OperonSEQer: A set of machine-learning algorithms with threshold voting for detection of operon pairs using short-read RNA-sequencing data

Raga Krishnakumar[1*] and Anne M. Ruffing[2]

1. Systems Biology Department, Sandia National Laboratories, Livermore, CA 94551. USA
2. Biological Sciences and Engineering Department, Sandia National Laboratories, Albuquerque, NM 87185. USA

* Correspondence to rkrishn@sandia.gov

Abstract

Operon prediction in prokaryotes is critical not only for understanding the regulation of endogenous gene expression, but also for exogenous targeting of genes using newly developed tools such as CRISPR-based gene modulation. A number of methods have used transcriptomics data to predict operons, based on the premise that contiguous genes in an operon will be expressed at similar levels. While promising results have been observed using these methods, most of them do not address uncertainty caused by technical variability between experiments, which is especially relevant when the amount of data available is small. In addition, many existing methods do not provide the flexibility to determine whether the stringency with which genes should be evaluated for being in an operon pair. We present OperonSEQer, a set of machine learning algorithms that uses the statistic and p-value from a non-parametric analysis of variance test (Kruskal-Wallis) to determine the likelihood that two adjacent genes are expressed from the same RNA molecule. We implement a voting system to allow users to choose the stringency of operon calls depending on whether your priority is high coverage of operons or high accuracy of the calls. In addition, we provide the code so that users can retrain the algorithm and re-establish hyperparameters based on any data they choose, allowing for this method to be expanded on as additional data is generated and incorporated. We show that our approach detects operon pairs that are missed by current methods by comparing our predictions to publicly available long-read sequencing data. OperonSEQer therefore improves on existing methods in terms of accuracy, flexibility and adaptability.

Author Summary

Bacteria and archaea, single-cell organisms collectively known as prokaryotes, live in all imaginable environments and comprise the majority of living organisms on this planet. Prokaryotes play a critical role in the homeostasis of multicellular organisms (such as animals and plants) and ecosystems. In addition, bacteria can be pathogenic, and cause a variety of diseases in these same hosts and ecosystems. In short, understanding the biology and molecular functions of bacteria and archaea and devising mechanisms to engineer and optimize their properties are critical scientific endeavors with significant implications in healthcare, agriculture, manufacturing and climate science among others. One major molecular difference between unicellular and multicellular organisms is the way the express genes – rather than making individual RNA molecules like multicellular organisms, prokaryotes express genes in long contiguous RNA molecules known as operons, which are subsequently processed. Understanding which genes exist within operons is critical for elucidating basic biology and for engineering organisms. In this work, we use a combination of statistical and machine learning-based methods to use next-generation sequencing data to predict operon structure across a range of prokaryotes. Our method provides a easily implemented, robust, accurate and flexible way to determine operon structure in an organism-agnosic manner using readily-available data.

Introduction

Bacteria often transcribe functionally related genes not as single units but as contiguous RNA molecules (i.e., operons) - these molecules are under the control of a single promoter, allowing them to be co-expressed when required[1-6]. While there are a number of well-characterized operons and operon prediction methods in the literature, qPCR and more recently, deep sequencing technology, are revealing novel, previously uncharacterized operons in many bacterial species[7,8].

Existing operon predictions often show high precision and accuracy for well-annotated organisms, but many of them require information about gene function and conservation[9-12]. Newer methods include the use of visual representations of the genome to categorize operons[13].

A drawback of some of these approaches is the challenge in incorporating empirical real-world data regarding operon structure, which is constantly being generated and evolving our understanding and cataloging of operons. It is therefore imperative to couple methods based on existing genomic information with data-based predictions.

Recent work has shown that using RNA-sequencing (RNA-seq) data can significantly help increase the accuracy of operon prediction[14-19]. While this previous work is critical for the advancement of the understanding of operon biology as it demonstrates the usability of RNA-seq data in this context, there is still a gap in the technology with respect to software that is both broadly-applicable across experimental conditions and species, but also flexible in allowing the user to decide whether catching the highest number of operon pairs (high recall) or being very discerning (high precision) is most important. We believe that an approach that leverages not raw signal in RNA-seq data (which is highly variable and prone to batch effects), but rather uses statistics to determine the distribution of signal across two genes and an intergenic region provides a broader approach to operon prediction that can be used across a range of data sets and species. In addition, using multiple methods, and tallying the results gives the opportunity for a voting system that can give the user flexibility in what they decide to call a relevant operon pair. It is also increasingly clear that careful characterization of the resulting predictions against long-read-confirmed operons is necessary to truly evaluate the performance of a model, which is an technological opportunity that has recently arisen. And since novel data will continue to be generated, both using long- and short-read sequencing, it is necessary to provide the code to re-train and re-evaluate any method developed as this novel data emerges. To continue the work established by these studies and show that individual RNA-seq experiments can be sufficient for operon calls, we developed an operon prediction method, trained using a range of RNA-seq data from different organisms with a range of GC-content, to predict operon structure from a single set of RNA-seq data for two adjacent genes from data that has never been seen by the algorithm. Our approach addresses the issue of variability between RNA-seq data sets without requiring two or more matched experimental conditions, or any information about gene function, thereby building on and advancing the current state of the art in operon prediction. Our method also seeks to address the challenge of normalizing and featurizing the sequencing data to makes it generalizable across experiments without any prerequisites.

Operon-SEQer uses a non-parametric statistical test (chosen since the data is not necessarily normally distributed) to obtain the likelihood that the RNA-seq signal coverage across two genes and the intergenic region come from the same distribution. Our hypothesis is that the result of this statistical test, along with intergenic distance, is accurately predictive of an operon pair from any short-read RNA-seq data set, and we demonstrate this using a set of machine learning algorithms trained on existing data. We also show that using this method to identify operons in previously unseen organisms and data sets does not significantly reduce the accuracy, while leaving open the possibility to train the models with additional data sets if necessary. We evaluate six different algorithms and show that while specificity and recall vary for each algorithm, they all perform on-par with existing operon prediction methods; By taking advantage of a mutli-algorithm method that uses a threshold voting system, we further improve on this performance. In addition, we show that Operon-SEQer identifies new operon pairs that are not found in previous standard predictions but are likely to be true operons based on empirical evidence from previously published long-read *E.coli* RNA-seq data[7]. Finally, we demonstrate that while Operon-SEQer can call operons based on a single data point (without replicates) of a gene pair and the intergenic region, having 2 or more replicates per gene pair greatly increases its performance. In summary, our operon calling method matches the state of the art in operon prediction by determining operon status of gene pairs with high precision and recall, and advances the state of the art by identifying new operon pairs, and by providing flexibility to the user to determine whether they want their results to favor higher recall (i.e. catch every single operon pair) or higher specificity (i.e. make sure anything called is a true positive).

Results

Statistical analysis of features from RNA-seq data for operon prediction

The main aims of Operon-SEQer are to predict operon status from individual data sets to produce a comprehensive list of potential operons, for these predictions to be statistically robust despite only having single data sets, and to be species-agnostic. While we acknowledge that there are species-specific differences that may affect the outcome of such an algorithm (e.g., intergenic distances are of different lengths in different organisms), our premise was that each two-way comparison of adjacent genes on the same DNA strand, regardless of any other features, was an individual data point, and that a range of algorithms could be trained on a compilation of such data points across species, conditions, and replicates. This also allowed us to have many more data points than if we had taken a gene-specific approach. To this end, we established a statistical method that determines whether the RNA-seq coverage signal across the intergene-flanking regions of two adjacent genes on the same strand is from a single distribution. Using RNA-seq signal from the gene regions directly flanking the intergenic region, as well as the intergenic region itself, a non-parametric rank test (Kruskal-Wallis) was applied to obtain both a statistic and p-value for the comparison of the coverage signal at the three regions – Gene A, Gene B and the intergenic region (Figure 1). Previous reports have shown that intergenic distance is an important factor in determining whether two genes belong to the same operon, so we used the intergenic distance as well as the Kruskal-Wallis statistic and p-value as features for calling operon gene pairs[20,21].

A challenge in using RNA-seq data to model operons, especially when users do not have the computational resoureces with bandwidth to train algorithms on enormous amounts of data, is having enough diversity in the input data to cover a wide range of conditions that might be relevant to your organisms of interest. Therefore, OperonSEQer was trained on a wide range of organisms and was designed to allow for user input of additional organism and RNA-seq data for customization and iterative improvement. We used publicly deposited RNA-seq data sets from 7 different bacterial species (both Gram-positive and Gram-negative as well as heterotrophic and photoautotrophic): *Burkholderia pseudomallei* (*B. pseu*), *Clostridium difficile* (*C. diff*), *Escherichia coli* (*E. coli*), *Synechococcus* sp. PCC 7002 (*Syn.* 7002), *Synechocystis sp.* PCC 6803, *Synechococcus elongatus* PCC 7942 (*S. elon*), *Staphylococcus aureus* (*S. aure*) and *Bacillus subtilis* (*B. subt*)[22-37].

The data were processed and annotated as outlined in the Methods, using standard pipelines and publicly available software. In addition, we downloaded standard operon predictions by finding common operon calls between MicrobesOnline and ProOpDB where available[11,12]. Operon predictions from these online tools agreed to a high degree (83% agreement), and therefore, we chose the MicrobesOnline prediction as ground truth for operon structure, as this database had the largest number of organisms. We chose not to combine existing operon calls for *E. coli* since that would skew the accuracy of *E.coli* over other organisms and therefore the skew the trained models.

We performed a correlation analysis between pairs of genes in an operon (gene A and gene B with intermediate region I) and a number of important features from Kruskal-Wallis (KW) analysis of the RNA-seq data (Figure 2). The features used were: Kruskal-Wallis statistic and Kruskal-Wallis p-value (all 2-way comparisons plus the 3-way comparison) and intergenic distance. A large KW statistic represents a large difference in signal between the groups being compared, and a small p-value indicates that this difference is significant. Using the 2-way and 3-way comparisons, we get 8 dimensions of information, and while it is possible that each of these is uniquely impactful in defining an operon, we acknowledge that some of them may be related (eg. the 3-way comparison is likely to correlate with individual 2-way comparisons). Nevertheless, we include all these parameters in our analysis to maximize information use. We used a log10 transformation for the KW p-values to improve resolution. As expected, the length of genes A and B do not correlate with operon structure, and as previously reported[20,21,38,39], intergenic distance correlates negatively with likelihood of an operon pair (Figure 2). In terms of gene expression, the KW statistic correlates negatively with operon pair likelihood, and the log value of the KW p-value correlates positively (Figure 2). Despite RNA-seq data coming from different organisms and disparate sources, we find that the KW statistic and p-value have a higher correlation with operon pairs than intergenic distance, highlighting the importance of the information coming from RNA-seq across species. In addition, metrics that assay RNA-seq coverage of the intergenic region are the most predictive of operon pairs as expected. However, no single data point had a higher than 50% correlation, suggesting that inferring a direct linear relationship between any features and the outcome of being in an operon would be too simplistic, therefore requiring a more complex model.

Operon-SEQer improves recall and specificity for operon prediction

To improve operon prediction from RNA-seq data, we used intergenic length, KW statistics, and KW p-values as features for machine learning. We tested a range of classification algorithms that have previously been used in similar applications: logistic regression (LR), support vector machine (SVM, using the radial basis function which we determined to perform better than the linear, sigmoid or polynomial kernels), random forest (RF), XGBoost (XGB) and Gaussian Naïve Bayes (GNB). We used all of the data sets outlined in the methods and initially validated the various models using 50 random bootstraps of 75% of the data for training and 25% of the data for validation[40-43]. Recall and specificity served as measures of success to match previous reports[10,14]. As we are aiming for a species- and gene-agnostic method, these results are an aggregate of all the species and data sets that we included in our analysis.

While there was some trade-off between recall and specificity, all algorithms performed with both recall and specificity of at least 80% (Figure 2b). In particular, the tree-based methods (i.e. RF and XGB) had the best performance, with XGBoost having almost perfect recall and specificity in this validation set. We then conducted an independent test of our program to truly understand the broad applicability of our algorithms. We downloaded new RNA-seq data sets from *E.coli* and *B. subtilis*, organisms that were represented in the training data (but this new data is unseen by the algorithm), as well as RNA-seq data sets from *Mycobacterium tuberculosis* (*M. tuberculosis*) and *Pseudomonas syringiae* (*P. syringiae*), organisms (and data) absent from the training data[40-43]. We compared operon calls from our algorithms using these new, unseen data sets against operon annotations from MicrobesOnline. To get a confidence interval for our calls, we sub-sampled 10% of the data with replacement over 100 iterations for each algorithm. These results are plotted along with 95% confidence intervals in Figure 3. There was a range of performance depending on the algorithm used. The GNB and MLP algorithms, for the most part, had higher specificity compared with recall, which suggests that these methods are preferable for conservative operon calls. Ideally, however, we want to capture the largest number of operons. The logistic regression, SVM and tree-based methods (RF and XGB) have higher recall compared with specificity, which allows for a more complete annotation of operons but raises the concern of potential false-positive results. All results were confirmed by plotting receiver operating characteristics (ROC) curves (Sup. Figure 1). The higher recall and slightly lower specificity brings up the question of whether there may be some operons called by Operon-SEQer that are not annotated in MicrobesOnline, which is used as the standard. The question is whether these truly are false-positives or whether we are discovering new operon pairs that have not yet been annotated. To determine if there was a bias in recall and specificity related to the depth and coverage of the sequencing data, we analyzed the *M. tuberculosis* data since the various experiments had a large range of sequencing depth (Sup. Figure 2). We found no correlation of total reads, total mapped reads and percent mapped reads, with recall or specificity, suggesting that depth of sequencing is not limiting when using Operon-SEQer.

We compared the Operon-SEQer results for *E. coli* and *B. subtilis* with two state-of-the-art methods for operon detection, DOOR and Rockhopper, to ensure that the flexibility of our method did not affect the performance relative to other methods[10,14]. For Operon-SEQer, we calculated the recall and specificity for operon calls that were confirmed by 1 – 6 of the algorithms in our method. In other words, we set cutoffs ranging from 1 to 6 for how many algorithms had to call an operon pair before it was considered a true result (Sup Figure 3). We found that overall, Operon-SEQer performs on-par or better than the state-of-the art methods. The heat map in Sup Figure 3 shows that with just one of the six algorithms required for calling an operon pair, Operon-SEQer has perfect recall for both organisms. There is an expected trade-off between recall and specificity, however, with the compromise point somewhere between 2 and 4 algorithms, depending on the organism. This suggests that using 3 algorithms to call an operon pair is likely a good starting point.

Operon-SEQer enables prediction of new operons

Prior calculations of specificity assume that the operon structure provided by the standard, MicrobesOnline, is ground truth[12]. However, it is possible that the application of RNA-seq data enables prediction of new operons, previously missed by the standard. To address this issue of lower specificity versus novel operons, we sought to corroborate operon calls from Operon-SEQer using long-read PacBio SMRTseq transcriptomic data from *E. coli*[7]. In this prior study, a new set of previously unreported operons were discovered based on direct evidence of individual molecules of RNA spanning two genes. We separated the operon calls made by Operon-SEQer (using the different algorithms) in *E. coli* into four categories: (i) operon pairs called by neither SMRTseq nor the standard, (ii) operon pairs called by the standard only, (iii) operon pairs called by SMRTseq only, and (iv) operon pairs called by both. We then examined what proportion of the calls in these various groups were confirmed by Operon-SEQer. We used a threshold voting method by which cutoffs were designed based on how many Operon-SEQer algorithms identified an operon pair (1-6). When the SMRTseq data and standard agree, Operon-SEQer can identify a vast majority (>80%) of these operon pairs while requiring that 5/6 algorithms call the operon pair, suggesting a high level of three-way agreement between the methods (Figure 4a). When both SMRTseq and the standard do not find an operon pair, no more than 10% of those get called as an operon pair by Operon-SEQer, even when that is only by 1/6 algorithms. If we require a higher number of algorithms to call an operon pair, that percentage is in the single digits. Of note, when SMRTseq calls an operon pair not identified by the standard, at least one of our algorithms calls almost half of those operon pairs, suggesting that there are in fact operon pairs missed by the standard that can be predicted by Operon-SEQer (Figure 4a). We confirm this increase in specificity for each individual algorithm when looking at operon pairs with or without SMRTseq calls (Figure 4B). We note the lower recall (Figure 4b) and attribute this to lowly expressed gene pairs being called as operons in the SMRTseq experiment that our reliability cutoffs for short-read RNA-seq data likely miss.

Next, we looked at the specificity and recall of our method for operons that are called by the standard, by SMRTseq, or by either one. As expected, we see a trade-off between the specificity and the recall of all operon pairs as we increase the number of algorithms required to call an operon pair in *E. coli* (Figure 4c), and this tradeoff exists with data sets for other organisms as well (Sup. Figure 3). Since the SMRTseq data represents only one experimental condition, we do not expect that all operon pairs will be detected with this data set, which is why our method shows lower specificity with SMRTseq-called pairs than with standard-called pairs (Figure 4c). Again, the lower recall with SMRTseq data suggests that some operon pairs with very low expression are detected with long-read sequencing but are difficult to detect with short-read sequencing. The specificity of Operon-SEQer is higher (especially at lower algorithm number cutoffs) when we consider all operon pairs called by either SMRTseq or the standard (Figure 4c). This suggests that Operon-SEQer is likely detecting operon pairs that are missed by traditional operon callers, which rely on sequence and conservation information, and that these operon pairs can be identified using RNA-seq data. A similar result was demonstrated by the authors of Rockhopper, where they show that some of the operons Rockhopper detects that are not called by the standard can be confirmed by RT-qPCR[14]. Here, we show this on a global scale using long-read sequencing data, and we only require a single experimental condition to achieve this (as opposed to a comparison of two experimental conditions).

While Operon-SEQer allows for calls from a single experiment, and all our data until now is representative of operon pair calls based on a single RNA-seq result for each gene pair, we tested whether we could use the incidence of RNA-seq replicates (either biological replicates of a single condition or multiple experimental conditions) to strengthen our predictions. We therefore focused only on gene pairs that had data in at least 2 instances of data (i.e. crossed expression thresholds at least twice) and required agreement between the two replicates to make a final call. Replicate agreement was defined as the operon call made for each replicate being the same within an algorithm. We see that requiring two or more calls in agreement drastically improves the recall and specificity for all our comparisons (Figure 4d. Specifically, when we look at operon pairs that are called by either the standard or SMRTseq (solid line in Figure 4d), having even a single algorithm in our set of algorithms call the operon pair ensures a specificity of 96% and a recall of almost 90%, demonstrating that replicates significantly improved the performance of our program without requiring more training.

Discussion

The emergence of long-read sequencing data has shown us that the discovery of operons in prokaryotes is far from complete. In fact, there are many nuances to operon structure, including modular transcription terminators, that lead to combinations of operons that are difficult to predict based solely on sequence and conservation[7]. While long-read RNA-sequencing is an effective way to address this gap, the limitation with this approach is the need for a wide range of experimental conditions to ensure capture of all operon pairs, which can be time-consuming and costly. As an alternative, we have demonstrated here that the abundance of short-read RNA-sequencing data that has been accumulated of these past decades can be used to discover operon pairs. We show that by using an set of algorithms, we can call operon pairs using short-read sequencing data from a range of organisms with high recall and specificity. In addition, we demonstrate that it is likely that we are identifying non-annotated operon pairs using this method, based on confirmation by long-read sequencing data (ref).

Our approach uses a set of algorithms and a threshold voting system, as we found the results both more robust and more flexible compared to individual algorithms. While there are advantages and disadvantages to each approach, the threshold voting system can provide some level of confidence in the call and allows the user to decide whether recall or specificity is more important for their particular needs. An example of an ensemble operon caller is CONDOP, which also uses RNA-seq for determining operon gene pairs[18]. The main distinction with our method is that CONDOP requires annotated operons from the DOOR database and outputs a list of condition-specific operons using RNA-seq data based on this previous annotation, while Operon-SEQer does *de novo* operon detection using only RNA-seq data and intergenic distance as inputs[18]. We also improve on the methods used by rSeqTU by incorporating a statistical front-end to allow for more variability across organisms and data sets, and we also use a wide range of training data, as well as multiple ML models and a voting system[15]. We also provide the code required to re-train our models as data acquisition evolves and novel sequencing data types emerge, which given the statistical front-end transformation, should be broadly applicable. Other applications in genomics where ensemble methods have proven very useful include annotation of genomic islands, detection of genomic mutations, and gene expression-based phenotype prediction[44-47]. The development of these flexible methods is critical for weathering the natural and technical variation between organisms and data sets, which we can see even between the data sets that we chose to analyze in this study. In addition to flexibility, generalizability has long been an issue with operon calling, with training data often dictating the subset of organisms that can be tested using an algorithm. Our approach circumvents this by taking a gene-agnostic, function-agnostic approach, while simultaneously transforming the data into a statistic and p-value. This allowed Operon-SEQer to make calls on organisms and data sets that were unseen during testing with high recall and specificity. In addition, the algorithm can be trained with additional data sets as RNA-seq technology evolves, highlighting the longevity of such an approach.

Operon-SEQer has the potential to identify unannotated operon gene pairs that are confirmed by long-read RNA-seq data. This suggests that there are still a number of design rules for operon structure in bacteria that remain unknown, and Operon-SEQer can be used as a tool to discover these rules by marking novel operon pairs that are detected through RNA-sequencing but had not previously been identified. We can also ask which of these rules are organism-specific and which are general based on the results of our prediction. There has been a significant amount of work demonstrating that there are a number of dynamic and ever-evolving forces at play when it comes to operon structure, including RNA decay, overlapping transcription and previously uncharacterized functional relationships[2,3,5,48]. Using Operon-SEQer, we can survey the large amounts of RNA-seq data that are currently available through public repositories, and we can identify novel operons that can point to new or understudied functions of genes in any prokaryotic organism. Furthermore, since Operon-SEQer only requires a single experiment for operon calling, we can compare operon calls between conditions to see whether there are any changes in operon structure based on the state of the cells.

A future goal for Operon-SEQer is to incorporate long-read RNA-sequencing as the data becomes available. In fact, Operon-SEQer can be consolidated into a larger, modular algorithm that incorporates data from many information streams. It may also be interesting to adapt Operon-SEQer for transfer learning for this purpose, as it has been demonstrated that transfer learning can be useful in the generalizability of operon calling[13]. Importantly, our approach of using a statistical method to determine the similarity in expression of different regions of the genome in RNA-seq data, and then using the outputs of this method for machine learning can be applied broadly not only to prokaryotes, but also in understanding regulation of gene expression in higher organisms. Such an endeavor would complement the plethora of work that is currently ongoing in the field of machine learning for understanding gene regulation[49-54]. Ultimately, the key to fully unlocking the potential of machine learning in understanding gene regulation is likely to arise from a combination of computational approaches, with carefully curated and processed data, and methods such as Operon-SEQer can be used, adapted, and expanded upon to achieve this goal.

Materials and Methods

Data sets

For training Operon-SEQer, publicly available RNA-seq data were downloaded from Sequence Read Archive (SRA) for *Escherichia coli* (PRJNA274573, PRJNA436580 and PRJNA473128), *Bacillus subtilis* (PRJNA511580 and PRJNA555096), *Clostridium difficile* (PRJNA244679, PRJNA283975, PRJNA338449 and PRJNA217778), *Burkholderia pseudomallei* (PRJNA413621 and PRJNA312225), *Staphylococcus aureus* (PRJNA514046, PRJNA541911 and PRJNA546264), *Synechococcus elongatus PCC 7942* (PRJNA315938), *Synechocystis sp. PCC 6803* (PRJNA361291) and *Synechococcus sp. PCC 7002* (PRJNA310120, PRJNA361291 and PRJNA212552).

For testing Operon-SEQer, publicly available RNA-seq data were downloaded from SRA for *Escherichia coli* (PRJNA274573, PRJNA436580 and PRJNA473128), *Bacillus subtilis*

(PRJNA511580 and PRJNA555096), *Clostridium difficile* (PRJNA244679, PRJNA283975, PRJNA338449 and PRJNA217778), *Burkholderia pseudomallei* (PRJNA413621 and PRJNA312225)

Preparing, aligning, quantifying and annotating RNA-seq data

RNA-seq data was aligned with Hisat2, and bedtools genomecov was used to extract coverage across the genome[55,56]. A gff3 file corresponding to each organism being surveyed was downloaded from Ensembl Bacteria (https://bacteria.ensembl.org/) and filtered for genes only. Importantly, we next filtered the data for where the mean coverage across at least one gene from the pair of genes being compared is 10 reads, thereby eliminating gene pairs that are not expressed or where no conclusion can be reached. This is an important step in training the algorithm so that it recognizes true negatives and positives and is not side-tracked by regions that are not expressed and therefore cannot be used as predictors.

Following this, we collected pairwise coverage data for adjacent genes, as well as the intergenic region between these genes. With the 5' most gene referred to as gene A and the 3' most gene referred to as gene B, we extract coverage from the 3' 50 bp of gene A (or the whole gene if it is shorter than 50 bp), the central 50 bp of the intergenic region (or the whole intergenic region if it is shorter than 50bp), and the 5' 50bp of gene B (or the whole gene if it is shorter than 50 bp). We performed a Kruskal-Wallis test on pairwise comparisons of coverage or a three-way comparison, and recorded the statistic and p-value associated with each test. These, along with the intergenic distance were used as input features for machine learning. Operon calls referred to as 'the standard' were downloaded from MicrobesOnline (www.microbesonline.org/). Long-read SMRT-seq Pacbio data was obtained from doi.org/10.1038/s41467-018-05997-6[7].

Operon-SEQer

Operon-SEQer is a set of models with a threshold voting system, and our code is publicly available at https://github.com/sandialabs/OperonSEQer. Briefly, we use the scikit-learn module of Python3 to implement the machine learning algorithms. Algorithms that were used include Logistic Regression with L2 ridge regularization (LR), Support Vector Machine with a RBF kernel

(SVM), Random Forest (RF), XGBoost (XGB), Multi-Layer Perceptron (MLP) and Gaussian Naïve Bayes (GNB). Features were scaled for all algorithms except RF and XGB.

The downloaded data was processed as outlined above, and the following features were used for machine learning: length of gene A, length of gene B, intergenic length, Kruskal-Wallis statistics and p-values for pairwise and three-way comparison of gene A, gene B and intergenic coverage (as outlined above), and strand match between gene A and B. The data were scaled (for all relevant algorithms) using MinMaxScalar. Each algorithm's hyperparameters were optimized using Bayesian Optimization (using Gaussian Processes) from GPyOpt methods. The hyperparameters for each algorithm are as follows:

Algorithm	**Categorical features**	**Continuous features**
Logistic regression	Lasso vs ridge regularization	-
Random Forest	-	Minimum sample split, maximum depth, number of estimators (all integer)
Support Vector Machine	Kernel	C (as applicable), gamma (as applicable)
XGBoost	-	Gamma, learning rate, number of estimators (integer)
Gaussian Naïve Bayes	-	Variance smoothing
Multilayer Perceptron	-	Alpha, Maximum iterations (integer), number of hidden layers (integer), number of neurons per layer (integer)

For the MLP, we used adam as the solver and relu as the activation function. We used only 10 iterations of optimization for all the methods (which we judged as sufficient given high accuracy during optimization) but we provide the code, which can be modified and used to reoptimize hyperparameters in parallel. For each iteration of the optimizer, the model with the current set of hyperparameters was cross-validated 10-fold and the average accuracy of these 10 iterations was used as the metric to evaluate performance. Final validation recall and specificity shown in Table 1.

The model was then saved with the optimized hyperparameters, and new, unseen data from four organisms (two from which we had used alternative data for training, and two from which we had used no data) were used for testing the algorithms. Individual precision and recall values were recorded across each run, with the comparison being made to the 'standard' operons called by MicrobesOnline[12]. Results were reported as an average of 100 runs, with 95% confidence intervals. ROC curves and AUC (area under the curve) were calculated using scikit-learn. Calls for n (1-6) number of algorithms were made by tallying the number of times a gene pair got called.

Additional details for Operon-SEQer are available at https://github.com/sandialabs/OperonSEQer.

ROC (receiving operating characteristic) curve analysis

The prediction probability for each Operon-SEQer algorithm was calculated in python using with predict_proba function in scikit-learn. False positive and true positive rates were determined using the roc_curve function across a range of probabilities from 0 to 1. AUC (area under the curve) score was determined using the roc_auc_score, with areas closer to 1 being closer to the ideal.

Acknowledgements

We would like to thank Joshua Podlevsky and Chuck Smallwood for discussions and advice regarding this work, Drew Levin, Bernard Nguyen and Steven Verzi for critical review of the manuscript, and Cameron Kunstadt for testing and troubleshooting of the software package. This work was supported by the Laboratory Directed Research and Development program at Sandia National Laboratories, a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Competing Interests

The authors do not have any competing interests to report.

Code availability

OperonSEQer is available at https://github.com/sandialabs/OperonSEQer

Figures

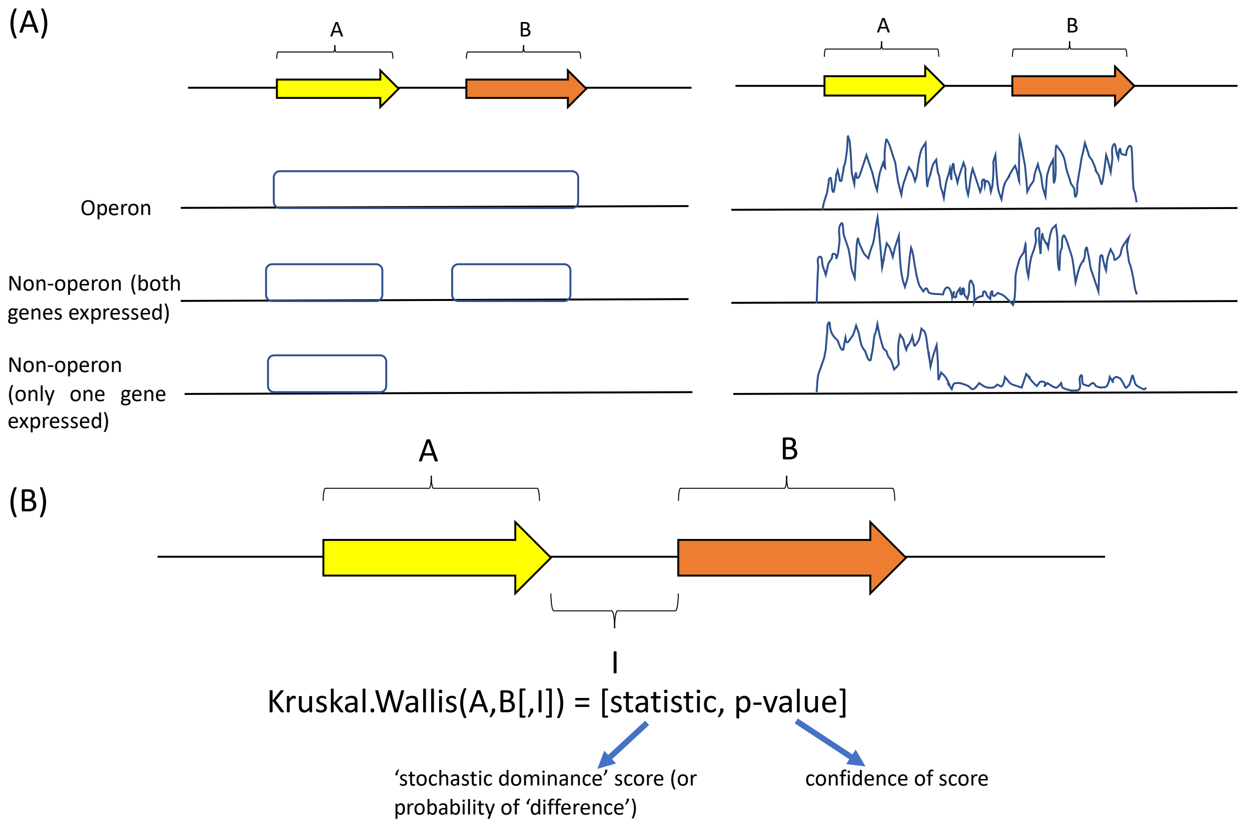

(A)
A
B
A
B
Operon
Non-operon (both genes expressed)
Non-operon (only one gene expressed)
(B)
A
B
I
Kruskal.Wallis(A,B[,I]) = [statistic, p-value]
'stochastic dominance' score (or probability of 'difference')
confidence of score

Figure 1 – Schematic of our method for determining similarity of RNA-seq signal between two adjacent genes. (A) Identification of an operon pair requires at least one of the two genes to be detectably expressed, and significant signal in the intergenic space. Idealized data on the left, and hypothetical real-world data on the right. (B) Usage of the Kruskal-Wallis statistic and p-value for pairwise comparisons of genes A, B and the intergenic (I) region, as well as the 3-way comparison. These values, along with the intergenic distance, serve as features for training our operon prediction model.

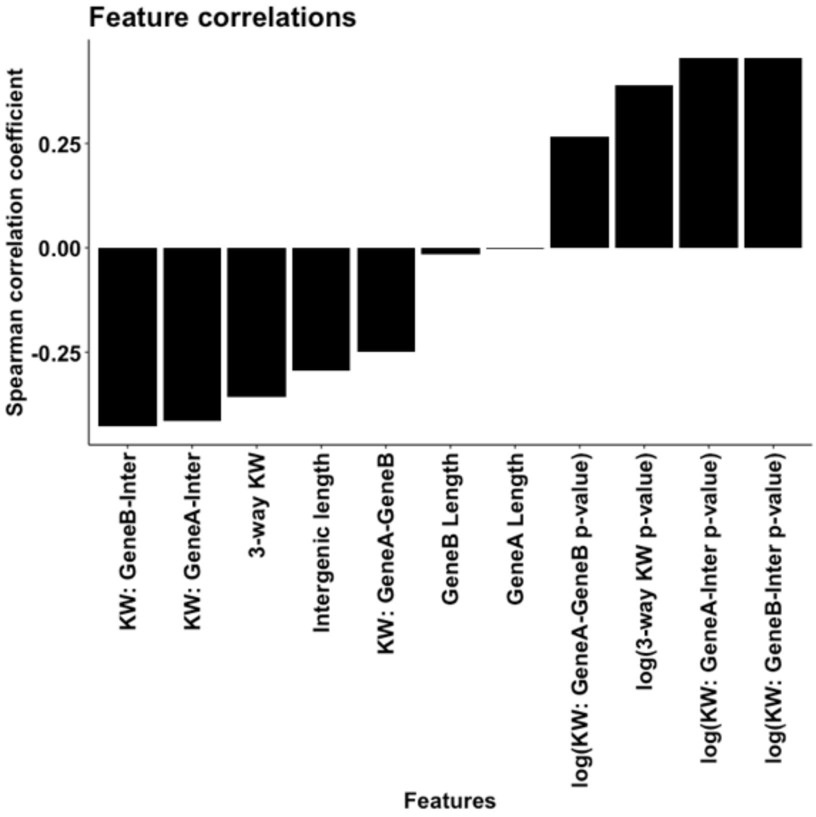


Figure 2 – Operon-SEQer features and performance across the various algorithms used. (A) Spearman's correlation coefficients between the features considered for use in machine learning and operon pair calls made by MicrobesOnline across 7-species (see main text). KW = Kruskal Wallis statistic.

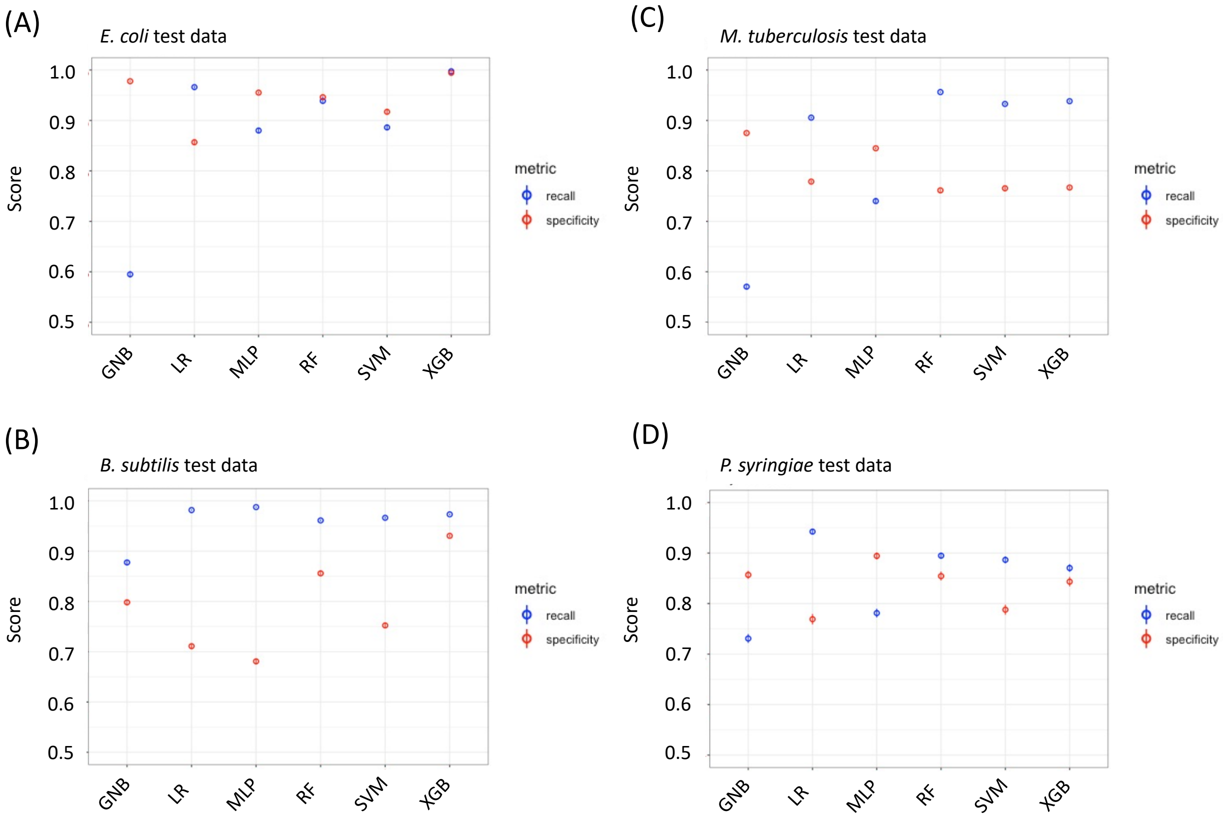


Figure 3 – Operon-SEQer can identify operon pairs in new, unseen data. Recall (blue) and specificity (red) for new data sets from (A) *E. coli*, (B) *B. subtilis*, (C) *M. tuberculosis,* and (D) *P. syringiae*. Mean numbers for 100 bootstrapped iterations are shown with 95% confidence intervals (central line in circle).

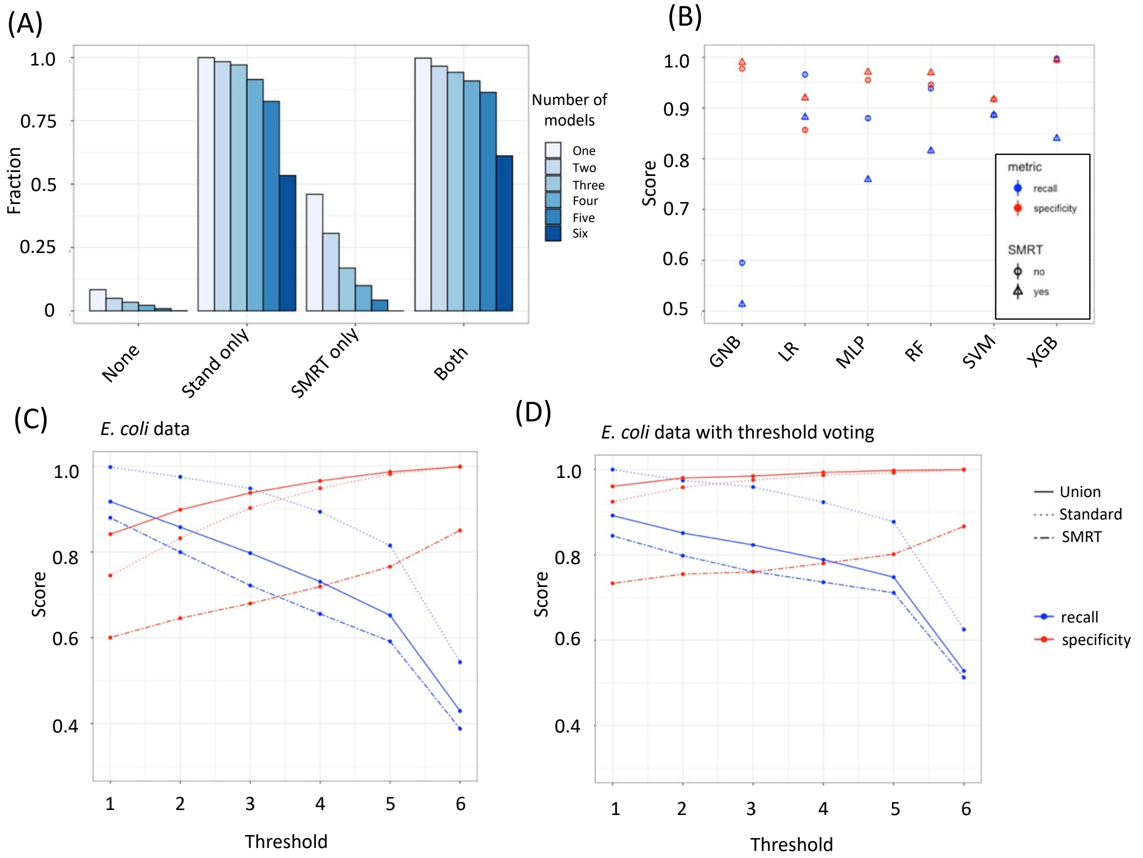
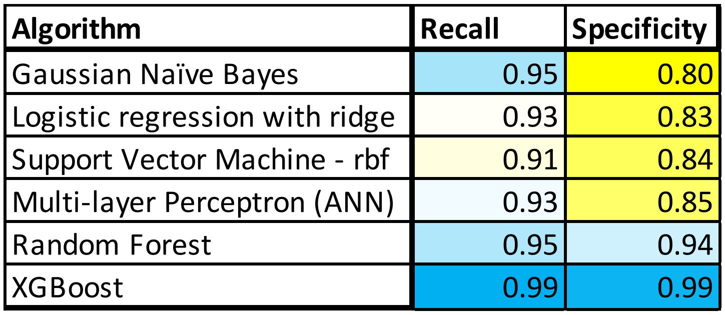


Figure 4 – Operon-SEQer is best used as an ensemble of methods and finds operons not annotated by the standard but detected by PACBIO SMRTseq. (A) Fraction of Operon-SEQer operon pair calls that are confirmed by SMRTseq, the standard, neither, or both. Cutoffs for Operon-SEQer operon calls are set at agreement of 1 – 6 algorithms within the ensemble. (B) Recall (blue) and specificity (red) of individual algorithms within Operon-SEQer for operon calls made only by the standard (circle) versus the standard plus SMRTseq calls (triangle). 95% confidence intervals of 100-fold bootstraps are shown as lines within the shape. (C and D) Recall (blue) and specificity (red) of the Operon-SEQer ensemble with algorithm agreement cutoffs of 1-6 for operon pair calls made by the standard (dotted lines), SMRTseq (dashed line), or by the union of calls made by both (solid line); (C) represents all available operon pair data for the new *E. coli* data sets and (D) represents operon pairs that have agreement between two or more replicates.



Algorithm	Recall	Specificity
Gaussian Naïve Bayes	0.95	0.80
Logistic regression with ridge	0.93	0.83
Support Vector Machine - rbf	0.91	0.84
Multi-layer Perceptron (ANN)	0.93	0.85
Random Forest	0.95	0.94
XGBoost	0.99	0.99

Table 1- Recall and specificity for the validation set for Operon-SEQer across six different algorithms. Heat map colors range from yellow (lowest) to white (mid-point) to blue (highest).

References

1 Bervoets, I. & Charlier, D. Diversity, versatility and complexity of bacterial gene regulation mechanisms: opportunities and drawbacks for applications in synthetic biology. *FEMS Microbiol Rev* **43**, 304-339, doi:10.1093/femsre/fuz001 (2019).

2 Bundalovic-Torma, C., Whitfield, G. B., Marmont, L. S., Howell, P. L. & Parkinson, J. A systematic pipeline for classifying bacterial operons reveals the evolutionary landscape of biofilm machineries. *PLoS Comput Biol* **16**, e1007721, doi:10.1371/journal.pcbi.1007721 (2020).

3 Dar, D. & Sorek, R. Extensive reshaping of bacterial operons by programmed mRNA decay. *PLoS Genet* **14**, e1007354, doi:10.1371/journal.pgen.1007354 (2018).

4 Osbourn, A. E. & Field, B. Operons. *Cell Mol Life Sci* **66**, 3755-3775, doi:10.1007/s00018-009-0114-3 (2009).

5 Saenz-Lahoya, S. *et al.* Noncontiguous operon is a genetic organization for coordinating bacterial gene expression. *Proc Natl Acad Sci U S A* **116**, 1733-1738, doi:10.1073/pnas.1812746116 (2019).

6 Jacob, F., Perrin, D., Sanchez, C. & Monod, J. [Operon: a group of genes with the expression coordinated by an operator]. *C R Hebd Seances Acad Sci* **250**, 1727-1729 (1960).

7 Yan, B., Boitano, M., Clark, T. A. & Ettwiller, L. SMRT-Cappable-seq reveals complex operon variants in bacteria. *Nat Commun* **9**, 3676, doi:10.1038/s41467-018-05997-6 (2018).

8 Conway, T. *et al.* Unprecedented high-resolution view of bacterial operon architecture revealed by RNA sequencing. *mBio* **5**, e01442-01414, doi:10.1128/mBio.01442-14 (2014).

9 Taboada, B., Estrada, K., Ciria, R. & Merino, E. Operon-mapper: a web server for precise operon identification in bacterial and archaeal genomes. *Bioinformatics* **34**, 4118-4120, doi:10.1093/bioinformatics/bty496 (2018).

10 Mao, X. *et al.* DOOR 2.0: presenting operons and their functions through dynamic and integrated views. *Nucleic Acids Res* **42**, D654-659, doi:10.1093/nar/gkt1048 (2014).

11 Taboada, B., Ciria, R., Martinez-Guerrero, C. E. & Merino, E. ProOpDB: Prokaryotic Operon DataBase. *Nucleic Acids Res* **40**, D627-631, doi:10.1093/nar/gkr1020 (2012).

12 Dehal, P. S. *et al.* MicrobesOnline: an integrated portal for comparative and functional genomics. *Nucleic Acids Res* **38**, D396-400, doi:10.1093/nar/gkp919 (2010).

13 Assaf, R., Xia, F. & Stevens, R. Detecting operons in bacterial genomes via visual representation learning. *Sci Rep* **11**, 2124, doi:10.1038/s41598-021-81169-9 (2021).

14 Tjaden, B. A computational system for identifying operons based on RNA-seq data. *Methods* **176**, 62-70, doi:10.1016/j.ymeth.2019.03.026 (2020).

15 Niu, S. Y., Liu, B., Ma, Q. & Chou, W. C. rSeqTU-A Machine-Learning Based R Package for Prediction of Bacterial Transcription Units. *Front Genet* **10**, 374, doi:10.3389/fgene.2019.00374 (2019).

16 Fortino, V., Smolander, O. P., Auvinen, P., Tagliaferri, R. & Greco, D. Transcriptome dynamics-based operon prediction in prokaryotes. *BMC Bioinformatics* **15**, 145, doi:10.1186/1471-2105-15-145 (2014).

17 Price, M. N., Huang, K. H., Alm, E. J. & Arkin, A. P. A novel method for accurate operon predictions in all sequenced prokaryotes. *Nucleic Acids Res* **33**, 880-892, doi:10.1093/nar/gki232 (2005).

18 Fortino, V., Tagliaferri, R. & Greco, D. CONDOP: an R package for CONdition-Dependent Operon Predictions. *Bioinformatics* **32**, 3199-3200, doi:10.1093/bioinformatics/btw330 (2016).

19 Zaidi, S. S. A. & Zhang, X. Computational operon prediction in whole-genomes and metagenomes. *Brief Funct Genomics* **16**, 181-193, doi:10.1093/bfgp/elw034 (2017).

20 Salgado, H., Moreno-Hagelsieb, G., Smith, T. F. & Collado-Vides, J. Operons in Escherichia coli: genomic analyses and predictions. *Proc Natl Acad Sci U S A* **97**, 6652-6657, doi:10.1073/pnas.110147297 (2000).

21 Okuda, S. *et al.* Characterization of relationships between transcriptional units and operon structures in Bacillus subtilis and Escherichia coli. *BMC Genomics* **8**, 48, doi:10.1186/1471-2164-8-48 (2007).

22 Lazar Adler, N. R. *et al.* Perturbation of the two-component signal transduction system, BprRS, results in attenuated virulence and motility defects in Burkholderia pseudomallei. *BMC Genomics* **17**, 331, doi:10.1186/s12864-016-2668-4 (2016).

23 Camara-Almiron, J. *et al.* Dual functionality of the amyloid protein TasA in Bacillus physiology and fitness on the phylloplane. *Nat Commun* **11**, 1859, doi:10.1038/s41467-020-15758-z (2020).

24 Kim, D. *et al.* Systems assessment of transcriptional regulation on central carbon metabolism by Cra and CRP. *Nucleic Acids Res* **46**, 2901-2917, doi:10.1093/nar/gky069 (2018).

25 Payne, S. R. *et al.* Inhibition of Bacterial Gene Transcription with an RpoN-Based Stapled Peptide. *Cell Chem Biol* **25**, 1059-1066 e1054, doi:10.1016/j.chembiol.2018.05.007 (2018).

26 Guyet, A. *et al.* Mild hydrostatic pressure triggers oxidative responses in Escherichia coli. *PLoS One* **13**, e0200660, doi:10.1371/journal.pone.0200660 (2018).

27 Burton, A. T., DeLoughery, A., Li, G. W. & Kearns, D. B. Transcriptional Regulation and Mechanism of SigN (ZpdN), a pBS32-Encoded Sigma Factor in Bacillus subtilis. *mBio* **10**, doi:10.1128/mBio.01899-19 (2019).

28 Sekulovic, O. & Fortier, L. C. Global transcriptional response of Clostridium difficile carrying the CD38 prophage. *Appl Environ Microbiol* **81**, 1364-1374, doi:10.1128/AEM.03656-14 (2015).

29 Maldarelli, G. A. *et al.* Type IV pili promote early biofilm formation by Clostridium difficile. *Pathog Dis* **74**, doi:10.1093/femspd/ftw061 (2016).

30 Girinathan, B. P. *et al.* Effect of tcdR Mutation on Sporulation in the Epidemic Clostridium difficile Strain R20291. *mSphere* **2**, doi:10.1128/mSphere.00383-16 (2017).

31 Scaria, J. *et al.* Differential stress transcriptome landscape of historic and recently emerged hypervirulent strains of Clostridium difficile strains determined using RNA-seq. *PLoS One* **8**, e78489, doi:10.1371/journal.pone.0078489 (2013).

32 Goncheva, M. I. *et al.* Stress-induced inactivation of the Staphylococcus aureus purine biosynthesis repressor leads to hypervirulence. *Nat Commun* **10**, 775, doi:10.1038/s41467-019-08724-x (2019).

33 Crosby, H. A. *et al.* The Staphylococcus aureus ArlRS two-component system regulates virulence factor expression through MgrA. *Mol Microbiol* **113**, 103-122, doi:10.1111/mmi.14404 (2020).

34 Sause, W. E. *et al.* The purine biosynthesis regulator PurR moonlights as a virulence regulator in Staphylococcus aureus. *Proc Natl Acad Sci U S A* **116**, 13563-13572, doi:10.1073/pnas.1904280116 (2019).

35 Choi, S. Y. *et al.* Transcriptome landscape of Synechococcus elongatus PCC 7942 for nitrogen starvation responses using RNA-seq. *Sci Rep* **6**, 30584, doi:10.1038/srep30584 (2016).

36 Lacey, R. F., Allen, C. J., Bakshi, A. & Binder, B. M. Ethylene causes transcriptomic changes in Synechocystis during phototaxis. *Plant Direct* **2**, e00048, doi:10.1002/pld3.48 (2018).

37 Begemann, M. B. *et al.* An organic acid based counter selection system for cyanobacteria. *PLoS One* **8**, e76594, doi:10.1371/journal.pone.0076594 (2013).

38 Dam, P., Olman, V., Harris, K., Su, Z. & Xu, Y. Operon prediction using both genome-specific and general genomic information. *Nucleic Acids Res* **35**, 288-298, doi:10.1093/nar/gkl1018 (2007).

39 Edwards, M. T., Rison, S. C., Stoker, N. G. & Wernisch, L. A universally applicable method of operon map prediction on minimally annotated genomes using conserved genomic context. *Nucleic Acids Res* **33**, 3253-3262, doi:10.1093/nar/gki634 (2005).

40 Krogh, T. J., Franke, A., Moller-Jensen, J. & Kaleta, C. Elucidating the Influence of Chromosomal Architecture on Transcriptional Regulation in Prokaryotes - Observing Strong Local Effects of Nucleoid Structure on Gene Regulation. *Front Microbiol* **11**, 2002, doi:10.3389/fmicb.2020.02002 (2020).

41 Plocinski, P. *et al.* Proteomic and transcriptomic experiments reveal an essential role of RNA degradosome complexes in shaping the transcriptome of Mycobacterium tuberculosis. *Nucleic Acids Res* **47**, 5892-5905, doi:10.1093/nar/gkz251 (2019).

42 Nobori, T. *et al.* Transcriptome landscape of a bacterial pathogen under plant immunity. *Proc Natl Acad Sci U S A* **115**, E3055-E3064, doi:10.1073/pnas.1800529115 (2018).

43 Morrison, M. D., Fajardo-Cavazos, P. & Nicholson, W. L. Comparison of Bacillus subtilis transcriptome profiles from two separate missions to the International Space Station. *NPJ Microgravity* **5**, 1, doi:10.1038/s41526-018-0061-0 (2019).

44 Li, Y. L., Y. Performance-weighted-voting model: an ensemble machine learning method for cancer type classification using whole-exome sequencing mutation. *Quantitative Biology* **8**, 347-358, doi:https://doi.org/10.1007/s40484-020-0226-1 (2020).

45 Jubair, S. D., M. in *IEEE International Conference on Bioinformatics and Biomedicine (BIBM)* (2019).

46 Wang, C. W. in *Proceedings of the 28th IEEE - EMBS Annual International Conference* (New York, NY, USA, 2006).

47 Abdollahi-Arpanahi, R., Gianola, D. & Penagaricano, F. Deep learning versus parametric and ensemble methods for genomic prediction of complex phenotypes. *Genet Sel Evol* **52**, 12, doi:10.1186/s12711-020-00531-z (2020).

48 Tavormina, P. L., Orphan, V. J., Kalyuzhnaya, M. G., Jetten, M. S. & Klotz, M. G. A novel family of functional operons encoding methane/ammonia monooxygenase-related proteins in gammaproteobacterial methanotrophs. *Environ Microbiol Rep* **3**, 91-100, doi:10.1111/j.1758-2229.2010.00192.x (2011).

49 Song, Q. *et al.* Prediction of condition-specific regulatory genes using machine learning. *Nucleic Acids Res* **48**, e62, doi:10.1093/nar/gkaa264 (2020).

50 Agarwal, V. & Shendure, J. Predicting mRNA Abundance Directly from Genomic Sequence Using Deep Convolutional Neural Networks. *Cell Rep* **31**, 107663, doi:10.1016/j.celrep.2020.107663 (2020).

51 Yang, Y., Fang, Q. & Shen, H. B. Predicting gene regulatory interactions based on spatial gene expression data and deep learning. *PLoS Comput Biol* **15**, e1007324, doi:10.1371/journal.pcbi.1007324 (2019).

52 Piles, M. *et al.* Machine learning applied to transcriptomic data to identify genes associated with feed efficiency in pigs. *Genet Sel Evol* **51**, 10, doi:10.1186/s12711-019-0453-y (2019).

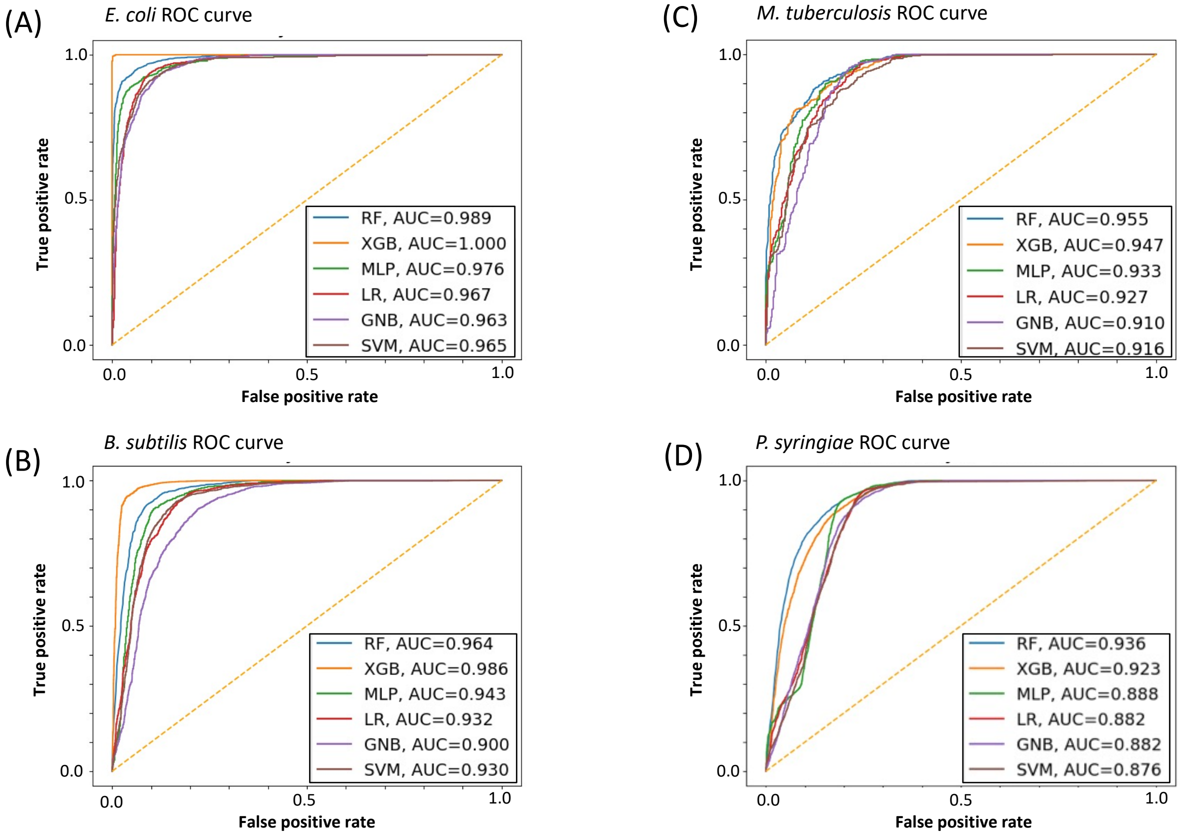
53 Yuan, Y. & Bar-Joseph, Z. Deep learning for inferring gene relationships from single-cell expression data. *Proc Natl Acad Sci U S A*, doi:10.1073/pnas.1911536116 (2019).

54 Wang, Y. *et al.* Using Machine Learning to Measure Relatedness Between Genes: A Multi-Features Model. *Sci Rep* **9**, 4192, doi:10.1038/s41598-019-40780-7 (2019).

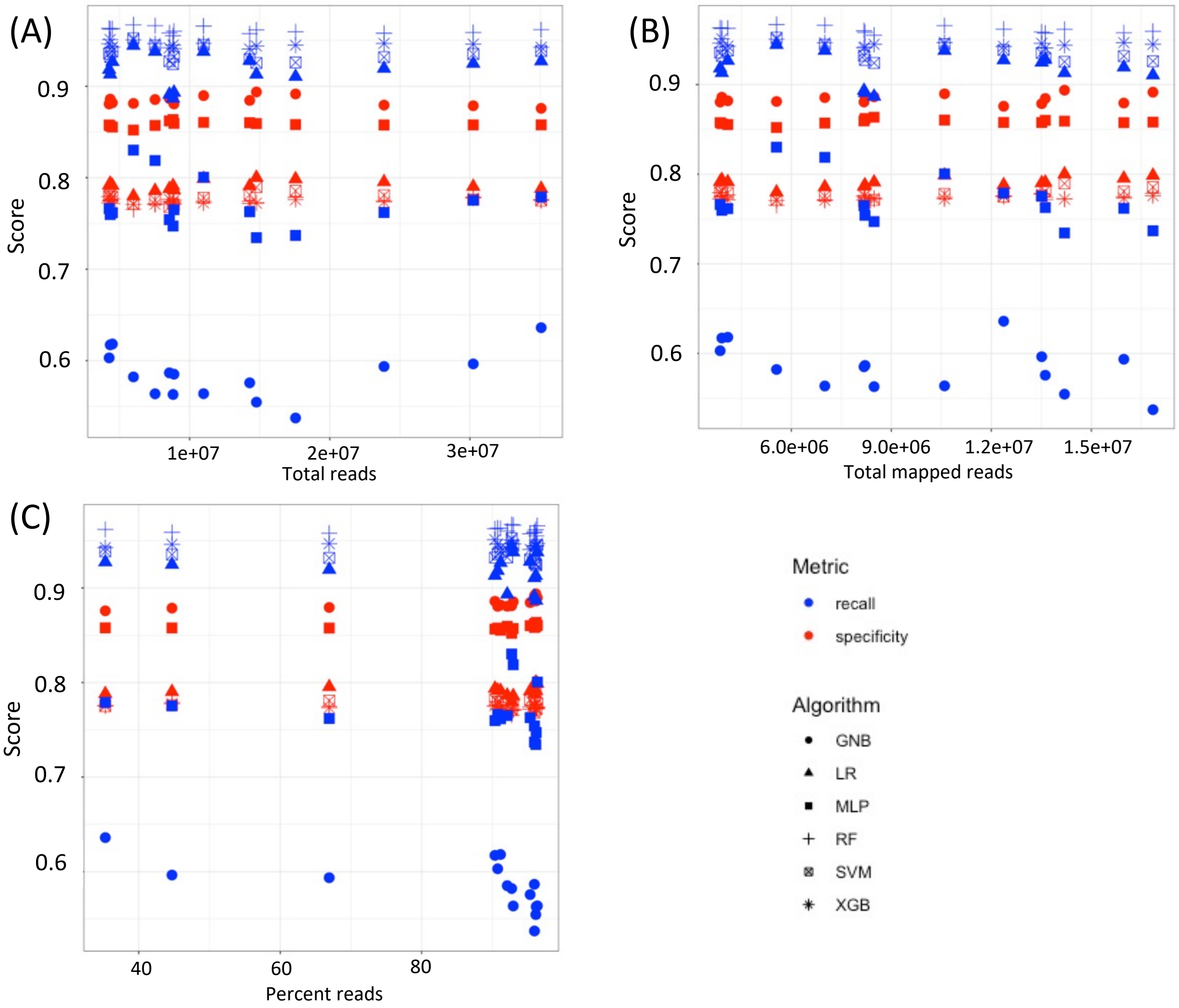
55 Kim, D., Paggi, J. M., Park, C., Bennett, C. & Salzberg, S. L. Graph-based genome alignment and genotyping with HISAT2 and HISAT-genotype. *Nat Biotechnol* **37**, 907-915, doi:10.1038/s41587-019-0201-4 (2019).

56 Quinlan, A. R. & Hall, I. M. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics* **26**, 841-842, doi:10.1093/bioinformatics/btq033 (2010).

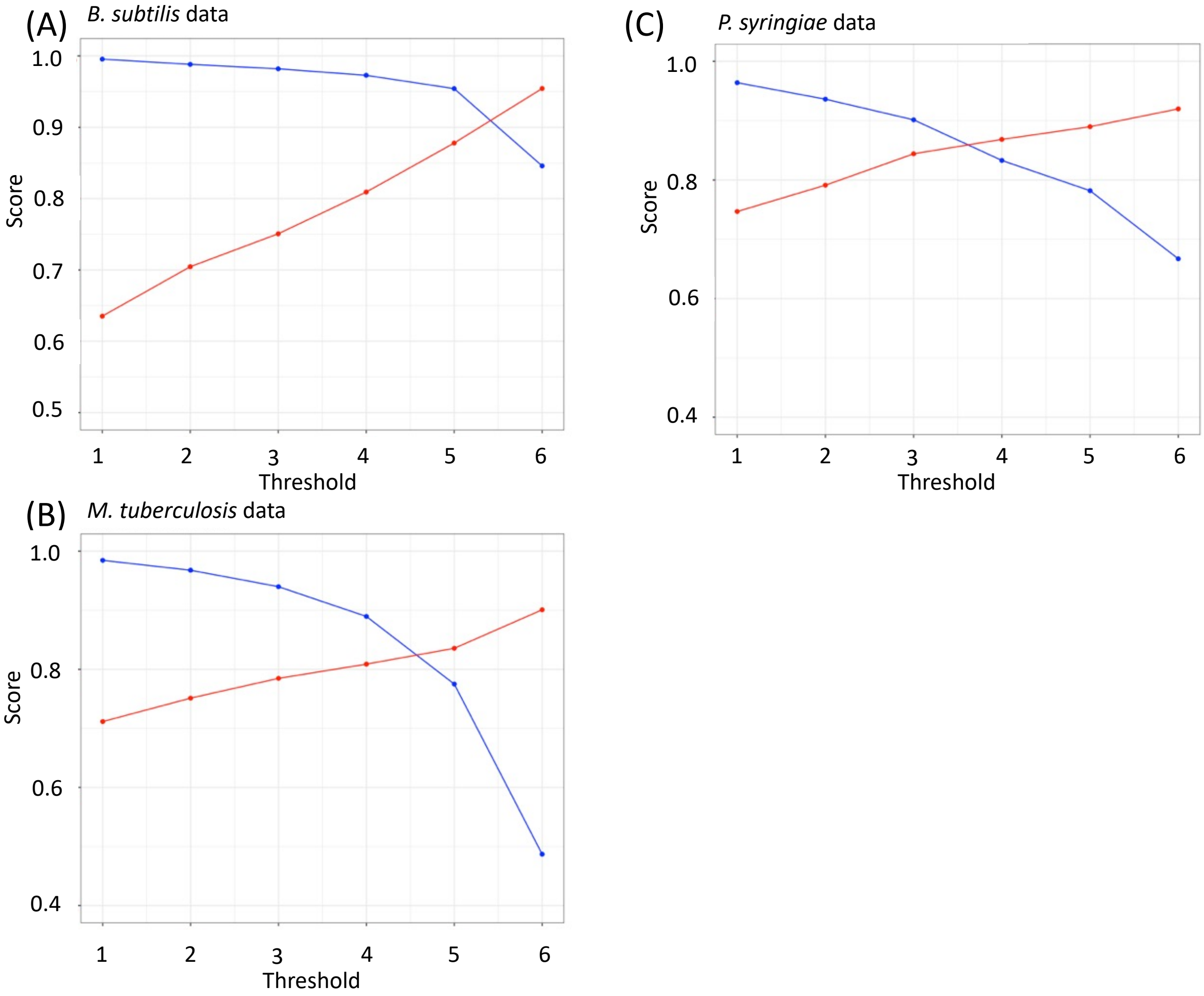
Supporting Information



Sup Figure 1 – ROC curves for Operon-SEQer performance. ROC (receiver operating characteristics) curves, and AUC (area under the curve) for the 7 algorithms in Operon-SEQer for the (A) *E. coli,* (B) *B. subtilis*, (C) *M. tuberculosis,* and (D) *P. syringiae* data sets.



Sup Figure 2 – Number of reads in a data set does not correlate with outcome of Operon-SEQer. Relationship between recall (blue) and specificity (red) of the 6 algorithms of Operon-SEQer for (A) total reads, (B) total mapped reads, and (C) percent mapped reads in each data set from M. tuberculosis (see Materials and Methods for accession numbers).



667

668 **Sup. Figure 3 – Operon-SEQer ensemble tested against new data sets.** Recall (blue) and
669 specificity (red) of the Operon-SEQer ensemble with algorithm agreement cutoffs of 1-6 for
670 operon pair calls for the new data set from (A) *B. subtilis,* (B) *M. tuberculosis,* and (C) *P. syringiae*.

671

	E. coli		B. subtilis	
	Recall	Specificity	Recall	Specificity
DOOR	0.85	0.80	0.84	0.95
Rockhopper	0.90	0.81	0.88	0.96
Operon-SEQer 1	1.00	0.85	1.00	0.64
Operon-SEQer 2	0.97	0.91	0.99	0.70
Operon-SEQer 3	0.95	0.94	0.98	0.75
Operon-SEQer 4	0.91	0.96	0.97	0.81
Operon-SEQer 5	0.85	0.99	0.95	0.88
Operon-SEQer 6	0.59	1.00	0.85	0.95

Sup Table 1 – Comparison of Operon-SEQer with DOOR and Rockhopper. Comparing the recall and specificity of DOOR and Rockhopper with the Operon-SEQer ensemble (with agreement of anywhere between 1 and 6 of the algorithms that make up Operon-SEQer being used to make operon pair calls). Heat map colors range from yellow (lowest) to white (mid-point) to blue (highest).