# Predicting bacterial growth conditions from mRNA and protein abundances.

Mehmet U. Caglar1, 2, 3, Claus O. Wilke3, 4, 5\*

1Department of Integrative Biology, The University of Texas at Austin, Austin, Texas, USA

2Center for Computational Biology and Bioinformatics, The University of Texas at Austin, Austin, Texas, USA

3Institute for Cellular and Molecular Biology, The University of Texas at Austin, Austin, Texas, USA

4Center for Systems and Synthetic Biology, The University of Texas at Austin, Austin, Texas, USA

5Department of Molecular Biosciences, The University of Texas at Austin, Austin, Texas, USA

\*Corresponding author: [wilke@austin.utexas.edu](mailto:wilke@austin.utexas.edu) (COW)

## Abstract

Predicting bacterial phenotype from external perturbations is a problem investigated many times; on the other hand, the question of predicting external conditions by using phenotype is not investigated much. Here we use a *E.coli* transcriptomic and proteomic dataset to predict the growth conditions. Our findings indicate that we can make far from random predictions related with carbon source, salt levels and growth phase of the bacteria from both transcriptome and proteome data. We can also predict multiple conditions for a sample up to 75% accuracy. There is a small synergistic effect related with combining transcriptome and proteome data. It is also possible to make predictions for continuous parameters like the time that the sample was taken and salt concentrations. The analysis indicates 2 clear signals exponential data have a higher predictive value and combining mRNA and protein data results in a small but consistent increase in prediction power independent of the machine learning model used.

## Introduction

## Results

### Data structure and pipeline design

We used data from previous study [our paper cite] to generate predictive models based on mRNA and protein abundances to generate predictive models that try to figure out growth conditions that the sample is collected. Although the methods and procedures is general and can be applied to different growth conditions here we focus on prediction of four different parameters that are systematically varies in the data; growth phase, carbon source, Mg and Na concentrations. Data set composes from 155 samples, mRNA abundances were measured for 152 of them and protein abundances were measured for 105 of them. For 102 of the samples we have both mRNA and protein concentrations. [figure 1]

For the analyze of data we generated a pipeline using four different machine learning models including SVM with radial kernel, SVM with sigmoidal kernel, SVM with linear kernel and random forest algorithm with the help of *e1071*1 *and random forest*2 *packages*. We use *C-Classification* for training classification model and *eps-regression* for training regression models which we use for predicting growth rates. We adjusted weights of samples in a way that each class ends up with equal weight in order to prevent the prediction bias in favor of more populated classes.

We also generate a tuning loop for the analyze in which we were optimizing cost value for models SVM with linear, radial and sigmoidal kernels, and gamma for SVM with radial and sigmoidal kernels. For random forest algorithm, we optimize mtry, ntrees, and nodesize parameters. We use the multi conditional f1 score3, in order to weight all conditions equally and assign same importance to false positives and false negatives.

Before SVM we apply DeSeq24 for size factor normalization, fSVA5 to normalize batch effects and PCA6 to obtain the principal components of the data. Despite the fact that our dataset has strong batch effects [our paper cite], we believe we get rid of the batch effects as much as possible before training our data with the help of fSVA algorithm. We write our tuning algorithm that enables us to divide the dataset into subsets semi-randomly in a way that the ratios of samples tried to be constant as much as possible between training tuning and test sets. We also calculate the conditional class weights for each training data in tuning process [figure 2].

So overall our pipeline is designed for tuning four different models, SVM with linear kernel, SVM with radial kernel, SVM with sigmoidal kernel, and random forest; independent of weight and batch effects and by using principal components in order to prevent overfitting and assigning same importance to false positives, false negatives through all individual conditions.

### We can make predictions on both datasets with all four models

Predicting 4 different components carbon source, Mg levels, Na levels and growth phase which makes 16 distinct conditions at the same time on the test dataset is a challenging task. We apply our pipeline and the results indicate we can make reasonable predictions by using all our algorithms there are clear winners in the tuning stage. The table1 shows the winning models for mRNA and protein data in the tuning stage, as can be seen SVM with radial kernel is the winner in mRNA data, and SVM with sigmoidal kernel is the winner in protein data. The best parameter sets after tuning for mRNA and protein samples are xx, yy as cost and xx, yy as gamma, for radial and sigmoid kernels respectively (Supplementary fig 1-2). Although the test set scores are less significant and more similar to each compared to tuning set results, the same trends can be observed in test set results (Figure2). The dramatic drop between tuning set results and test set results is due to xxx. Although the results on test set is much less significant compared to tuning set the results for test set are still far from random, enable us to predict the 4 components of the sample conditions correctly 62.7%of the time for mRNA and 54.3% of the time for protein data if the number of test set examples are equally distributed and independent from the actual sample set of training set. Corresponding to *multi condition F1 score* of mRNA and protein data are 0.63 and 0.56 respectively. (Figure 3). If we look at the distribution of results in a confusion matrix normalized with respect to number of training data in each category we observe that categories with high number of samples behave as an attractor slightly, i.e. we can get rid of the effect of sample sizes quite successfully. External condition of base Mg+2 base Na+1, lactate as carbon source in exponential phase is the most predictable condition for both mRNA and protein data. On the other hand, the base Mg+2, base Na+1, lactate as carbon source in stationary phase have a high predictability for protein data but not that much predictability for mRNA data (Figure 4).

### Combining mRNA and protein data causes a significant increase in predictability

We can use the pipeline to gather some information about internal workings of biological system. Figuring out the amount of distinct information between mRNA and protein data is an important task and can show how much of the information in mRNA level is lost in protein level and how much of new information generated by post transcriptional regulations. To see this, we run our pipeline on subsets of data that matches in between proteins and mRNAs. The results indicate protein data includes more information compared to mRNA data in three of four models used, and combined mRNA protein data includes more information compared to both mRNA and proteins with an exception of for radial model f1 score distribution associated with combined data is not different from protein data (Table 2, Figure 5). By using this information, we can say that after normalization for number of samples protein data contains more information about external conditions the bacteria lived in than mRNA data, in addition to that combined mRNA protein data contains more information that individual samples which indicates some information about external conditions in mRNA data was lost during translation process. The characteristics of confusion matrix for individual datasets and combined dataset indicates the pattern of false negatives and false positives do not match with each other (Supplementary figures 3-5).

### Information about external conditions lost as time passes

To see how the information content changes in time we used the sub-data sets that only have either exponential or stationary phase samples for both protein and mRNA data. Results indicate the prediction power of data for the external conditions decreases from exponential to stationary phase for both protein and mRNA data. This trend is independent from the model used. We can say that the bacteria begin to have similar composition in terms of protein or mRNA abundances as time passes regardless of external conditions (Figure 6).

### Most predictable external condition is growth phase for mRNA data and carbon source for protein data

The results indicate that the performances of models for specific data types and experiments are similar to each other and chosen model is not the main driver of successful prediction on the other hand there is a big variation between conditions. The most predictable external condition is the growth phase from mRNA data and the least predictable one is carbon source. On the other hand, the most predictable external condition is carbon source for protein data and least predictable one is the Mg+2 concentration (Figure 7 and supplementary figure xx). Comparing the results of mRNA and protein data does not make sense because of different sample sizes. These results are consistent with the “Clustering of mRNA and protein abundances by different growth conditions” table generated for the same dataset (cite)

## Discussion

We try to find methods to classify growth conditions of bacteria by analyzing its mRNA and protein concentrations, this is the inverse problem of finding concentrations of specific mRNA’s and protein under given growth conditions. Overall results indicate that we can predict the growth conditions up to 90% percent of the time. We used the dataset [cite], which includes corresponding mRNA and protein reads for analyze.

We can predict the

The biggest handicap of the work is the sample size, although the sample size is big compared to similar studies [cite], the comparison between multi variable and multi variable intersection analyses for both mRNA and proteins the prediction power decreases with larger sample set. This indicates we are not in stationary regime in terms of number of samples.

The second problem seems to be associated with sample number bias, although we made a correction with weight factors and it seems still there is a correlation between sample size and prediction performance.

## Materials and Methods

### Data

We use the same data sets that were used by the paper [cite]. For all single variable test, and also for multi-variable prediction tests except for combining mRNA and proteins we use all available data. For combination tests we use intersection of mRNA and protein samples (102 sample).

### Prediction Methodology and Parameters

The initial preparation of the data is similar to the paper [cite]. After finding suitable subsets of the data for the tests and summing up technical replicate results for proteins, we calculate size factor normalized data with DeSeq24 and apply variance stabilizing transformation7 (vst) on it.

We then divide the data into two subsets; training&tune set and test set. The division is semi random i.e. algorithm preserves the ratios of different conditions in training&tune and test subsets. We preserve the condition labels for training&tune data but we delete the labels of the samples for test set. We then apply frozen Surrogate Variable Analysis5 (fSVA) to get rid of the batch effects in the sample. The algorithm can correct the batch effects on both training&tune and test data without knowing the labels of the test data. With the fSVA algorithm we generate batch effect normalized training&tune and test datasets without knowing the labels of test data. After fSVA we use principal component analysis6 (PCA) to define principle axis of training&tune set and rotate the test data set with respect to principal axis of training&tune set. We then pick the most significant top 10 axis for both training&tune, and test data sets. The next step is to train and tune our candidate machine learning algorithms with *dimension reduced training&tune dataset* and apply those trained and tuned algorithms on *dimension reduced test dataset* to make predictions on them. In the case of combined mRNA and protein data analysis we calculate size factors with DeSeq2 algorithm and batch effects with fSVA algorithm individually then combine 2 datasets and apply PCA on combined data. We run this process and call the training and tuning algorithm 60 times with independent divisions between training&tune, and test data sets.

The training and tuning algorithm firstly divide the data as train and tune dataset semi randomly, trying to keep the ratios of individual conditions same in both training and tuning data. We repeat this division 10 times and generate 10 training and tuning dataset pairs. Then we generate a parameter grid for tuning process. We tune cost parameter for SVM with linear kernel, cost and gamma parameters for SVM with radial and sigmoid kernels, ntree, mtry, and nodesize parameters for random forest algorithms (Table 3). We use e1071 package1 for implementing support vector machines with linear, radial and sigmoid kernels that uses libSVM8 library and randomForest package2 for random forest algorithm. We apply multiple sets of tests; we either try to predict an individual parameter such as growth phase, carbon source, Mg+2 concentration or Na+ concentration; or we try to predict all multiple conditions at one single run. We run our tests on all mRNA, all protein, and combined protein and mRNA data. We run tests on all phases, only on exponential phase or only on stationary phase (Supplementary Table 1).

We train each four models in training set and make predictions on tune set for all variables that we are tuning and for all 10 training and tuning dataset pairs, we apply a class weight normalization for training of all four models; where class weights are inversely proportional to number of training samples and calculated independently for each training process. We average the errors calculated by score metric over training tuning dataset pairs and obtain an average error value estimate for each algorithm and for each parameter combination. Then we find the winner parameter combination for each algorithm, and label this parameter combination, algorithm match as winner algorithm for each of 60 independent divisions between training&tune, and test data sets.

At the end of all procedure we end up with 60\*4=240 winner algorithms with their tuning parameters and their average error values.

### Calculation of the score metric

The metric we use is multi class macro F1 score 3,9,10 that normalized over individual condition; i.e. each condition have equal weight instead of each sample.

### Statistical analysis and data availability

## References

1. Meyer, D. & Wien, T. U. Support Vector Machines. The Interface to libsvm in package e1071. Online-Documentation of the package e1071 for "R. (2001).

2. Liaw, A. & Wiener, M. Classification and Regression by randomForest. R News **2,** 18–22 (2002).

3. Sokolova, M. & Lapalme, G. A systematic analysis of performance measures for classification tasks. Inf. Process. Manag. **45,** 427–437 (2009).

4. Love, M. I., Huber, W. & Anders, S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biol. **15,** 550 (2014).

5. Parker, H. S., Bravo, H. C. & Leek, J. T. Removing batch effects for prediction problems with frozen surrogate variable analysis. ArXiv13013947 Stat (2013).

6. Jolliffe, I. in Wiley StatsRef: Statistics Reference Online (John Wiley & Sons, Ltd, 2014). doi:10.1002/9781118445112.stat06472

7. Anders, S. & Huber, W. Differential expression analysis for sequence count data. Genome Biol. **11,** R106 (2010).

8. Chang, C.-C. & Lin, C.-J. LIBSVM: A Library for Support Vector Machines. ACM Trans Intell Syst Technol **2,** 27:1–27:27 (2011).

9. Yang, Y. An Evaluation of Statistical Approaches to Text Categorization. Inf. Retr. **1,** 69–90 (1999).

10. Ghamrawi, N. & McCallum, A. Collective Multi-label Classification. in Proceedings of the 14th ACM International Conference on Information and Knowledge Management 195–200 (ACM, 2005). doi:10.1145/1099554.1099591

## Acknowledgments

This study was funded by Army Research Office (ARO,) grant W911NF-12-1-0390 to COW. COW also acknowledges support from the NIH (R01 GM088344, R01 AI120560) and the NSF (Cooperative agreement no. DBI-0939454, BEACON Center). The Texas Advanced Computing Center (TACC) at The University of Texas at Austin provided high-performance computing resources.

## Contributions

M.U.C, C.O.W. conceived the study and designed the pipeline and analyze the data.

contributed computer code used for data analysis.

M.U.C, C.O.V. prepared the figures.

M.U.C., C.O.W. wrote the initial paper draft. All authors reviewed and edited the final manuscript

## Competing interests

The authors declare no competing financial interests.