBaG classifier led to promising results that could not be notably improved by further parameter tuning or feature selection efforts. Across all cross-validation folds, the out-of-bag training error was 0.24 and the error of the (imbalanced) test data set was 0.22. Furthermore, the area-under-curve (AUC) of the training reads was 0.84 and of the test reads 0.79. Hence, the classification problem generalised well to independent data. The degree of certainty of a read prediction can be measured by the prediction probability. Certainty increases continuously from noisy predictions at probabilities around 0.5 to very accurate predictions at probabilities close to 0 or 1. Thus, we could confirm that prediction probability is indeed related to prediction certainty.

Comparison of read information content

Each method initially provides predictions for all individual reads. In PaPrBaG, the majority of trees either votes for a pathogenic or non-pathogenic origin of a given read. For the other tools, the prediction is either a match to a pathogen, a non-pathogen or no match at all. An overview of the per-read results are given in Figure 2. PaPrBaG always makes a prediction, but numerous false negative and false positive predictions exist. Bowtie2, Pathoscope2 and Kraken can only make predictions over a minority of all reads. Kraken-16 and BLAST are able to map the majority of reads, but still leave a considerable fraction unmapped. All other methods also make false predictions, in particular false positives. As the bottom plot reveals, there are almost as many false positive as true negative predictions. This problem reflects the imbalanced training data set, which has been addressed explicitly in the design of PaPrBaG.

Phenotype prediction by majority vote

Ultimately, the goal of all approaches discussed here is the inference of an organism's phenotype directly from sequencing reads. Due to the high number of false predictions in all approaches (see Figure 2), the presence of a single read matching to a pathogen is obviously not a sufficient criterion for overall phenotype prediction. An elementary prediction metric was introduced by the *majority prediction rule* in section Methods. It compares the amount of read evidence for the presence of a pathogen and a

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2379v1 | CC BY 4.0 Open Access | rec: 19 Aug 2016, publ: 19 Aug 2016

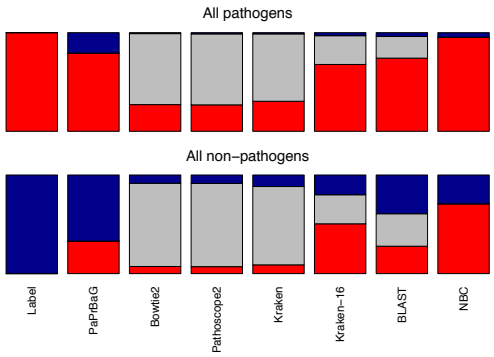

Figure 2. Read predictions for all pathogens (top) and all non-pathogens (bottom). Each bar shows the number of reads predicted to be of pathogenic (red), non-pathogenic (blue) or unknown (gray) origin. The left-most bars show the ground truth. Strikingly, Bowtie2, Pathoscope2 and Kraken fail to classify the majority of the reads. Kraken-16 and BLAST still miss a considerable fraction of reads whereas the machine-learning based approaches always return a prediction. All methods show true and false predictions to a varying extent. While PaPrBaG shows similar errors for both pathogens and non-pathogens, all other methods suffer from a substantial bias. Few reads from pathogens are falsely classified as non-pathogenic. Conversely, in the case of non-pathogens, the number of falsely classified reads is similar to or even exceeds the number of correctly classified reads.

	TPR	TNR	ACC	F1	MCC
PaPrBaG	0.91	0.70	0.88	0.93	0.54
Bowtie2	0.95	0.66	0.91	0.95	0.61
Pathoscope2	0.94	**0.72**	0.91	0.95	0.62
Kraken	0.97	0.64	**0.93**	**0.96**	**0.66**
Kraken-16	**0.99**	0.19	0.89	0.94	0.37
BLAST	0.96	0.60	0.92	0.95	0.61
Bowtie2 $_{\text{All strains}}$	0.96	0.60	0.91	0.95	0.58
Pathoscope2 $_{\text{All strains}}$	0.96	0.66	0.92	0.95	0.63
NBC	0.99	0.23	0.90	0.94	0.41
Bowtie2 + PaPrBaG	0.97	0.77	0.95	0.97	0.71
Pathoscope2 + PaPrBaG	0.97	**0.81**	0.95	0.97	0.74
BLAST+ PaPrBaG	0.97	0.74	0.95	0.97	0.70
Kraken + PaPrBaG	0.97	0.76	0.95	0.98	0.73
Pathoscope2 + Kraken	0.97	0.69	0.94	0.97	0.70
Pathoscope2 + NBC	**0.99**	0.44	0.95	0.98	0.60
Bowtie2 + Kraken + PaPrBaG	0.98	0.78	**0.96**	**0.98**	**0.75**

Table 1. Prediction statistics for majority prediction rule (TPR = True positive rate, TNR = True negative rate, ACC = Accuracy, F1 = F1-score, MCC = Matthews-correlation coefficient). The first set of entries shows the performance of the individual methods. Bowtie *All Strains* and Pathoscope *All Strains* represent a variation where the reference data set contains all strains of a species in the training set. Below the horizontal line, we show results for the combination of methods with the *consensus filter*. In these cases, the performance is given for those genomes that have predictions agreeing between two or more individual classifiers. Overall, combining PaPrBaG with Bowtie2 and Kraken yields the best performance.

non-pathogen and assigns the better-supported phenotype. Note that, in this classification scheme, the uncertainty originating from the large number of unmapped reads has been ignored. Hence, conclusions are drawn based on the information from an average of 6% (Bowtie2, Pathoscope2), 14 % (Kraken) and 78 % (BLAST) of all available reads. For Bowtie2 and Pathoscope2, 89 test sets have less than 100 mapped reads and two produce no mapped read at all. Conversely, PaPrBaG and NBC provide predictions for all reads. We discuss a different performance metric that explicitly considers the unmapped reads in the *minimum detection threshold* evaluation below.

The classification results for all organisms are shown in Table 1. Most organisms are classified correctly by all methods, with accuracy values ranging from 0.88 to 0.93. This demonstrates that it is indeed possible to infer the pathogenicity phenotype of a novel species solely based on sequencing data. Further, the different methods show different degrees of specificity and sensitivity. Since the test data set is largely imbalanced, the *Matthews Correlation Coefficient* (MCC) is more appropriate to compare the performance between different methods. As Table 1 shows, Kraken performs best followed by Pathoscope2, Bowtie2, BLAST and PaPrBaG. Kraken-16 and NBC yield strongly biased predictions and have a lower MCC. We additionally evaluated the performance of Bowtie2 and Pathoscope2 with a larger reference database containing all strains of all training species. Here, the classifications become more sensitive and less specific, which reflects the larger bias towards pathogens in the training set. Overall, the effect of the larger database is small. Another interesting question is whether or not PaPrBaG can predict the correct phenotype in cases where closely related species have a different phenotype. We therefore reassessed the cross-validation results for all test species found in genera with both HP and non-HP members (in the training data set of the respective fold). Across all folds, we obtain an accuracy of 90 % (46/51 species). Similar performance is observed in a second test, focusing on cases where the training data set does not contain any member of the same genus (89 %, 71/80). For the most difficult scenario - all training species from the same genus having the opposite label - PaPrBaG can correctly predict the phenotype in only 2 out of 9 cases, though. Still, overall this suggests that PaPrBaG, in most cases, makes correct predictions even for closely related species from genera with mixed phenotype.

Consensus filter

The heterogeneous prediction results of the individual classifiers suggest it might be worthwhile to combine them. Accordingly, we have introduced the *consensus filter* in section Methods. It filters and evaluates predictions that coincide between different classifiers. The lower part of Table 1 shows the performance of selected combinations of methods. Combining PaPrBaG with either Bowtie2, Pathscope2, Kraken or BLAST leads to a substantial improvement of classification over any of the individual methods. We find accuracy values above 0.95 and MCC values above 0.7. Combining PaPrBaG with Bowtie2 and Kraken achieves the highest performance, closely followed by PaPrBaG with either Kraken or Pathoscope. The single best combination without PaPrBaG, Pathoscope2 + Kraken, yields good results, but is outperformed by the combinations including PaPrBaG. The combination Pathoscope2 + NBC has a lower MCC than Pathoscope2 alone. Other combinations of Bowtie2, Pathoscope2, Kraken, BLAST and NBC without PaPrBaG showed no substantial improvements and have been omitted from Table 1. As

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2379v1 | CC BY 4.0 Open Access | rec: 19 Aug 2016, publ: 19 Aug 2016

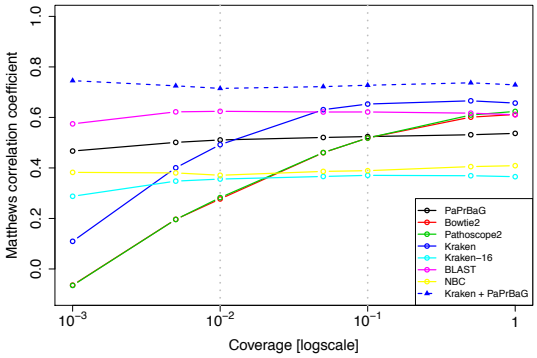

Figure 3. Classification performance for different genome coverages. As coverage decreases, so does the performance of Bowtie2, Pathoscope2 and Kraken. Conversely, BLAST and PaPrBaG still deliver sound results at coverages as low as 0.001. The triangles show results for the consensus filter when combining PaPrBaG and Kraken. It achieves high performances at all coverage levels, however, at the cost of filtering out more and more data.

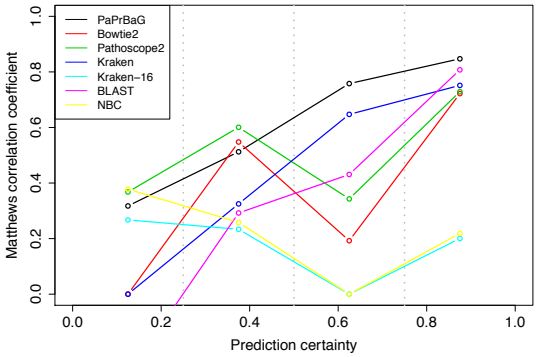

Figure 4. Fidelity of prediction certainty. Each prediction is associated with uncertainty. Here, we pooled predictions within each certainty interval and measured the prediction performance (MCC). PaPrBaG, Kraken and BLAST show a steady increase in performance with increasing certainty. PaPrBaG achieves the highest MCC among all methods compared.

the read mapping tools make highly overlapping predictions, they also tend to make the same errors. Conversely, PaPrBaG behaves differently and makes unique predictions. In conclusion, although combining two or more classifiers does not increase the overall performance, it increases prediction confidence for a subset of the data. Therefore, it is favorable to perform different and heterogeneous classification steps when maximum confidence is desired.

Coverage dependency

The results discussed so far were based on test genomes sequenced with a coverage of 1. However, in an experimental situation the coverage may be well below 1, in particular for metagenomic data. In the following, we elucidate how classification performance depends on the coverage of the test genomes. Since fluctuations may play a more important role for low coverages, we averaged over 100 simulation repeats. The corresponding results are shown in Figure 3. The performance of BLAST and PaPrBaG is rather stable over the entire range of coverages. Both are still reasonably sensitive even at extremely low coverages of about 0.001. The same holds for Kraken-16 and NBC albeit with a lower MCC across all coverages. Kraken's performance substantially decreases for coverages lower than 0.05. Bowtie2 and Pathoscope2 only work well for high coverages; below 0.1, their performance decreases rapidly. As discussed above, for Kraken, Bowtie2 and Pathoscope2 only a small amount of reads can be mapped at all at a coverage of 1. Hence, a reduction of the number of reads means that it becomes more and more likely that no read can be mapped to the reference at all. Consequently, their performance drops to the noise level. Also shown are the results obtained after applying the *consensus filter*. Combining Kraken and PaPrBaG leads to confident predictions at all coverage levels.

Prediction certainty

Each prediction made by the *majority prediction rule* is associated with a confidence. We can quantify this as *prediction certainty*, as explained in the section Methods. Figure 4 shows the performance of the different classifiers at different certainty levels. It reveals that the performances of PaPrBaG, Kraken and BLAST increase strongly with prediction certainty. Predictions of PaPrBaG with certainty values between 0.75 and 1 achieve the highest MCC of 0.85. Note moreover that for the other approaches the prediction certainty is almost always found at high values. Thus, for these approaches there is a smaller performance gain when comparing very certain to average predictions. Hence, we can conclude that a high prediction certainty is related to a particularly high prediction performance for PaPrBaG.

Minimum detection threshold

All performance evaluations above were based on the *majority prediction rule*. There, the overall prediction is determined by the majority of the individual read predictions. However, the basis for these predictions can be small: for the read mappers only a few hundred reads (a few percent of all reads) may map to any of the reference genomes. In these cases, e.g. a small number of contaminant reads may falsify the prediction result. Therefore, we further studied the effect of varying the *minimum detection threshold*. Generally, choosing a higher threshold should lead to increased prediction confidence. Figure 5 summarises results in terms of informedness. For low detection thresholds most methods attain high sensitivity and specificity, and therefore high informedness. However, for detection thresholds around 0.1, requiring 10 % of all reads to support a phenotype, only PaPrBaG and BLAST reach an informedness above 0.5. Conversely, the informedness of Bowtie2, Pathoscope2 and Kraken drops below 0, i.e. their predictions are at noise level. Increasing the detection threshold further, fewer and fewer predictions can be made and eventually the informedness of all methods approaches -1. However, at most threshold levels, PaPrBaG exhibits the highest

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2379v1 | CC BY 4.0 Open Access | rec: 19 Aug 2016, publ: 19 Aug 2016

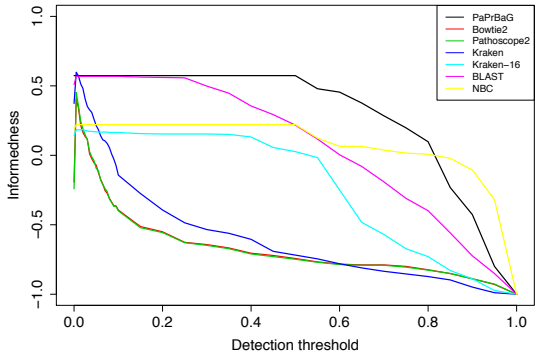

Figure 5. Classification with minimum detection threshold. Predictions are only made for read sets where the read evidence supporting a phenotype exceeds the detection threshold (given relative to the total number of reads). Initially, most approaches show high informedness, which is a joint measure of sensitivity and specificity defined as $I = \text{TPR} + \text{TNR} - 1$. As the detection threshold is increased above 0.1, the methods Bowtie2, Pathoscope and Kraken yield insufficient numbers of reads with phenotype evidence and they are no longer informative. Only PaPrBaG and BLAST show an informedness above 0.5. For most values of the detection threshold, PaPrBaG remains the method with the highest informedness.

Method	Pre-processing	Prediction	Post-processing
PaPrBaG	180	29	0
Bowtie2	0	14	78
Pathoscope2	0	15	2
Bowtie2 All Strains	0	165	105
Pathoscope2 All Strains	0	169	12
Kraken	990	62	33
Kraken-16	1833	18	48
BLAST	0	498	1
NBC	0	13901	311

Table 2. Comparison of run times. All tools except NBC were run in single-threaded mode on an SMP machine with 48 cores and 256 GB RAM. Given are the median times (in seconds) for a complete genome prediction as well as for the required pre- and post-processing steps. Bowtie2 and Pathoscope2 are the fastest methods, followed by Kraken and PaPrBaG. Note that Kraken takes particularly long to load its database.

informedness. Hence, when it is desired that a high number of reads support a phenotype, PaPrBaG is the most informative method.

Prediction run times

Table 2 lists the median run times for a complete genome prediction, respectively. While prediction with PaPrBaG is relatively fast, feature extraction and loading the trained random forest consumes a considerable amount of time. The fastest method is read mapping with Bowtie2. However, post-processing the mapped reads takes time. Note that this step is not part of Bowtie2 itself and hence has not been optimised for speed. Pathoscope2 requires additional filtering of read mapping results, which leads to faster post-processing. Mapping reads to the larger reference database containing all training strains increases the run times considerably. Prediction with Kraken takes unexpectedly long. It benefits from its high speed only for larger read sets. Note that pre-processing here includes the time-consuming step of loading the Kraken database, as well as the the relatively slow translation module. Finally, BLAST is relatively slow, and NBC is the slowest method by far.

5 DISCUSSION

In this contribution, we investigated the potential of predicting the phenotype of unknown pathogenic and non-pathogenic species directly from sequencing reads. To that end, we developed a novel method that combines feature extraction with random forest prediction, PaPrBaG. Furthermore we generated a new data set of bacterial genomes for which we could infer reliable pathogenicity information via a rules-based protocol. PaPrBaG as well as several other alignment-based and compositional approaches were extensively tested on this new data set. We evaluated the performance of all methods under the majority rule and with flexible detection threshold. Furthermore, we elucidated the potential of combining methods.

It is notable that all methods achieved high accuracy for the difficult task of new species classification. Remarkably, PaPrBaG belonged to the few tools that could achieve solid predictions across a wide range of coverages. In contrast to most approaches, it yields reliable predictions for genomic coverages as low as 0.001. At high coverages, PaPrBaG performed competitive and in particular it performed better than composition based approaches.

For a reliable pathogen identification it is desirable to obtain relevant information from as many reads as possible. Whereas most methods could match only a small fraction of reads to pathogens or non-pathogens, PaPrBaG always makes a prediction. This proved to be key when requiring a certain minimum fraction of read evidence for prediction. In this evaluation PaPrBaG was found to be the most informative approach. The reliability of a prediction is also related to the prediction certainty. In this work, we could show that when selecting the most certain predictions, PaPrBaG achieved the best performance of all methods discussed.

Whereas the existing tools are based on taxonomic classification, PaPrBaG is a conceptually novel approach. It is a binary classifier that learns directly from a set of genome sequences of pathogenic and non-pathogenic species. Therefore, it is not surprising that the predictions of PaPrBaG are more diverse compared to the other approaches. In particular, PaPrBaG makes unique true and false predictions, which is beneficial when using a consensus approach. Combining the existing approaches with PaPrBaG led to particular high performance, better than any individual classifier.

It is furthermore interesting that although PaPrBaG was trained on sequencing reads that cover only a small fraction of the training genomes, the approach worked strikingly well. Hence, PaPrBaG is able to make solid predictions while it *sees* much less of the training data than the other methods.

In terms of run times, PaPrBaG runs much faster than BLAST and NBC though it is not optimised for speed like Bowtie2 and Kraken. Nevertheless, the pure prediction times are competitive with Bowtie2 and Kraken. The main bottleneck of PaPrBaG

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2379v1 | CC BY 4.0 Open Access | rec: 19 Aug 2016, publ: 19 Aug 2016

is the extraction of features from sequencing reads. However, this step could be further optimised, e.g. choosing genomic features only would result in a considerable speed-up. As our analysis has shown, considering a larger reference database leads to a strong increase of prediction times. An advantage of PaPrBaG is that the feature extraction times would remain constant. Additionally, larger training data size can be handled with pruning the trees in the random forest.

Apart from the applications of PaPrBaG in the classification of genomes, one can also envision its potential in a metagenomic context, e.g. for the analysis of clinical samples. In particular, its solid performance at very low coverages may give PaPrBaG a key role. A potential workflow would proceed as follows. Initially, fast tools designed for similar sequences such as Bowtie2 and Kraken can identify reads belonging to known references of bacterial, human or viral origin. In a next step, PaPrBaG can be used to scan the remaining set of unmapped reads for hints of pathogenicity. In particular, confident predictions are very likely to be true predictions as well. Thus, PaPrBaG can supply additional information about the sample that would otherwise not be accessible. Finally, further information about these very confident predictions could be gathered by a very sensitive protein BLAST or PSI-BLAST, which would be prohibitive on the entire read set. Hence, in such a metagenomic setting the role of PaPrBaG would be to prioritise the reads with highest associations to pathogens for further downstream analysis.

This work strongly depends on reliable phenotype information. We introduced a strategy within PaPrBaG to overcome the pathogen bias. Nevertheless, PaPrBaG as well as the other methods have higher sensitivity than specificity and it would be interesting to see how the methods would work with higher numbers of labeled bacterial species, in particular non-pathogens. Moreover, we presume that the newly created data set can stimulate further development of pathogenicity prediction workflows.

It is worth mentioning that the approach pursued by PaPrBaG is not restricted to the classification of the complex phenotype *pathogenicity*. It is rather a general workflow for the classification of labeled genomes, potential further applications range from bacterial host and habitat prediction, taxonomic classification to human and microbial read separation.

ACKNOWLEDGEMENTS

The authors thank Thilo Muth, Vitor Piro, and Mathias Kuhring for stimulating discussions (NG 4, Robert Koch Institute).

FUNDING

This work was supported by the German Federal Ministry of Health [IIA5-2512-FSB-725 to B.Y.R.].

REFERENCES

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990). Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410.

Barbosa, E., Röttger, R., Hauschild, A.-C., Azevedo, V., and Baumbach, J. (2014). On the limits of computational functional genomics for bacterial lifestyle prediction. *Briefings in Functional Genomics*, 13(5):398–408.

Benson, D. A., Cavanaugh, M., Clark, K., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., and Sayers, E. W. (2013). GenBank. *Nucleic Acids Research*, 41(Database issue):D36–42.

Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1):5–32.

Byrd, A. L., Perez-Rogers, J. F., Manimaran, S., Castro-Nallar, E., Toma, I., McCaffrey, T., Siegel, M., Benson, G., Crandall, K. A., and Johnson, W. E. (2014). Clinical PathoScope: rapid alignment and filtration for accurate pathogen identification in clinical samples using unassembled sequencing data. *BMC Bioinformatics*, 15:262.

Cosentino, S., Voldby Larsen, M., Møller Aarestrup, F., and Lund, O. (2013). PathogenFinder - Distinguishing Friend from Foe Using Bacterial Whole Genome Sequence Data. *PLoS ONE*, 8(10):e77302.

Creighton, T. E. (1993). *Proteins: Structures and Molecular Properties*. W. H. Freeman.

Diaz, N. N., Krause, L., Goesmann, A., Niehaus, K., and Nattkemper, T. W. (2009). TACOA - Taxonomic classification of environmental genomic fragments using a kernelized nearest neighbor approach. *BMC Bioinformatics*, 10(1):56.

Folleco, A., Khoshgoftaar, T., Van Hulse, J., and Bullard, L. (2008a). Identifying learners robust to low quality data. In *IEEE International Conference on Information Reuse and Integration, 2008. IRI 2008*, pages 190–195.

Folleco, A., Khoshgoftaar, T., Van Hulse, J., and Bullard, L. (2008b). Software quality modeling: The impact of class noise on the random forest classifier. In *IEEE Congress on Evolutionary Computation, 2008.*, pages 3853–3859.

Francis, O. E., Bendall, M., Manimaran, S., Hong, C., Clement, N. L., Castro-Nallar, E., Snell, Q., Schaalje, G. B., Clement, M. J., Crandall, K. A., and Johnson, W. E. (2013). Pathoscope: Species identification and strain attribution with unassembled sequencing data. *Genome Research*, page gr.150151.112.

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

Hatem, A., Bozdağ, D., Toland, A. E., and Çatalyürek, Ü. V. (2013). Benchmarking short sequence mapping tools. *BMC Bioinformatics*, 14:184.

Hogan, J. M., Holland, P., Holloway, A. P., Petit, R. A., and Read, T. D. (2013). Read classification for next generation sequencing. In *ESANN 2013 proceedings : European Symposium on Artificial Neural Networks, Computational Intelligence*, pages 485–490, Bruges, Belgium. The European Symposium on Artificial Neural Networks.

Holtgrewe, M. (2010). Mason - A Read Simulator for Second Generation Sequencing Data. *Technical Report FU Berlin*.

Hong, C., Manimaran, S., Shen, Y., Perez-Rogers, J. F., Byrd, A. L., Castro-Nallar, E., Crandall, K. A., and Johnson, W. E. (2014). PathoScope 2.0: a complete computational framework for strain identification in environmental or clinical sequencing samples. *Microbiome*, 2:33.

Human Microbiome Project Consortium (2012a). A framework for human microbiome research. *Nature*, 486(7402):215–221.

Human Microbiome Project Consortium (2012b). Structure, function and diversity of the healthy human microbiome. *Nature*, 486(7402):207–214.

Iraola, G., Vazquez, G., Spangenberg, L., and Naya, H. (2012). Reduced Set of Virulence Genes Allows High Accuracy Prediction of Bacterial Pathogenicity in Humans. *PLoS ONE*, 7(8):e42144.

Juhas, M. (2015). Horizontal gene transfer in human pathogens. *Critical Reviews in Microbiology*, 41(1):101–108.

Kanehisa, M., Goto, S., Sato, Y., Kawashima, M., Furumichi, M., and Tanabe, M. (2014). Data, information, knowledge and principle: back to metabolism in KEGG. *Nucleic Acids Research*, 42(Database issue):D199–205.

Kawashima, S. and Kanehisa, M. (2000). AAindex: Amino Acid index database. *Nucleic Acids Research*, 28(1):374.

Langmead, B. and Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. *Nature methods*, 9(4):357–359.

Leimeister, C.-A., Boden, M., Horwege, S., Lindner, S., and Morgenstern, B. (2014). Fast alignment-free sequence comparison using spaced-word frequencies. *Bioinformatics*, 30(14):1991–1999.

Li, H. and Durbin, R. (2009). Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14):1754–1760.

Lindner, M. S. and Renard, B. Y. (2015). Metagenomic profiling of known and unknown microbes with microbeGPS. *PloS One*, 10(2):e0117711.

Ma, B., Tromp, J., and Li, M. (2002). PatternHunter: faster and more sensitive homology search. *Bioinformatics*, 18(3):440–445.

Malley, J. D., Kruppa, J., Dasgupta, A., Malley, K. G., and Ziegler, A. (2012). Probability machines: consistent probability estimation using nonparametric learning machines. *Methods of Information in Medicine*, 51(1):74–81.

Mande, S. S., Mohammed, M. H., and Ghosh, T. S. (2012). Classification of metagenomic sequences: methods and challenges. *Briefings in Bioinformatics*, 13(6):669–681.

Marçais, G. and Kingsford, C. (2011). A fast, lock-free approach for efficient parallel counting of occurrences of k-mers. *Bioinformatics*, 27(6):764–770.

Markowitz, V. M., Chen, I.-M. A., Palaniappan, K., Chu, K., Szeto, E., Pillay, M., Ratner, A., Huang, J., Woyke, T., Huntemann, M., Anderson, I., Billis, K., Varghese, N., Mavromatis, K., Pati, A., Ivanova, N. N., and Kyrpides, N. C. (2014). IMG 4 version of the integrated microbial genomes comparative analysis system. *Nucleic Acids Research*, 42(D1):D560–D567.

McHardy, A. C., Martín, H. G., Tsirigos, A., Hugenholtz, P., and Rigoutsos, I. (2007). Accurate phylogenetic classification of variable-length DNA fragments. *Nature Methods*, 4(1):63–72.

Melsted, P. and Pritchard, J. K. (2011). Efficient counting of k-mers in DNA sequences using a bloom filter. *BMC Bioinformatics*, 12:333.

Merhej, V., Georgiades, K., and Raoult, D. (2013). Postgenomic analysis of bacterial pathogens repertoire reveals genome reduction rather than virulence factors. *Briefings in Functional Genomics*, 12(4):291–304.

Miller, R. R., Montoya, V., Gardy, J. L., Patrick, D. M., and Tang, P. (2013). Metagenomics for pathogen detection in public health. *Genome Medicine*, 5(9):81.

Naccache, S. N., Federman, S., Veeraraghavan, N., Zaharia, M., Lee, D., Samayoa, E., Bouquet, J., Greninger, A. L., Luk, K.-C., Enge, B., Wadford, D. A., Messenger, S. L., Genrich, G. L., Pellegrino, K., Grard, G., Leroy, E., Schneider, B. S., Fair, J. N., Martínez, M. A., Isa, P., Crump, J. A., DeRisi, J. L., Sittler, T., Hackett, J., Miller, S., and Chiu, C. Y. (2014). A cloud-compatible bioinformatics pipeline for ultrarapid pathogen identification from next-generation sequencing of clinical samples. *Genome Research*, 24(7):1180–1192.

Nakai, K., Kidera, A., and Kanehisa, M. (1988). Cluster analysis of amino acid indices for prediction of protein structure and function. *Protein Engineering*, 2(2):93–100.

Patil, K. R., Roune, L., and McHardy, A. C. (2012). The PhyloPythiaS Web Server for Taxonomic Assignment of Metagenome Sequences. *PLoS ONE*, 7(6).

Patthy, L. (1999). Genome evolution and the evolution of exon-shuffling - a review. *Gene*, 238(1):103–114.

Roach, D. J., Burton, J. N., Lee, C., Stackhouse, B., Butler-Wu, S. M., Cookson, B. T., Shendure, J., and Salipante, S. J. (2015). A Year of Infection in the Intensive Care Unit: Prospective Whole Genome Sequencing of Bacterial Clinical Isolates Reveals Cryptic Transmissions and Novel Microbiota. *PLoS Genet*, 11(7):e1005413.

Rosen, G., Garbarine, E., Caseiro, D., Polikar, R., and Sokhansanj, B. (2008). Metagenome Fragment Classification Using N-Mer Frequency Profiles. *Advances in Bioinformatics*, 2008:e205969.

Rosen, G. L., Reichenberger, E. R., and Rosenfeld, A. M. (2011). NBC: the Naive Bayes Classification tool webserver for taxonomic classification of metagenomic reads. *Bioinformatics (Oxford, England)*, 27(1):127–129.

Röttger, R., Kalaghatgi, P., Sun, P., Soares, S. d. C., Azevedo, V., Wittkop, T., and Baumbach, J. (2013). Density parameter estimation for finding clusters of homologous proteins-tracing actinobacterial pathogenicity lifestyles. *Bioinformatics*, 29(2):215–222.

Schlaberg, R., Simmon, K. E., and Fisher, M. A. (2012). A Systematic Approach for Discovering Novel, Clinically Relevant Bacteria. *Emerging Infectious Diseases*, 18(3):422–430.

Segata, N., Waldron, L., Ballarini, A., Narasimhan, V., Jousson, O., and Huttenhower, C. (2012). Metagenomic microbial community profiling using unique clade-specific marker genes. *Nature Methods*, 9(8):811–814.

Tomii, K. and Kanehisa, M. (1996). Analysis of amino acid indices and mutation matrices for sequence comparison and structure prediction of proteins. *Protein Engineering*, 9(1):27–36.

Vouga, M. and Greub, G. (2016). Emerging bacterial pathogens: the past and beyond. *Clinical Microbiology and Infection*, 22(1):12–21.

Wood, D. E. and Salzberg, S. L. (2014). Kraken: ultrafast metagenomic sequence classification using exact alignments. *Genome Biology*, 15(3):R46.

Wright, M. N. and Ziegler, A. (2015). ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R. *arXiv:1508.04409 [stat]*.

Youden, W. J. (1950). Index for rating diagnostic tests. *Cancer*, 3(1):32–35.

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.2379v1 | CC BY 4.0 Open Access | rec: 19 Aug 2016, publ: 19 Aug 2016