Decoding sequence-level information to predict membrane protein expression

Shyam M. Saladi[1], Nauman Javed[1], Axel Müller[1], & William M. Clemons, Jr.[1*]

[1]Division of Chemistry and Chemical Engineering, California Institute of Technology, Pasadena, CA

[*]Corresponding author

Email: clemons@caltech.edu (WMC)

Abstract

The expression of membrane proteins remains a major bottleneck in the characterization of these important proteins. Expression levels are currently unpredictable, which renders the pursuit of these targets challenging and inefficient. Evidence demonstrates that small changes in the nucleotide or amino-acid sequence can dramatically affect membrane protein biogenesis; yet these observations have not resulted in generalizable approaches to improve expression. Here, we develop a data-driven statistical model, named IMProve, that enriches for the likelihood of selecting membrane proteins that express in *E. coli* directly from sequence. The model, trained on experimental data, combines a set of sequence-derived variables resulting in a score that predicts the likelihood of expression. We test the model against various independent datasets that contain a variety of experimental outcomes demonstrating that the model significantly enriches for expressed proteins. Analysis of the underlying features reveals a significant role for nucleotide derived features in predicting expression. This computational model can immediately be used to identify favorable targets for characterization.

Author Summary

Membrane proteins play a pivotal role in biology, representing a quarter of all proteomes and a majority of drug targets. While considerable effort has been focused on improving our functional understanding of this class, much of the investment has been hampered by the inability to obtain sufficient amounts of sample. Until now, there have been no broadly successful strategies for predicting and improving expression which means that each target requires an *ad hoc* adventure. Complex biological processes govern membrane protein expression; therefore, sequence characteristics that influence protein biogenesis are not simply additive. Many properties must be considered simultaneously in predicting the expression level of a protein.

We provide a first solution to the membrane protein expression problem by learning from published data to develop a statistical model that predicts the outcomes of expression trials across families, scales, and laboratories (all independent of the model's training data). Given that the process of finding a target for large-scale expression is arduous, often requiring a long trial-and-error process that consumes significant financial and human resources, this work will have immediate applicability. The ability to study and engineer inaccessible membrane proteins becomes feasible with the use of our predictor. Furthermore, this work will enable others in developing new computational methods to assist in the experimental study of membrane proteins.

Introduction

The central role of integral membrane proteins motivates structural and biophysical studies that require large amounts of purified protein, often at considerable cost of both material and labor. Only a small percentage can be produced at high-levels resulting in membrane protein structural characterization lagging behind that of soluble proteins presently constituting just 1.7% of known atomic-level structures [1]. To increase the pace of structure determination, the scientific community created large government-funded structural genomics consortia facilities, like the NIH-funded New York Consortium on Membrane Protein Structure (NYCOMPS)[2]. For this representative example, more than 8000 genes, chosen based on characteristics hypothetically related to success, yielded only 600 (7.1%) highly expressing proteins [3] resulting to date in 34 (5.6% of expressed proteins) unique structures (based on annotation in the RCSB PDB [4]). This highlights the funnel problem of structural biology where each stage of the structure pipeline eliminates a large percentage of targets compounding into an overall low rate of success [5]. With new and rapidly advancing technologies like cryo-electron microscopy and micro-electron diffraction, we expect that the latter half of the funnel, structure determination, will increase in success rate [6,7]. In any case, membrane protein expression will continue to limit targets accessible for study [8].

Tools for improving the number of expressed membrane proteins are needed. While significant work has shown promise on a case-by-case basis, *e.g.* growth at lower temperatures, codon optimization [9], and regulating transcription [10], a generalizable solution remains elusive. Currently, each target must be addressed individually as the conditions that were successful for a previous target seldom carry over to other proteins, even amongst closely related homologs [5,11]. For individual cases, simple changes can have dramatic effects on the amount of expressed proteins [12,13]. Considering the scientific value of membrane protein studies, it is surprising that there are no methods that can provide solutions for improved expression outcomes with broad applicability across protein families and genomes.

Currently no approaches are available that decode sequence-level information for predicting membrane protein expression; yet the concept that sequence variation can measurably influence membrane protein biogenesis is commonplace. For example, positive-charges on cytoplasmic loops are important determinants of membrane protein topology [14,15]; yet introduction of mutations presumed to enhance certain properties, such as the positive inside rule, has not proven generalizable for improving expression [11]. The reasons for this likely lie in the complex underpinnings of membrane protein biogenesis, where the interplay between sequence features at the protein and nucleotide levels must be considered. Optimizing for a single sequence-level feature likely diminishes the beneficial effect of other features (*e.g.* increasing positive residues on internal loops might diminish favorable mRNA properties). Without accounting for the broad set of features related to membrane protein expression, it is impossible to predict differences in expression.

Attempts to develop algorithms that predict membrane protein expression have failed. Several examples, Daley, von Heijne, and coworkers [9,16,17] as well as NYCOMPS, were unable to use experimental expression data sets to train models that returned any predictive performance (personal communication). Statistical tools have been developed to predict expression and/or crystallization propensities from sequence information based on outcomes. These are primarily based on results from the Protein Structure Initiative where experimental outcomes are deposited in TargetTrack[18,19] and include well-known methods such as SPINE[20], Xtalpred[21–23], and P_{XS}[24] as well as others[25–35]. While collectively these methods have supported significant advances in biochemistry, each suffers from similar issues when predicting membrane protein outcomes due to the criteria applied during the model development process. As membrane proteins have an extremely low success rate compared to soluble proteins, they are either explicitly excluded from the training process or are implicitly down-weighted by the statistical model. The result is that these methods do not predict membrane protein expression (representative methodology [21]).

In an ideal world, a perfect predictor would define the subset of protein sequences that can be expressed in a given host. As discussed elsewhere [9,16,17], none have successfully been able to map membrane protein expression to sequence. Given the scale of difficulty in expressing membrane proteins, we demonstrate here for the first time that it is possible to predict membrane protein expression purely based on sequence allowing one to enrich their expression trials for proteins with a higher probability of success.

To connect sequence to prediction, we develop a statistical model that maps a set of sequences to experimental expression levels via calculated features—thereby simultaneously accounting for the many potential determinants of expression. The resulting model allows ranking of any arbitrary set of membrane protein sequences in order of their relative likelihood of successful expression. In this first demonstration of prediction, we sought to select the simplest framework necessary to capture the problem. In particular, we train a linear equation that provides a score based on calculating the sum of weighted features where the weights are derived from fitting to experimental expression data, a "training set." These features attempt to encapsulate the corpus of work that shows that sequence-level characteristics are important determinants of protein biogenesis, *e.g.* RNA secondary structure [36,37], transmembrane segment hydrophobicity [38–40], the positive inside rule [41], and loop disorder [42].

We extensively validate our model against a variety of independent datasets demonstrating its generalizability. This model can be used broadly to score any membrane protein based on its calculated features. In the process, we have built a method to enrich for positive expression outcomes with respect to the low positive rate attained from randomly selecting targets. To support further experimental efforts, we showcase the performance of the model across protein families and we broadly score the membrane proteome from a variety of important genomes. This approach and resulting model provides an exciting example for connecting sequence space to complex experimental outcomes.

Results

For this study, we focus on heterologous expression in *E. coli*, due to its ubiquitous use as a tool for membrane protein expression. While the benefits derived from low cost and low barriers for adoption are obvious, the applicability to the spectrum of the membrane proteome are becoming clearer. Of note, 43 of the 216 unique eukaryotic membrane protein structures were solved using protein expressed in *E. coli* (based on annotation in the RCSB PDB [4]). This demonstrates the utility of *E. coli* as a broad tool and its potential if the expression problem can be overcome.

Development of a computational model trained on *E. coli* expression data

A key component of any data-driven statistical model is the choice of dataset used for training. Having searched the literature, we identified two publications that contained quantitative datasets on the IPTG-induced overexpression of *E. coli* polytopic membrane proteins in *E. coli*. The first set, Daley, Rapp *et al.*, contained activity measures, proxies for expression level, from C-terminal tags of either GFP or PhoA (alkaline phosphatase)[16]. The second set, Fluman *et al.*, used a subset of constructs from the first and contained a more detailed analysis utilizing in-gel fluorescence to measure folded protein[43] (see Methods 4c). The expression results strongly correlated (Spearman's $\rho = 0.73$) between the two datasets demonstrating that normalized GFP activity was a good measure of the amount of folded membrane protein (Fig 1A and [43,44]). The experimental set-up employed multiple 96-well plates over multiple days resulting in pronounced variability in the absolute expression level of a given protein between trials. Daley, Rapp *et al.* calculated average expression levels by dividing the raw expression level of each protein by that of a control construct (Inverse LepB-GFP or LepB-PhoA) on the corresponding plate. While the resulting values were useful for the relevant question of identifying topology, we were unable to successfully fit a linear regression or a standard linear Support Vector Machine (SVM) to predict either the raw expression data compiled from all plates or averaged outcomes of each gene using numerical features calculated from nucleotide and protein sequences (see S1 Table;

Methods 2,3). This unexpected outcome suggested that the measurements required a more complex analysis.

Fig 1. Training performance. (**A)** A comparison of GFP activity [16] with measured folded protein [43] where each point represents the mean for a given gene tested in both works, and error bars plot the extrema. Spearman's rank correlation coefficient and 95% confidence interval (CI) [45] are shown. **(B)** Plates are the number of independent sets of measurements within which expression levels can be reliably compared. Genes are the number of proteins for which the C-terminus was reliably ascertained [16]. Observations are the total number of expression data points accessible. Total pairs are the number of comparable expression measurements (*i.e.* those within a single plate). Kendall's τ is the metric maximized by the training process (See Methods 4b). The color of the column heading identifying each experimental set is retained throughout the figure. **(C)** Agreement against the normalized outcomes plotted as the mean activity (see Methods 5 for definition) versus the score with error bars providing the extent of observed activities (Spearman's ρ and 95% CI noted). **(D)** Illustrative Receiver Operating Characteristics (ROC) for thresholds at 25th and 75th percentile in activity with the number of positive outcomes at that threshold, the Area Under the Curve (AUC), and 95% CI indicated. **(E)** The AUC of the ROC at every possible activity threshold.

We hypothesized that measurements could be more accurately compared within an individual plate then across the entire dataset. To account for this, a preference-ranking linear SVM algorithm (SVM^{rank} [46]) was chosen (see Methods 4b). Simply put, the SVM^{rank} algorithm determines the optimal weight for each feature to best rank the order of expression outcomes within each plate over all plates, which results in a model where higher expressing proteins have higher scores. The outcome is identical in structure to a multiple linear regression, but instead of minimizing the sum of squared residuals, the SVM cost function is used accounting for the plate-wise constraint specified above. In practice, the process optimizes the correlation coefficient Kendall's τ, as a training metric, to converge upon a set of weights. Kendall's τ measures the agreement between ordinal quantities by calculating the number of correctly ordered and swapped pairs.

Various metrics related to the training data can be derived to assess the accuracy with which the model fits the input data (see Methods 4c). The SVM^{rank} training metric shows varying agreement for all groups (*i.e.*, $\tau_{kendall} > 0$) (Fig 1B). For individual genes, activity values normalized and averaged across trials were not directly used for the training procedure (see Methods 4a); yet one would anticipate that scores for each gene should broadly correlate with the expression average. Indeed, the observed normalized activities positively correlate with the score (dubbed IMProve score for Integral Membrane Protein expression improvement) output by the model (Fig 1C). Since SVM^{rank} transforms raw expression levels within each plate to ranks before training, there is no expectation or guarantee that magnitude differences in expression level manifest in magnitude differences in score. As a result, Spearman's ρ, a rank correlation coefficient describing the agreement between two ranked quantities, is better suited for quantifying correlation over more common metrics like the R^2 of a regression and Pearson's r.

For a more quantitative approach to assessing the model's success within the training data, we turn to the Receiver Operating Characteristic (ROC). ROC curves quantify the tradeoff between true positive and false positive predictions across the numerical scores output from a predictor. This is a more reliable assessment of prediction than simply calculating accuracy and precision from a single, arbitrary score threshold [47]. The figure of merit that quantifies a ROC curve is the Area Under the Curve (AUC). Given that the AUC for a perfect predictor corresponds to 100% and that of a random predictor is 50% (Fig 1D, grey dashed line), an AUC greater than 50% indicates predictive performance of the model (percentage signs hereafter omitted) (see Methods 5 and [47]). Here, the ROC framework will be used to quantitatively assess the ability of our model to predict the outcomes within the various datasets.

The training datasets are quantitative measures of activity requiring that an activity threshold be chosen that defines positive or negative outcomes. For example, ROC curves using two distinct activity thresholds, at the 25th or 75th percentile of highest expression, are plotted with their calculated AUC values (Fig 1D). While both show that the model has predictive capacity, a more useful visualization would consider all possible activity thresholds. For this, the AUC value for every activity threshold is plotted showing that the model has predictive power regardless of an arbitrarily chosen expression threshold (Fig 1E). In total, the analysis demonstrates that the model can rank expression outcomes across all proteins in the training set. Interestingly, for PhoA-tagged proteins the model is progressively less successful with increasing activity. Since PhoA activity is an indirect measure of expression of proteins with their C-termini in the periplasm, this brings into question either the utility of this quantification method relative to GFP activity or perhaps that this class of proteins are special in the model. An argument for the former is presented later (Fig 2E).

Fig 2. Success of the model against outcomes from NYCOMPS. (A) An overview of the NYCOMPS outcomes and **(B)** a histogram of the number of conditions tested per gene colored based on outcome. **(C)** Receiver Operating Characteristics for positive groupings given by Only Positive outcomes genes (red) and genes with at least one positive outcome (pink). The percent positive for each group (corresponding color), total counts (black), and Area Under the Curve (AUC) values with 95% Confidence Interval (CI) are shown. The ROC considering genes with Mixed outcomes only as positive is shown as a blue dashed line with an AUC of 53.5 (51.8-55.2). The grey dashed line shows the performance of a completely random predictor (AUC = 50). **(D)** Histograms of genes with Only Positive (red) and Only Negative outcomes (grey) across IMProve scores (binned as described in Methods 5). The percentage of Only Positive outcomes in each bin is overlaid as a brown line (right axis). **(E)** The Positive Predictive Value (PPV) plotted for each percentile IMProve score, *e.g.* 75 on the x-axis indicates the PPV for the top 25% of genes based on score for genes, where positive indicates genes with Only Positive outcomes. The dashed line shows the overall success rate of the NYCOMPS experimental outcomes (~11% Only Positive). **(F)** The fold change in the PPV as a function of IMProve score relative to the success rate of NYCOMPS. **(G)** The AUCs for outcomes in each individual plasmid and solubilization condition (DDM except LDAO where noted) along with 95% CI (numerically in S2 Table). Performances are also split by predicted C-terminal localization [48]. The numbers below indicate the total number of trials for each group and the percent within that group that were positive.

Demonstration of prediction against an independent large expression dataset

While the above analyses show that the model successfully fits the training data, we assess the broader applicability of the model outside the training set based on its success at predicting the outcomes of independent expression trials from distinct groups and across varying scales. The first test considers results from NYCOMPS, where 8444 membrane protein genes entered expression trials, in up to eight conditions, resulting in 17114 expression outcomes (Fig 2A) [2]. The majority of genes were attempted in only one condition (Fig 2B), and, importantly, outcomes were non-quantitative (binary: expressed or not expressed) as indicated by the presence of a band by Coomassie staining of an SDS-PAGE gel after small-scale expression, solubilization, and nickel affinity purification [3]. For this analysis, the experimental results are either summarized as outcomes per gene or broken down as raw outcomes across defined expression conditions. For outcomes per gene, we can consider various thresholds for considering a gene as positive based on NYCOMPS expression success (Fig 2B). The most stringent threshold only regards a gene as positive if it has no negative outcomes ("Only Positive", Fig 2B, red). Since a well expressing gene would generally advance in the NYCOMPS pipeline without further small-scale expression trials, this positive group likely contains the best expressing proteins. A second category comprises genes with at least one positive and at least one negative trial ("Mixed", Fig 2B, blue). These genes likely include proteins that are more difficult to express.

ROCs assess predictive power across these groups (Fig 2C). IMProve scores markedly distinguish genes in the most stringent positive group (Only Positive) from all other genes (Fig 2C red). A permissive threshold considering genes as positive with at least one positive trial (Only Positive plus Mixed genes) shows more moderate predictive power (Fig 2C pink, AUC = 59.7 versus 67.1). If instead solely the Mixed genes are considered positive (excluding the Only Positive), the difference in the two positive groups is clear as the model very weakly distinguishes the mixed group from Only Negative genes (Fig 2C dashed blue, AUC = 53.5 (51.8-55.2)). This likely supports the notion that this pool largely consists of more difficult-to-express genes. For further analysis of NYCOMPS, we focus on the Only Positive pool as this likely represents the pool of best expressing proteins.

This predictive power can be qualitatively visualized as a histogram of the IMProve scores for genes separated by protein group (Only Positive, red; Only Negative, grey) (Fig 2D). Visually, the distribution of the scores for the Only Positive group is shifted to a higher score relative to the Only Negative group. This is emphasized considering the dramatic increase in the percentage of positive genes as a function of increasing IMProve score (overlaid as a brown line). A major aim of this work is to enrich the likelihood of choosing positively expressing proteins. The positive predictive value (PPV, true positives ÷ predicted positives) becomes a useful metric for positive enrichment as it conveys the degree of improved prediction over the experimental baseline of the dataset. The PPV of the model is plotted as a function of the percentile of the IMProve score for the Only Positive group (Fig 2E). In the figure, the experimental baseline is represented by a dashed line (11.1%); therefore, a relative increase reflects the predictive power of the algorithm. For example, considering the PPV of 20% for the top fourth of genes by IMProve score (75^{th} percentile) shows that the algorithm increases the positive outcomes by 9% over baseline. For further illustration, we plot the fold-change in PPV across the various thresholds (Fig 2F). Here, if only genes with an IMProve score greater than -0.21 (75^{th} percentile) were tested, the experiments would have returned nearly twice as many positives, a 1.82 fold change (Fig 2D). Higher score cut-offs would have even better returns.

Because there were eight different expression conditions, a final consideration looks at the NYCOMPS data based on the type of trial. Importantly, the model shows consistent performance throughout each of the eight conditions tested (Fig 2F, numerically in S2 Table). This highlights that the model is not sensitive to the experimental design of the training set and appears to predict broadly against different vector backbones. With this in mind, as an overall perspective, using a reasonable threshold for IMProve score (91^{st} percentile or 0.5 (Fig 2E, yellow line)), had NYCOMPS tested the same number of genes an additional 1207 proteins would have been positive, representing a significant improvement in the return on investment.

The ability to predict the experimental data from NYCOMPS allows returning to the question of alkaline phosphatase as a metric for expression. For the training set, proteins with C-termini in the periplasm show less consistent fitting by the model (Fig 1, orange). To assess the generality of this result, the NYCOMPS outcomes are split into pools for either cytoplasmic or periplasmic C-terminal localization and AUCs are calculated for each. There are no significant differences in predictive capacity across all conditions (Fig 2G, green vs. orange) demonstrating that the model is applicable for all topologies.

Further demonstration of prediction against small-scale independent datasets

The NYCOMPS example demonstrates the predictive power of the model across the broad range of sequence space encompassed by that dataset. Next, the performance of the model is tested against relevant subsets of sequence space (*e.g.* a family of proteins or the proteome from a single organism), which are reminiscent of laboratory-scale experiments that precede structural or biochemical analyses. While a number of datasets exist [5,49–59], we identified six for which complete sequence information could be obtained to calculate all the necessary sequence features [49–54].

The first dataset is derived from the expression of 14 archaeal transporters in *E. coli* chosen based on their homology to human proteins [49]. For each putative transporter, expression was performed in three plasmids and two strains (six total conditions) with the membrane fraction quantified by both a Western blot against a histidine-affinity tag and Coomassie Blue staining of an SDS-PAGE gel. Here, the majority of the expressing proteins fall into the higher half of the IMProve scores, 7 out of 9 of those with multiple positive outcomes (Fig 3A). Strikingly, quantification of the Coomassie Blue staining highlights a clear correlation with the IMProve score where the higher expressing proteins have the highest score (Fig 3B). ROC curves are plotted for the two thresholds: expression detected at least by Western blot or, for the smaller subset, by Coomassie Blue (Fig 3C). In both cases, the model shows predictive power. Consistent with what was seen for NYCOMPS, selecting only the top half of proteins by IMProve score would have captured the majority of the positive outcomes.

Fig 3. Success of the model against a variety of small scale outcomes. For each set, vertical lines indicate the median IMProve score. Receiver Operating Characteristics (ROC) along with Areas Under the Curves (AUC) and 95% confidence interval as well as the total number of positives for the given threshold (red hues) along with the total outcomes (black) are presented. In each curve, increasing expression thresholds as defined by the original publication are displayed as deeper red. **(A,B)** The expression of archaeal transporters in up to 6 trials. **(A)** Positive expression count is plotted above the dashed line and negative outcomes below the line. **(B)** From the same work, the expression of proteins detected by Coomassie Blue [49]. **(C)** ROC curves for each positive threshold (*i.e.* Coomassie Blue or Western Blot) from trials in **A,B**. **(D)** Experimental expression of *M. tuberculosis* membrane proteins plotted based on outcomes. **(E)** ROC curves for each possible threshold from trials in **D**. **(F)** Mammalian GPCR expression in either *E. coli* (top) or *P. pastoris* (bottom). **(G)** ROC curves for each possible threshold from trials in **F**.

The next test considers the expression of 105 *Mycobacterium tuberculosis* proteins in *E. coli* [50]. Protein expression was measured both by Coomassie Blue staining of an SDS-PAGE gel and Western blot with only outcomes from the membrane fraction considered for this analysis. The highest expressing proteins (detected via Coomassie Blue) follow the trend given by the IMProve score with 7 of the 9 falling within the higher half of scoring proteins (Fig 3D) and is reflected in the ROC (Figure 3E). In contrast, using the positive Western blot outcomes as the minimum threshold (Fig 3D) shows an AUC no better than random (Fig 3E). Given that no internal standard was used and that each expression trial was performed only once, proteins that were positive by Western blot may represent a pool indistinguishable in expression from those not detected; alternatively, these results support that IMProve accurately captures the most highly expressing proteins. Again, selecting only the top half of the proteins based on their IMProve score would have captured nearly all of the high expressing proteins.

A broader test considers expression trials of 101 mammalian GPCRs in bacterial and eukaryotic systems [51]. Trials in *E. coli*, measured via Western blot of an insoluble fraction, again show highly expressing proteins at higher IMProve scores while the expression of the same proteins in *P. pastoris*, measured via dot blot, fail to show broad agreement (Fig 3F,G). The lack of predictive performance in *P. pastoris* suggests that the parameterization of the model, calibrated for broadly characterizing *E. coli* expression, requires retraining to generate a different model that captures the distinct interplay of sequence parameters in yeast. Still, the higher IMProve score clearly enriches for expressing proteins in *E. coli*.

Further expression trials of membrane proteins from *H. pylori*, *T. maritima* as well as microbial secondary transporters continues to show the same broad agreement [52–54] (S1 Fig). *H. pylori* membrane proteins showed that as the threshold for positive expressing proteins increases, the performance of the model improves (using the highest threshold n=46 and AUC=67.7) (S1 Fig. A,B). For *T. maritima* expression, the model weakly captures outcomes for two defined thresholds (n=5 and 19, AUC=61.7 and 58.7), but due to the small number of successful outcomes, the confidence intervals are broad (S1 Fig. C,D). The expression of microbial secondary transporters shows varied agreement with the model. Taking proteins at the lower defined expression threshold shows predictive performance (n=59, AUC=60.5); however, considering the defined high-expressing proteins is less conclusive (n=26, AUC=52.0) (S1 Fig. E,F). Broadly, independent of laboratory and experimental set-up, the IMProve score can enrich for the highest expressing proteins.

Performance of the model across protein families

To provide a clear path forward for experiment, we consider the performance of the model with regards to protein homology families, as defined by Pfam family classifications [60]. The 8444 genes in the NYCOMPS dataset fall into 555 families with ~15% not classified. To understand whether IMProve score is biased towards families present in the training set, we separate genes in the NYCOMPS dataset into three groups: part of the 153 families found in the training set, family not in the training set, and no defined Pfam family. There is no significant difference in AUC at 95% confidence between these groups (Fig 4A, bottom row). Therefore, the predictive power for a gene does not depend on the presence of its family within the training set.

Fig 4. Model performance across protein families. (A) The NYCOMPS dataset split by the presence or absence of a Pfam family in the training set with AUCs calculated by considering Only Positive genes as positive outcomes. **(B)** For each family within NYCOMPS with at least five outcomes (including one positive and one negative), the AUC across all outcomes is plotted with horizontal bars indicating the 95% confidence interval. The color indicates the significance of the prediction within the family: purple, predictive at 95% confidence, blue, predictive but not at 95% confidence, green, not predictive. The size of each significance group and total number of families (grey) are indicated on the plot. **(C)** Outcomes for specific protein families with an optimal IMProve score threshold indicated. Each was only tested in a single condition (N: His-FLAG-TEV-gene). CopD is classified as TCDB 9.B.62 and AtoE as TCDB 2.A.73 [61]. **(D)** For the families in **C**, a ROC curve with the overall positive percentage within the group, total number of outcomes, and AUC with 95% CI is labelled.

The scale of NYCOMPS allows us to investigate whether there are protein families for which the model does better or worse than the aggregate. For this, an AUC is calculated for each protein family that has minimally five total outcomes (including at least one positive and one negative). Fig 4B plots the AUC for each protein family in increasing order as a cumulative distribution function. The breadth of the AUC values highlights the variability in predictive power across families. Most families can be predicted by the model (115 of 159 have an AUC > 0.5, visually blue and purple) though some not at 95% confidence (57 of 115, blue), likely due to an insufficient number tested. Therefore, the NYCOMPS dataset provides some perspective on the protein families that IMProve best predicts.

For the protein families that are well-predicted within the NYCOMPS set, IMProve gives highly accurate insight into the likelihood of expression of a given protein. We demonstrate the utility of this prediction by looking at protein families that have yet to be characterized structurally. While there are a number of choices, one example is the protein family annotated as copper resistance proteins (CopD, PF05425), that typically contains eight transmembrane domains with an overall length of ~315 amino acids. A second example is the protein family annotated as short-chain fatty-acid transporters (AtoE, PF02667), that typically contains 10 transmembrane domains with an overall length of ~450 amino acids. In Fig 4C, genes from the two families are plotted by IMProve score and colored by outcome. In both cases, as indicated by the ROCs (Fig 4D), the model provides a clear score cut-off to guide target selection for future expression experiments. For example, considering CopD homologs, one would expect that those with IMProve scores above -1 will have a higher likelihood of expressing than on average across all homologs. This analysis can be broadly applied across the families that are predicted with high accuracy (S3 Table).

Forward predictions on genomes of interest

The model successfully enriches for heterologous expression of membrane proteins in *E. coli* strikingly across scales, laboratories, quantification methods, and protein families supporting its broad generalizability. While few genes express in every condition tested (Fig 2B and 3A), IMProve predicts the likelihood that a gene will express within a set of conditions and enriches for those that will work in any condition (Fig 2G, numerically in S2 Table).

To expand on the utility of this model, IMProve scores were calculated for membrane proteins from a variety of metazoan and microbial genomes (Fig 5A and S2 Fig. A). Many genomes have a significant proportion of proteins with high scores particularly evidenced by portions of the distributions ahead of the median in *E. coli* given by the vertical dashed line (Fig 5A). The likelihood for successful expression may be inferred by equating IMProve score with the PPV of Only Positive gene outcomes within the NYCOMPS dataset which rises significantly at scores above zero (Fig 5B). The range of scores spans those representative of high-expressing membrane proteins in both *E. coli* (Fig 1C) as well as in the NYCOMPS dataset (Fig 2C) and provides suggested targets for future biophysical studies (S4 Table).

Fig 5. Forward predictions of membrane protein expression for various genomes. (A) Calculated scores for proteins from a variety of genomes (count in parentheses; complete set provided in S2 Fig. A) plotted as contours of kernel density estimates of the number of proteins at a given score. Amplitude is only relative within a genome. The dot indicates the median, and the lines depict quantities of an analogous Tukey boxplot[62,63]. The vertical line shows the median score in *E. coli* to provide context for other distributions. **(B)** PPV of Only Positive gene outcomes within the NYCOMPS dataset. **(C)** Distribution of overlap coefficients (see Methods 7) for each sequence parameter comparing the entire *E. coli* membrane proteome vs. the training set from *E. coli*. The dashed line provides a threshold separating the cluster of highly-related features from those with lower overlap. (**D-F)** A comparison of overlap coefficients with the training set between NYCOMPS and **(D)** all forward predictions (S2 Fig. A), **(E)** thermophilic genomes (orange), or **(F)** *P. falciparum*. Mean Absolute Deviation is indicated for each plot.

The predictions present several surprises at the biological level. One such is that the distribution of membrane proteins from representative thermophilic bacterial genomes have generally lower relative IMProve scores than other genomes, which implies that these proteins, on average, are harder to express in *E. coli*. This is in contrast to the many empirical examples of proteins from thermophiles which are often primary targets of biophysical characterization, although analysis of structural genomics data of soluble proteins suggests only a small crystallization advantage for this group [24]. In the case of the malarial parasite *P. falciparum*, the inverse trend is true with higher than anticipated relative IMProve scores despite the expectation that these proteins would be hard to express in *E. coli*. A possible cause for the distribution of scores may lie in the differences in the features that define the proteins in these particular groups. As the training set consists only of native *E. coli* sequences, the range of values for each feature in the training set may not represent the full range of possible values for the feature. For the special cases highlighted, perhaps the underlying sequence features fall into a poorly characterized subset of sequence space bringing into question the applicability of the model for these cases.

To address the utility of the model relative to differences in the sampling of sequence features, we measure the overlap of the distributions of sequence features used for prediction (S1 Table) for a given subset (see Methods 7) (S2 Fig B). Simply put, if two subsets contain the same distribution of sequence features the expectation is that a given feature should approach 100%. In the simplest case, comparing the distribution of sequences features in all *E. coli* membrane proteins against the subset used in the training set shows that the majority of features have overlap values over 75% (Fig 5C), which provides a lower threshold for similarity of sequence feature range. For NYCOMPS sequences, most of the overlap values relative to the training set are above the threshold. As this set shows predictive performance, comparison to the training set provides a baseline to assess the reliability of predictions within other subsets (Fig 5D-F, x-axis). In the first case (Fig 5D), there is a strong correlation between all the forward predictions and NYCOMPS, *i.e.* values are near the diagonal (quantified by a Mean Absolute Deviation (MAD) = 11.6), suggesting that differences in feature space do not significantly affect the predictive power of the model. For the thermophiles subset (Fig 5E), the values again are close to the diagonal (*i.e.* low MAD = 10.6) implying that the predictions are credible. *P. falciparum* (Fig 5F), on the other hand, clearly shows stark differences as most features fall below the 75% cut-off (MAD = 29.0) bringing into question the reliability of these predictions. A training set with broader coverage of the feature space may generate a better predictor for all genomes.

Biological importance of various sequence features

Using a simple proof-of-concept linear model has allowed for a straightforward and useful predictor. Understanding if any single biological determinant is driving prediction may provide insight into membrane protein biogenesis and expression. With a linear model, as employed here, this task is ordinarily straightforward; assuming features are distributed identically and independently ("i.i.d."), the weight assigned to each feature corresponds its relative importance. However, in our case, the input features do not satisfy these conditions, *i.e.* a lack of uniformity in feature distributions (S2 Fig B) and significant correlation between individual features (S3 Fig). As a result, during the training procedure, unequal weight is placed across correlating features that represent the same underlying biological phenomena, thereby, complicating the process of determining the biological underpinnings of the IMProve score. For example, the importance of transmembrane segment hydrophobicity is distributed between several features: among these the average $\Delta G_{insertion}$ [40] of TM segments has a positive weight whereas average hydrophobicity, a correlating feature, has a negative weight (S1 Table, S3 Fig). As many features, such as those related to hydrophobicity, are correlated; conclusive information cannot be obtained simply using weights of individual features to interpret the relative importance of their underlying biological phenomena. We address this complication by coarsening our view of the features to two levels: First, we analyze features derived from protein versus those derived from nucleotide sequence, and then we look more closely at features groups after categorizing by biological phenomena.

The coarsest view of the features is a comparison of those derived from protein sequence versus those derived from nucleotide sequence. The summed weight for protein features is around zero, whereas for nucleotide features the summed weight is slightly positive suggesting that in comparison these features may be more important to the predictive performance of the model (Fig 6A). Within the training set, protein features more completely explain the score both via correlation coefficients (Fig 6B) as well as through ROC analysis (Fig 6C). However, comparison of the predictive performance of the two subsets of weights shows that the nucleotide features alone can give similar performance to the full model for the NYCOMPS dataset (Fig 6D). Within the small-scale datasets investigated, using only protein or nucleotide features shows no difference in predictive power at 95% confidence (Fig 6E). It is important to note that this does not suggest that protein features are not important for membrane protein expression. Instead, within the context of the trained model, nucleotide features are critical for predictive performance for a large and diverse dataset such as NYCOMPS. This finding corroborates growing literature that the nucleotide sequence holds significant determinants of biological processes [36,43,64–66].

Fig 6. Feature contributions to the model. (A) Classifying features by the type of sequence they are calculated from. **(B)** Considering the training set (as in Fig 1), Spearman correlation coefficients with 95% confidence intervals using individual feature categories for each grouping of data within the training set of *E. coli* membrane proteins. Colors indicate the subset being assessed (green, whole cell GFP fluorescence; orange, alkaline phosphatase activity; purple, folded protein by in-gel fluorescence). **(C)** Protein/nucleotide feature dependence within the training set substantiated by the AUC of the ROC at every possible activity threshold for feature subsets independently (as in Fig 1E). **(D)** The AUC and 95% confidence intervals using only protein or nucleotide features. (**E)** Protein/nucleotide feature dependence across small scale datasets shown as AUCs of the ROC along with 95% CI for the condition with the best overall predictive power (black).

To understand whether we may be able to provide more detailed evidence for feature importance, we collapse conceptually similar features into categories that allow for potential biological interpretation (S1 Table). As compared to the entire set of individual features, this process substantially reduces inter-feature correlation (S3 Fig, S4 Fig B). For example, the hydrophobicity group incorporates sequence features such as average hydrophobicity, maximum hydrophobicity, $\Delta G_{insertion}$, etc. The full list of groupings is provided in S1 Table and S3 Fig.

Analysis of categories suggests the phenomena that drive prediction. To visualize this, the collapsed weights are summarized in Fig 6B where each bar contains individual feature weights within a category. Features with a negative weight are stacked to the left of zero and those with a positive weight are stacked to the right. A red dot represents the sum of all weights, and the length of the bar gives the total absolute value of the combined weights within a category. Ranking the categories based on the sum of their weight suggests that some of categories play a more prominent role than others. These include properties related to transmembrane segments (hydrophobicity and TM size/count), codon pair score, loop length, and overall length/pI.

To explore the role of each category in prediction, the performance of the model is assessed using only features within a single category at a time. First understanding which categories perform well in the training set indicates which feature the model pulls information from and suggests hypotheses as to which categories ought to perform well across the validation datasets. Since the outcomes within the training set are real-valued, predictive power can be assessed via correlation coefficients with the predicted score yielding a single number (as in Fig 1C) or through AUCs across all possible expression thresholds (as in Fig 1D,E). Using the former metric, for simplicity, to assess the predictive capacity of feature subsets within the training set (Fig 6C) suggests several of interest with high correlation coefficients including 5' Codon Usage, Length/pI, Loop Length, and SD-like Sites. Only Length/pI shows some predictive across subsets of the NYCOMPS dataset (S4 Fig D).

Importantly, careful analysis of the training and large-scale testing dataset shows that no feature category independently drives the predictor. Excluding each individually does not significantly affect the overall predictive performance, except for Length/pI (isoelectric point) (S4 Fig D). Sequence length composes the majority of the weight within this category and is one of the highest weighted features in the model. This is consistent with the anecdotal observation that larger membrane proteins are typically harder to express. However, this parameter alone would not be useful for predicting within a smaller subset, like a single protein family, where there is little variance in length (*e.g.* Fig 3,4). One might develop a predictor that was better for a given protein family under certain conditions with a subset of the entire features considered here; yet this would require *a priori* knowledge of the system, *i.e.* which sequence features were truly most important, and would preclude broad generalizability as shown for the predictor presented here.

Sequence optimization for expression

The predictive performance of the model implies that the features defined here provide a coarse approximation of the fitness landscape for membrane protein expression. Attempting to optimize a single feature by modifying the sequence will likely affect the resulting score and expression due to changes in other features. Fluman, *et al*. provides an illustrative experiment [43]. They hypothesized that altering the number of Shine-Dalgarno (SD)-like sites in the coding sequence of a membrane protein would affect expression. To test this, silent mutations were engineered within the first 200 bases of three proteins (genes *ygdD*, *brnQ*, and *ybjJ* from *E. coli*) to increase the number of SD-like sites with the goal of improving expression. Expression trials demonstrated that only one of the proteins (BrnQ) had improved expression of folded protein (Fig 7). However, the resulting changes in the IMProve score correspond with the changes in measured expression as the model considers changes to other nucleotide features. Capture of the outcomes in this small test case by the model illustrates the utility of integrating the contribution of the numerous parameters involved in membrane protein biogenesis.

Fig 7. Synonymous mutations affect expression. Relative difference in SD-like sites (green), expression (purple), and IMProve score (yellow) between wild-type and mutants with silent mutations engineered to increase anti-SD sequence binding propensity [43]. See Methods 7 for further detail.

Discussion

Here, we have demonstrated the ability to predict membrane protein expression using computational methods, a feat some have considered impossible. Our success is built on encompassing a multitude of experimental results into a single computational model. The predictive power of IMProve provides a low barrier-to-entry method to enrich for positive expression outcomes.

The current best practice for characterization of a membrane protein target begins with the identification and testing of many homologs or variants for expression. IMProve will allow for prioritization of targets to test for expression thereby making more optimal use of limited human and material resources. In addition, due to the scale of NYCOMPS, protein families that were extensively tested provide ranges of scores (*e.g.* Fig 5C) where the score of an individual target directly indicates its likelihood of expression relative to known experimental results. We provide the current predictor as web service where scores can be calculated, and the method, associated data, and suggested analyses are publically available to catalyze progress across the community (clemonslab.caltech.edu).

Having shown that membrane protein expression can be predicted, the generalizability of the model is remarkable despite several known limitations. Using data from a single study for training precludes including certain variables that empirically influence expression such as the features corresponding to fusion tags and the context of the protein in an expression plasmid, *e.g.* the 5' untranslated region, for which there was no variation in the Daley, Rapp, *et al.* dataset. Moreover, using a simple proof-of-concept linear model allowed for a straightforward and robust predictor; however, intrinsically it cannot be directly related to the biological underpinnings. While we can extract some biological inference, a linear combination of sequence features does not explicitly reflect the reality of physical limits for host cells. To some extent, constraint information is likely encoded in the complex architecture of the underlying sequence space (*e.g.* through the genetic code, TM prediction, RNA secondary structure analyses). Future statistical models that improve on these limitations will likely hone predictive power and more intricately characterize the interplay of variables that underlie membrane protein expression in *E. coli* and other systems.

A perhaps surprising outcome of our results is the demonstration of the quantitatively important contribution of the nucleotide sequence as a component of the IMProve score. This echoes the growing literature that aspects of the nucleotide sequence are important determinants of protein biogenesis in general [36,43,64–66]. While one expects that there may be different weights for various nucleotide derived features between soluble and membrane proteins, it is likely that these features are important for soluble proteins as well. An example of this is the importance of codon optimization for soluble protein expression, which has failed to show any general benefit for membrane proteins [9]. Current expression predictors that have predictive power for soluble proteins have only used protein sequence for deriving the underlying feature set [22,35]. Future prediction methods will likely benefit from including nucleotide sequence features as done here.

The ability to predict phenotypic results using sequence based statistical models opens a variety of opportunities. As done here, this requires a careful understanding of the system and its underlying biological processes enumerated in a multitude of individual variables that impact the stated goal of the predictor, in this case enriching protein expression. As new features related to expression are discovered, future work will incorporate these leading to improved models. Based on these results, expanding to new expression hosts such as eukaryotes seems entirely feasible, although a number of new features may need to be considered, *e.g.* glycosylation sites and trafficking signals. Moreover, the ability to score proteins for expressibility creates new avenues to computationally engineer membrane proteins for expression. The proof-of-concept described here required significant work to compile data from genomics consortia and the literature in a readily useable form. As data becomes more easily accessible, broadly leveraging diverse experimental outcomes to decode sequence-level information, an extension of this work, is anticipated.

Methods

Sequence mapping & retrieval and feature calculation was performed in Python 2.7 [67] using BioPython [68] and NumPy [69]; executed and consolidated using Bash (shell) scripts; and parallelized where possible using GNU Parallel [70]. Data analysis and presentation was done in R [71] within RStudio [72] using magrittr [73], plyr [74], dplyr [75], asbio [76], and datamart [77] for data handling; ggplot2 [78], ggbeeswarm [79], GGally [80], gridExtra [81], cowplot [82], scales [83], viridis [84], and RColorBrewer [85,86] for plotting; multidplyr [87] with parallel [71] and foreach [88] with iterators [89] and doMC [90]/doParallel [91] for parallel processing; and roxygen2 [92] for code organization and documentation as well as other packages as referenced.

1. Collection of data necessary for learning and evaluation

***E. coli* Sequence Data** – The nucleotide sequences from [16] were deduced by reconstructing forward and reverse primers (*i.e.* ~20 nucleotide stretches) from each gene in Colibri (based on EcoGene 11), the original source cited and later verified these primers against an archival spreadsheet provided directly by Daniel Daley (personal communication). To account for sequence and annotation corrections made to the genome after Daley, Rapp, *et al.*'s work, these primers were directly used to reconstruct the amplified product from the most recent release of the *E. coli* K-12 substr. MG1655 genome [93] (EcoGene 3.0; U00096.3). Although Daniel Daley mentioned that raw reads from the Sanger sequencing runs may be available within his own archives, it was decided that the additional labor to retrieve this data and parse these reads would not significantly impact the model. The deduced nucleotide sequences were verified against the protein lengths given in S1 Table from [16]. The plasmid library tested in [43] was provided by Daniel Daley, and those sequences are taken to be the same.

***E. coli* Training Data** – The preliminary results using the mean-normalized activities echoed the findings of [16] that these do not correlate with sequence features either in the univariate sense (many simple linear regressions, S1 Table [16]) or a multivariate sense (multiple linear regression, data not shown). This is presumably due to the loss of information regarding variability in expression level for given genes or due to the increase in variance of the normalized quantity (See Methods 4a) due to the normalization and averaging procedure. Daniel Daley and Mikaela Rapp provided spreadsheets of the outcomes from the 96-well plates used for their expression trials and sent scanned copies of the readouts from archival laboratory notebooks where the digital data was no longer accessible (personal communication). Those proteins without a reliable C-terminal localization (as given in the original work) or without raw expression outcomes were not included in further analyses.

Similarly, Nir Fluman also provided spreadsheets of the raw data from the set of three expression trials performed in [43].

New York Consortium on Membrane Protein Structure (NYCOMPS) Data – Brian Kloss, Marco Punta, and Edda Kloppman provided a dataset of actions performed by the NYCOMPS center including expression outcomes in various conditions [2,3]. The protein sequences were mapped to NCBI GenInfo Identifier (GI) numbers either via the Entrez system [94] or the Uniprot mapping service[95]. Each GI number was mapped to its nucleotide sequence via a combination of the NCBI Elink mapping service and the "coded_by" or "locus" tags of Coding Sequence (CDS) features within GenBank entries. Though a custom script was created, a script from Peter Cock on the BioPython listserv to do the same task via a similar mapping mechanism was found [96]. To confirm all the sequences, the TargetTrack [18] XML file was parsed for the internal NYCOMPS identifiers and compared for sequence identity to those that had been mapped using the custom script; 20 (less than 1%) of the sequences had minor inconsistencies and were manually replaced.

Archaeal transporters Data – The locus tags ("Gene Name" in Table 1) were mapped directly to the sequences and retrieved from NCBI [49]. Pikyee Ma and Margarida Archer clarified questions regarding their work to inform the analysis.

GPCR Expression Data – Nucleotide sequences were collected by mapping the protein identifiers given in Table 1 from [51] to protein GIs via the Uniprot mapping service [95] and subsequently to their nucleotide sequences via the custom mapping script described above (see NYCOMPS). The sequence length and pI were validated against those provided. Renaud Wagner assisted in providing the nucleotide sequences for genes whose listed identifiers were unable to be mapped and/or did not pass the validation criteria as the MeProtDB (the sponsor of the GPCR project) does not provide a public archive.

***Helicobacter pylori* Data –** Nucleotide sequences were retrieved by mapping the locus tags given in Supplemental Table 1 from [52] to locus tags in the Jan 31, 2014 release of the *H. pylori* 26695 genome (AE000511.1). To verify sequence accuracy, sequences whose molecular weight matched that given by the authors were accepted. Those that did not match, in addition to the one locus tag that could not be mapped to the Jan 31, 2014 genome version, were retrieved from the Apr 9, 2015 release of the genome (NC_000915.1). Both releases are derived from the original sequencing project [97]. After this curation, all mapped sequences matched the reported molecular weight.

In this data set, expression tests were performed in three expression vectors and scored as 1, 2, or 3. Two vectors were scored via two methods. For these two vectors, the two scores were averaged to give a single number for the condition making them comparable to the third vector while yielding 2 additional thresholds (1.5 and 2.5) result in the 5 total curves shown (S1 Fig. B).

***Mycobacterium tuberculosis* Data** – The authors note using TubercuList through GenoList [98], therefore, nucleotide sequences were retrieved from the archival website based on the original sequencing project [99]. The sequences corresponding to the identifiers and outcomes in Table 1 from [50] were validated against the provided molecular weight .

***Secondary Transporter* Data** – GI Numbers given in Table 1 from [54] were matched to their CDS entries using the custom mapping script described above (see NYCOMPS). Only expression in *E. coli* with IPTG-inducible vectors was considered.

***Thermotoga maratima* Data** – Gene names given in Table 1 [100] were matched to CDS entries in the Jan 31, 2014 release of the *Thermotoga maritima* MSB8 genome (AE000512.1), a revised annotation of the original release[101]. The sequence length and molecular weight were validated against those provided.

2. Calculation of sequence features

Based on experimental analyses and anecdotal evidence, approximately 105 different protein and nucleotide sequence features thought to be relevant to expression were identified and calculated for each protein using custom code together with published software (codonW [102], tAI [103], NUPACK [104], Vienna RNA [105], Codon Pair Bias [106], Disembl [42], and RONN [107]). Relative metrics (*e.g.*

codon adaptation index) are calculated with respect to the *E. coli* K-12 substr. MG1655 [93] quantity. The octanol-water partitioning [39], GES hydrophobicity [38], ΔG of insertion [40] scales were employed as well. Transmembrane segment topology was predicted using Phobius Constrained for the training data and Phobius for all other datasets [48]. We were able to obtain the Phobius code and integrate it directly into our feature calculation pipeline resulting in significantly faster speeds than any other option. Two RNA secondary structure metrics were prompted in part by Goodman, et al. [36]. Several features were obtained by averaging per-site metrics (*e.g.* per-residue RONN3.2 disorder predictions) in windows of a specified length. Windowed tAI metrics are calculated over *all* 30 base windows (not solely over 10 codon windows). S1 Table lists a description of each feature. Features are calculated solely from a gene of interest excluding portions of the ORFs such as linkers and tags derived from the plasmid backbone employed (future work will explore contributions of these elements).

3. Preparation for model learning

Calculated sequence features for the membrane proteins in the *E. coli* dataset as well as raw activity measurements, *i.e.* each 96-well plate, were loaded into R. As is best practice in using Support Vector Machines, each feature was "centered" and "scaled" where the mean value of a given feature was subtracted from each data point and then divided by the standard deviation of that feature using `preprocess` [108]. As is standard practice, the resulting set was then culled for those features of near zero-variance, over 95% correlation (Pearson's *r*), and linear dependence (`nearZeroVar`, `findCorrelation`, `findLinearCombos`)[108]. In particular this procedure removed extraneous degrees of freedom during the training process which carry little to no additional information with respect to the feature space and which may over represent certain redundant features. Features and outcomes for each list ("query") were written into the SVM^{light} format using a modified `svmlight.write` [109].

The final features were calculated for each sequence in the test datasets, prepared for scoring by "centering" and "scaling" by the training set parameters via `preprocess` [108], and then written into SVM^{light} format again using a modified `svmlight.write`.

4. Model selection, training, and evaluation using SVM^{rank}

a. At the most basic level, our predictive model is a learned function that maps the parameter space (consisting of nucleotide and protein sequence features) to a response variable (expression level) through a set of governing weights (w_1, w_2, ..., w_N). Depending on how the response variable is defined, these weights can be approximated using several different methods. As such, defining a response variable that is reflective of the available training data is key to selecting an appropriate learning algorithm.

The quantitative 96-well plate results [16] that comprise our training data do not offer an absolute expression metric valid over all plates—the top expressing proteins in one plate would not necessarily be the best expressing within another. As such, this problem is suited for preference-ranking methods. As a ranking problem, the response variable is the ordinal rank for each protein derived from its overexpression relative to the other members of the same plate of expression trials. In other words, the aim is to rank highly expressed proteins (based on numerous trials) at higher scores than lower expressed proteins by fitting against the <u>order</u> of expression outcomes from each constituent 96-well plate.

b. As the first work of this kind, the aim was to employ the simplest framework necessary taking in account the considerations above. The method chosen computes all valid pairwise classifications (*i.e.* within a single plate) transforming the original ranking problem into a binary classification problem.

The algorithm outputs a score for each input by minimizing the number of swapped pairs thereby maximizing Kendall's τ [110]. For example, consider the following data generated via context A $(X_{A,1}, Y_{A,1}), (X_{A,2}, Y_{A,2})$ and B $(X_{B,1}, Y_{B,1}), (X_{B,2}, Y_{B,2})$ where observed response follows as index i, *i.e.* $Y_n < Y_{n+1}$. Binary classifier $f\,(X_i, X_j)$ gives a score of 1 if an input pair matches its ordering criteria and -1 if not, *i.e.* $Y_i < Y_j$:

$$f(X_{A,1}, X_{A,2}) = 1; f(X_{A,2}, X_{A,1}) = -1$$
$$f(X_{B,1}, X_{B,2}) = 1; f(X_{B,2}, X_{B,1}) = -1$$
$$f(X_{A,1}, X_{B,2}), f(X_{A,2}, X_{B,1}) \text{ are invalid}$$

Free parameters describing f are calculated such that those calculated orderings $f(X_{A,1}), f(X_{A,2}) \ldots;\ f(X_{B,1}), f(X_{B,2}) \ldots$ most closely agree (overall Kendall's τ) with the observed ordering $Y_n, Y_{n+1}, \ldots$. In this sense, f is a pairwise Learning to Rank method.

Within this class of models, a linear preference-ranking Support Vector Machine was employed [111]. To be clear, as an algorithm a preference-ranking SVM operates similarly to the canonical SVM binary classifier. In the traditional binary classification problem, a linear SVM seeks the maximally separating hyper-plane in the feature space between two classes, where class membership is determined by which side of the hyper-plane points reside. For some n linear separable training examples $D = \{\,(x_i) \mid x_i \in \mathbb{R}^d\}^n$ and two classes $y_i \in \{-1, 1\}$, a linear SVM seeks a mapping from the d-dimensional feature space $\mathbb{R}^d \rightarrow \{-1, 1\}$ by finding two maximally separated hyperplanes $w \cdot x - b = 1$ and $w \cdot x - b = -1$ with constraints that $w \cdot x_i - b \geq 1$ for all x_i with $y_i \in \{1\}$ and $w \cdot x_i - b \leq -1$ for all x_i with $y_i \in \{-1\}$. The feature weights correspond to the vector *w*, which is the vector perpendicular to the separating hyperplanes, and are computable in $O(n \log n)$ implemented as part of the SVM^{rank} software package, though in $O(n^2)$ [46]. See [111] for an in-depth, technical discussion.

c. In a soft-margin SVM where training data is not linearly separable, a tradeoff between misclassified inputs and separation from the hyperplane must be specified. This parameter C was found by training models against raw data from Daley, Rapp, *et al*. with a grid of candidate C values ($2^n\ \forall\, n \in [-5,\ 5]$) and then evaluated against the raw "folded protein" measurements from Fluman, *et al*. The final model was chosen by selecting that with the lowest error from the process above ($C = 2^5$). To be clear, the final model is composed solely of a single weight for each feature; the tradeoff parameter C is only part of the training process.

Qualitatively, such a preference-ranking method constructs a model that ranks groups of proteins with higher expression level higher than other groups with lower expression value. In comparison to methods such as linear regression and binary classification, this approach is more robust and less affected by the inherent stochasticity of the training data.

5. Quantitative Assessment of Predictive Performance

In generating a predictive model, one aims to enrich for positive outcomes while ensuring they do not come at the cost of increased false positive diagnoses. This is formalized in Receiver Operating Characteristic (ROC) theory (for a primer see [47]), where the true positive rate is plotted against the false positive rate for all classification thresholds (score cutoffs in the ranked list). In this framework, the overall ability of the model to resolve positive from negative outcomes is evaluated by analyzing the Area Under a ROC curve (AUC) where $AUC_{perfect}$=100% and AUC_{random}=50% (percentage signs are omitted throughout the text and figures). All ROCs are calculated through pROC [112] using the analytic Delong method for AUC confidence intervals [113]. Bootstrapped AUC CIs ($N = 10^6$) were precise to 4 decimal places suggesting that analytic CIs are valid for the NYCOMPS dataset.

With several of our datasets, no definitive standard or clear-cut classification for positive expression exists. However, the aim is to show and test all reasonable classification thresholds of positive expression for each dataset in order to evaluate predictive performance as follows:

Training data – The outcomes are quantitative (activity level), so each ROC is calculated by normalizing within each dataset to the standard well subject to the discussion in 4a above (LepB for PhoA, and InvLepB for GFP) (examples in Fig 1D) for each possible threshold, *i.e.* each normalized expression value with each AUC plotted in Fig 1E. 95% confidence intervals of Spearman's ρ are given by 10^6 iterations of a bias-corrected and accelerated (BCa) bootstrap of the data (Fig 1A,C) [45].

Large-scale – ROCs were calculated for each of the expression classes (Fig 2E). Regardless of the split, predictive performance is noted. The binwidth for the histogram was determined using the Freedman-Diaconis rule[114], and scores outside the plotted range comprising <0.6% of the density were implicitly hidden.

Small-scale – Classes can be defined in many different ways. To be principled about the matter, ROCs for each possible cutoff are presented based on definitions from each publication (Fig 3C,E,G, S1 Fig. B,D,F). See Methods 1 for any necessary details about outcome classifications for each dataset.

6. Feature Weights

Weights for the learned SVM are pulled directly from the model file produced by SVM^{light} and are given in S1 Table.

7. Forward Predictions

Data collection – We selected several genomes for comparison as shown in Fig 5, S2 Fig. A, and S4 Table. Coding sequences of membrane proteins from human and mouse genomes were gathered by mapping Uniprot identifiers of proteins noted to have at least one transmembrane segment by Uniprot [95] to Ensembl (release 82) coding sequences [115] via Biomart [116]. *C. elegans* coding sequences were similarly mapped via Uniprot but to WormBase coding sequences [117] also via Biomart. *S. cerevisiae* strain S288C coding sequences [118] were retrieved from the Saccharomyces Genome Database. *P. pastoris* strain GS115 coding sequences [119] were retrieved from the DOE Joint Genome Institute (JGI) Genome Portal [120]. Those sequences without predicted [48] TMs were excluded from subsequent analyses. Microbial sequences were gathered via a custom, in-house database populated with data compiled primarily from Pfam [60], DOE JGI Integrated Microbial Genomes [121], and the Microbial Genome Database [122].

Feature calculation – Because of the incredible number of sequences, we did not calculate the features derived from the most computationally expensive calculation (whole sequence mRNA pairing probability). Since predictive performance on the NYCOMPS dataset is slightly smaller, but not significantly different at 95% confidence, in the absence of these features (S2 Table), the forward predictions are still valid. For future experiments, these features can be calculated for the subset of targets of interest.

Parameter space similarity – As a first approximation of the similarity of the ~90 dimensional sequence parameter space between two groupings, features were compared pairwise via the following metric. Let f_i and g_i represent the true distributions for a given feature i between two groups of interest. The distribution overlap, *i.e.* shared area, Δ_i is formalized as

$$\Delta_i(f_i, g_i) = \int \min\{f_i(x), g_i(x)\}\, dx$$

ranging from 0, for entirely distinct distributions, to 1 for entirely identical distributions.

As written f_i and g_i are probability densities, they need to be approximated before calculating Δ_i and are done so via kernel density estimates (KDE) of the observed samples $\left[x_1^f, \ldots, x_n^f\right]$ and $\left[x_1^g, \ldots, x_n^g\right]$ using a nonparametric, locally adaptive method allowing for variable bandwidth smoothing implemented in LocFit[123] (`adpen`=$2\sigma^2$) providing $\hat{f}_i$ and $\hat{g}_i$. The distribution overlap Δ_i is evaluated over a grid of 2^{13} equally spaced points over the range of f_i and g_i.

Shine-Dalgarno-like mutagenesis – Folded protein is quantified by densitometry measurement [124,125] of the relevant band in Figure 6 of [43]. Relative difference is calculated as is standard:

$$\frac{\text{metric}_{\text{mutant}} - \text{metric}_{\text{wildtype}}}{\frac{1}{2}\left|\text{metric}_{\text{mutant}} - \text{metric}_{\text{wildtype}}\right|}$$

8. Availability

All analysis is documented in a series of R notebooks[126] available openly at github.com/clemlab/IMProve. These notebooks provide fully executable instructions for the reproduction of the analyses and the generation of figures and statistics in this study. The ranking engine is available as a web service at clemonslab.caltech.edu. Additional code is available upon request.

Acknowledgements

We thank Daniel Daley and Thomas Miller's group for discussion, Yaser Abu-Mostafa and Yisong Yue for guidance regarding machine learning, Niles Pierce for providing NUPACK source code [104], Welison Floriano and Naveed Near-Ansari for maintaining local computing resources, and Samuel Schulte for suggesting the model's name. We thank James Bowie, Michiel Niesen, Stephen Marshall, Thomas Miller, Reid van Lehn, and Tom Rapoport for critical reading of the manuscript. Models and analyses are possible thanks to raw experimental data provided by Daniel Daley and Mikaela Rapp [16]; Nir Fluman [43]; Edda Kloppmann, Brian Kloss, and Marco Punta from NYCOMPS [2,3]; Pikyee Ma [49]; Renaud Wagner [51]; and Florent Bernaudat [55].

Computational time was provided by Stephen Mayo and Douglas Rees. This work used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1053575 [127].

References

1. Hendrickson WA. Atomic-level analysis of membrane-protein structure. Nat Struct Mol Biol. 2016;23: 464–467. doi:10.1038/nsmb.3215

2. Punta M, Love J, Handelman S, Hunt JF, Shapiro L, Hendrickson WA, et al. Structural genomics target selection for the New York consortium on membrane protein structure. J Struct Funct Genomics. 2009;10: 255–268. doi:10.1007/s10969-009-9071-1

3. Love J, Mancia F, Shapiro L, Punta M, Rost B, Girvin M, et al. The New York Consortium on Membrane Protein Structure (NYCOMPS): a high-throughput platform for structural genomics of integral membrane proteins. J Struct Funct Genomics. 2010;11: 191–199. doi:10.1007/s10969-010-9094-7

4. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, et al. The Protein Data Bank. Nucleic Acids Res. 2000;28: 235–242.

5. Lewinson O, Lee AT, Rees DC. The funnel approach to the precrystallization production of membrane proteins. J Mol Biol. 2008;377: 62–73. doi:10.1016/j.jmb.2007.12.059

6. Merk A, Bartesaghi A, Banerjee S, Falconieri V, Rao P, Davis MI, et al. Breaking Cryo-EM Resolution Barriers to Facilitate Drug Discovery. Cell. 2016;165: 1698–1707. doi:10.1016/j.cell.2016.05.040

7. Nannenga BL, Gonen T. MicroED opens a new era for biological structure determination. Curr Opin Struct Biol. 2016;40: 128–135. doi:10.1016/j.sbi.2016.09.007

8. Bill RM, Henderson PJF, Iwata S, Kunji ERS, Michel H, Neutze R, et al. Overcoming barriers to membrane protein structure determination. Nat Biotechnol. 2011;29: 335–340. doi:10.1038/nbt.1833

9. Nørholm MHH, Light S, Virkki MTI, Elofsson A, von Heijne G, Daley DO. Manipulating the genetic code for membrane protein production: what have we learnt so far? Biochim Biophys Acta. 2012;1818: 1091–1096. doi:10.1016/j.bbamem.2011.08.018

10. Wagner S, Klepsch MM, Schlegel S, Appel A, Draheim R, Tarry M, et al. Tuning Escherichia coli for membrane protein overexpression. Proc Natl Acad Sci U S A. 2008;105: 14371–14376. doi:10.1073/pnas.0804090105

11. Marshall SS, Niesen MJM, Müller A, Tiemann K, Saladi SM, Galimidi RP, et al. A Link between Integral Membrane Protein Expression and Simulated Integration Efficiency. Cell Rep. 2016;16: 2169–2177. doi:10.1016/j.celrep.2016.07.042

12. Sarkar CA, Dodevski I, Kenig M, Dudli S, Mohr A, Hermans E, et al. Directed evolution of a G protein-coupled receptor for expression, stability, and binding selectivity. Proc Natl Acad Sci U S A. 2008;105: 14808–14813. doi:10.1073/pnas.0803103105

13. Schlinkmann KM, Honegger A, Türeci E, Robison KE, Lipovšek D, Plückthun A. Critical features for biosynthesis, stability, and functionality of a G protein-coupled receptor uncovered by all-versus-all mutations. Proc Natl Acad Sci U S A. 2012;109: 9810–9815. doi:10.1073/pnas.1202107109

14. Seppälä S, Slusky JS, Lloris-Garcerá P, Rapp M, von Heijne G. Control of membrane protein topology by a single C-terminal residue. Science. 2010;328: 1698–1700. doi:10.1126/science.1188950

15. Van Lehn RC, Zhang B, Miller TF. Regulation of multispanning membrane protein topology via post-translational annealing. eLife. 2015;4. doi:10.7554/eLife.08697

16. Daley DO, Rapp M, Granseth E, Melén K, Drew D, von Heijne G. Global topology analysis of the Escherichia coli inner membrane proteome. Science. 2005;308: 1321–1323. doi:10.1126/science.1109730

17. Nørholm MHH, Toddo S, Virkki MTI, Light S, von Heijne G, Daley DO. Improved production of membrane proteins in Escherichia coli by selective codon substitutions. FEBS Lett. 2013;587: 2352–2358. doi:10.1016/j.febslet.2013.05.063

18. Chen L, Oughtred R, Berman HM, Westbrook J. TargetDB: a target registration database for structural genomics projects. Bioinformatics. 2004;20: 2860–2862. doi:10.1093/bioinformatics/bth300

19. Gabanyi MJ, Adams PD, Arnold K, Bordoli L, Carter LG, Flippen-Andersen J, et al. The Structural Biology Knowledgebase: a portal to protein structures, sequences, functions, and methods. J Struct Funct Genomics. 2011;12: 45–54. doi:10.1007/s10969-011-9106-2

20. Bertone P, Kluger Y, Lan N, Zheng D, Christendat D, Yee A, et al. SPINE: an integrated tracking database and data mining approach for identifying feasible targets in high-throughput structural proteomics. Nucleic Acids Res. 2001;29: 2884–2898.

21. Slabinski L, Jaroszewski L, Rodrigues APC, Rychlewski L, Wilson IA, Lesley SA, et al. The challenge of protein structure determination--lessons from structural genomics. Protein Sci Publ Protein Soc. 2007;16: 2472–2482. doi:10.1110/ps.073037907

22. Slabinski L, Jaroszewski L, Rychlewski L, Wilson IA, Lesley SA, Godzik A. XtalPred: a web server for prediction of protein crystallizability. Bioinforma Oxf Engl. 2007;23: 3403–3405. doi:10.1093/bioinformatics/btm477

23. Jahandideh S, Jaroszewski L, Godzik A. Improving the chances of successful protein structure determination with a random forest classifier. Acta Crystallogr D Biol Crystallogr. 2014;70: 627–635. doi:10.1107/S1399004713032070

24. Price WN, Chen Y, Handelman SK, Neely H, Manor P, Karlin R, et al. Understanding the physical properties that control protein crystallization by analysis of large-scale experimental data. Nat Biotechnol. 2009;27: 51–57. doi:10.1038/nbt.1514

25. Overton IM, Padovani G, Girolami MA, Barton GJ. ParCrys: a Parzen window density estimation approach to protein crystallization propensity prediction. Bioinforma Oxf Engl. 2008;24: 901–907. doi:10.1093/bioinformatics/btn055

26. Mizianty MJ, Kurgan L. Meta prediction of protein crystallization propensity. Biochem Biophys Res Commun. 2009;390: 10–15. doi:10.1016/j.bbrc.2009.09.036

27. Babnigg G, Joachimiak A. Predicting protein crystallization propensity from protein sequence. J Struct Funct Genomics. 2010;11: 71–80. doi:10.1007/s10969-010-9080-0

28. Kandaswamy KK, Pugalenthi G, Suganthan PN, Gangal R. SVMCRYS: an SVM approach for the prediction of protein crystallization propensity from protein sequence. Protein Pept Lett. 2010;17: 423–430.

29. Mizianty MJ, Kurgan L. Sequence-based prediction of protein crystallization, purification and production propensity. Bioinforma Oxf Engl. 2011;27: i24-33. doi:10.1093/bioinformatics/btr229

30. Jahandideh S, Mahdavi A. RFCRYS: sequence-based protein crystallization propensity prediction by means of random forest. J Theor Biol. 2012;306: 115–119. doi:10.1016/j.jtbi.2012.04.028

31. Mizianty MJ, Kurgan LA. CRYSpred: accurate sequence-based protein crystallization propensity prediction using sequence-derived structural characteristics. Protein Pept Lett. 2012;19: 40–49.

32. Fusco D, Barnum TJ, Bruno AE, Luft JR, Snell EH, Mukherjee S, et al. Statistical analysis of crystallization database links protein physico-chemical features with crystallization mechanisms. PloS One. 2014;9: e101123. doi:10.1371/journal.pone.0101123

33. Gao J, Hu G, Wu Z, Ruan J, Shen S, Hanlon M, et al. Improved Prediction of Protein Crystallization, Purification and Production Propensity Using Hybrid Sequence Representation. Curr Bioinforma. 2014;9: 57–64. doi:10.2174/15748936113080990006

34. Hu J, Han K, Li Y, Yang J-Y, Shen H-B, Yu D-J. TargetCrys: protein crystallization prediction by fusing multi-view features with two-layered SVM. Amino Acids. 2016;48: 2533–2547. doi:10.1007/s00726-016-2274-4

35. Wang H, Feng L, Zhang Z, Webb GI, Lin D, Song J. Crysalis: an integrated server for computational analysis and design of protein crystallization. Sci Rep. 2016;6: 21383. doi:10.1038/srep21383

36. Goodman DB, Church GM, Kosuri S. Causes and effects of N-terminal codon bias in bacterial genes. Science. 2013;342: 475–479. doi:10.1126/science.1241934

37. Mirzadeh K, Martínez V, Toddo S, Guntur S, Herrgård MJ, Elofsson A, et al. Enhanced Protein Production in Escherichia coli by Optimization of Cloning Scars at the Vector-Coding Sequence Junction. ACS Synth Biol. 2015; doi:10.1021/acssynbio.5b00033

38. Engelman DM, Steitz TA, Goldman A. Identifying nonpolar transbilayer helices in amino acid sequences of membrane proteins. Annu Rev Biophys Biophys Chem. 1986;15: 321–353. doi:10.1146/annurev.bb.15.060186.001541

39. Wimley WC, Creamer TP, White SH. Solvation energies of amino acid side chains and backbone in a family of host-guest pentapeptides. Biochemistry (Mosc). 1996;35: 5109–5124. doi:10.1021/bi9600153

40. Hessa T, Meindl-Beinker NM, Bernsel A, Kim H, Sato Y, Lerch-Bader M, et al. Molecular code for transmembrane-helix recognition by the Sec61 translocon. Nature. 2007;450: 1026–1030. doi:10.1038/nature06387

41. Heijne G. The distribution of positively charged residues in bacterial inner membrane proteins correlates with the trans-membrane topology. EMBO J. 1986;5: 3021–3027.

42. Linding R, Jensen LJ, Diella F, Bork P, Gibson TJ, Russell RB. Protein disorder prediction: implications for structural proteomics. Structure. 2003;11: 1453–1459.

43. Fluman N, Navon S, Bibi E, Pilpel Y. mRNA-programmed translation pauses in the targeting of E. coli membrane proteins. eLife. 2014;3. doi:10.7554/eLife.03440

44. Geertsma ER, Groeneveld M, Slotboom D-J, Poolman B. Quality control of overexpressed membrane proteins. Proc Natl Acad Sci U S A. 2008;105: 5722–5727. doi:10.1073/pnas.0802190105

45. Canty A, Ripley BD. boot: Bootstrap R (S-Plus) Functions. 2015.

46. Tsochantaridis I, Joachims T, Hofmann T, Altun Y. Large Margin Methods for Structured and Interdependent Output Variables. J Mach Learn Res. 2005;6: 1453–1484.

47. Swets JA, Dawes RM, Monahan J. Better decisions through science. Sci Am. 2000;283: 82–87.

48. Käll L, Krogh A, Sonnhammer ELL. A combined transmembrane topology and signal peptide prediction method. J Mol Biol. 2004;338: 1027–1036. doi:10.1016/j.jmb.2004.03.016

49. Ma P, Varela F, Magoch M, Silva AR, Rosário AL, Brito J, et al. An efficient strategy for small-scale screening and production of archaeal membrane transport proteins in Escherichia coli. PloS One. 2013;8: e76913. doi:10.1371/journal.pone.0076913

50. Korepanova A, Gao FP, Hua Y, Qin H, Nakamoto RK, Cross TA. Cloning and expression of multiple integral membrane proteins from Mycobacterium tuberculosis in Escherichia coli. Protein Sci. 2005;14: 148–158. doi:10.1110/ps.041022305

51. Lundstrom K, Wagner R, Reinhart C, Desmyter A, Cherouati N, Magnin T, et al. Structural genomics on membrane proteins: comparison of more than 100 GPCRs in 3 expression systems. J Struct Funct Genomics. 2006;7: 77–91. doi:10.1007/s10969-006-9011-2

52. Psakis G, Nitschkowski S, Holz C, Kress D, Maestre-Reyna M, Polaczek J, et al. Expression screening of integral membrane proteins from Helicobacter pylori 26695. Protein Sci. 2007;16: 2667–2676. doi:10.1110/ps.073104707

53. Dobrovetsky E, Lu ML, Andorn-Broza R, Khutoreskaya G, Bray JE, Savchenko A, et al. High-throughput production of prokaryotic membrane proteins. J Struct Funct Genomics. 2005;6: 33–50. doi:10.1007/s10969-005-1363-5

54. Surade S, Klein M, Stolt-Bergner PC, Muenke C, Roy A, Michel H. Comparative analysis and "expression space" coverage of the production of prokaryotic membrane proteins for structural genomics. Protein Sci. 2006;15: 2178–2189. doi:10.1110/ps.062312706

55. Bernaudat F, Frelet-Barrand A, Pochon N, Dementin S, Hivin P, Boutigny S, et al. Heterologous expression of membrane proteins: choosing the appropriate host. PloS One. 2011;6: e29191. doi:10.1371/journal.pone.0029191

56. Eshaghi S, Hedrén M, Nasser MIA, Hammarberg T, Thornell A, Nordlund P. An efficient strategy for high-throughput expression screening of recombinant integral membrane proteins. Protein Sci. 2005;14: 676–683. doi:10.1110/ps.041127005

57. Gordon E, Horsefield R, Swarts HGP, de Pont JJHHM, Neutze R, Snijder A. Effective high-throughput overproduction of membrane proteins in Escherichia coli. Protein Expr Purif. 2008;62: 1–8. doi:10.1016/j.pep.2008.07.005

58. Petrovskaya LE, Shulga AA, Bocharova OV, Ermolyuk YS, Kryukova EA, Chupin VV, et al. Expression of G-protein coupled receptors in Escherichia coli for structural studies. Biochem Mosc. 2010;75: 881–891.

59. Szakonyi G, Leng D, Ma P, Bettaney KE, Saidijam M, Ward A, et al. A genomic strategy for cloning, expressing and purifying efflux proteins of the major facilitator superfamily. J Antimicrob Chemother. 2007;59: 1265–1270. doi:10.1093/jac/dkm036

60. Finn RD, Bateman A, Clements J, Coggill P, Eberhardt RY, Eddy SR, et al. Pfam: the protein families database. Nucleic Acids Res. 2014;42: D222-230. doi:10.1093/nar/gkt1223

61. Saier MH, Reddy VS, Tsu BV, Ahmed MS, Li C, Moreno-Hagelsieb G. The Transporter Classification Database (TCDB): recent advances. Nucleic Acids Res. 2016;44: D372-379. doi:10.1093/nar/gkv1103

62. Tukey JW. Exploratory data analysis. Reading, Mass: Addison-Wesley Pub. Co; 1977.

63. Tufte ER. The visual display of quantitative information. 2nd ed. Cheshire, Conn: Graphics Press; 2001.

64. Li G-W, Oh E, Weissman JS. The anti-Shine-Dalgarno sequence drives translational pausing and codon choice in bacteria. Nature. 2012;484: 538–541. doi:10.1038/nature10965

65. Gamble CE, Brule CE, Dean KM, Fields S, Grayhack EJ. Adjacent Codons Act in Concert to Modulate Translation Efficiency in Yeast. Cell. 2016;166: 679–690. doi:10.1016/j.cell.2016.05.070

66. Chartron JW, Hunt KCL, Frydman J. Cotranslational signal-independent SRP preloading during membrane targeting. Nature. 2016;536: 224–228. doi:10.1038/nature19309

67. Van Rossum G, Drake Jr FL. Python reference manual. Centrum voor Wiskunde en Informatica Amsterdam; 1995.

68. Cock PJA, Antao T, Chang JT, Chapman BA, Cox CJ, Dalke A, et al. Biopython: freely available Python tools for computational molecular biology and bioinformatics. Bioinformatics. 2009;25: 1422–1423. doi:10.1093/bioinformatics/btp163

69. van der Walt S, Colbert SC, Varoquaux G. The NumPy Array: A Structure for Efficient Numerical Computation. Comput Sci Eng. 2011;13: 22–30. doi:10.1109/MCSE.2011.37

70. Tange O. GNU Parallel - The Command-Line Power Tool. Login USENIX Mag. 2011;36: 42–47. doi:10.5281/zenodo.16303

71. R Core Team. R: A Language and Environment for Statistical Computing [Internet]. Vienna, Austria: R Foundation for Statistical Computing; 2015. Available: https://www.R-project.org/

72. RStudio Team. RStudio: Integrated Development Environment for R. Boston, MA: RStudio, Inc.; 2015.

73. Bache SM, Wickham H. magrittr: A Forward-Pipe Operator for R [Internet]. 2014. Available: https://CRAN.R-project.org/package=magrittr

74. Wickham H. The Split-Apply-Combine Strategy for Data Analysis. J Stat Softw. 2011;40: 1–29.

75. Wickham H, Francois R. dplyr: A Grammar of Data Manipulation [Internet]. 2015. Available: http://CRAN.R-project.org/package=dplyr

76. Aho K. asbio: A Collection of Statistical Tools for Biologists [Internet]. 2015. Available: http://CRAN.R-project.org/package=asbio

77. Weinert K. datamart: Unified access to your data sources [Internet]. 2014. Available: http://CRAN.R-project.org/package=datamart

78. Wickham H. ggplot2: elegant graphics for data analysis [Internet]. Springer New York; 2009. Available: http://had.co.nz/ggplot2/book

79. Clarke E, Sherrill-Mix S. ggbeeswarm: Categorical Scatter (Violin Point) Plots [Internet]. 2015. Available: https://github.com/eclarke/ggbeeswarm

80. Schloerke B, Crowley J, Cook D, Briatte F, Marbach M, Thoen E, et al. GGally: Extension to “ggplot2” [Internet]. 2016. Available: https://CRAN.R-project.org/package=GGally

81. Auguie B. gridExtra: Miscellaneous Functions for “Grid” Graphics [Internet]. 2015. Available: http://CRAN.R-project.org/package=gridExtra

82. Wilke CO. cowplot: Streamlined Plot Theme and Plot Annotations for “ggplot2” [Internet]. 2015. Available: http://CRAN.R-project.org/package=cowplot

83. Wickham H. scales: Scale Functions for Visualization [Internet]. 2015. Available: http://CRAN.R-project.org/package=scales

84. Garnier S. viridis: Default Color Maps from “matplotlib” [Internet]. 2016. Available: https://CRAN.R-project.org/package=viridis

85. Neuwirth E. RColorBrewer: ColorBrewer Palettes [Internet]. 2014. Available: http://CRAN.R-project.org/package=RColorBrewer

86. Harrower M, Brewer CA. ColorBrewer.org: an online tool for selecting colour schemes for maps. Cartogr J. 2003;40: 27–37.

87. Wickham H. multidplyr: Partitioned data frames for “dplyr” [Internet]. Available: https://github.com/hadley/multidplyr

88. Revolution Analytics, Weston S. foreach: Provides Foreach Looping Construct for R [Internet]. 2015. Available: http://CRAN.R-project.org/package=foreach

89. Revolution Analytics, Weston S. iterators: Provides Iterator Construct for R [Internet]. 2015. Available: https://CRAN.R-project.org/package=iterators

90. Revolution Analytics, Weston S. doMC: Foreach Parallel Adaptor for “parallel” [Internet]. 2015. Available: http://CRAN.R-project.org/package=doMC

91. Revolution Analytics, Weston S. doParallel: Foreach Parallel Adaptor for the “parallel” Package [Internet]. 2015. Available: https://CRAN.R-project.org/package=doParallel

92. Wickham H, Danenberg P, Eugster M. roxygen2: In-Source Documentation for R [Internet]. 2015. Available: https://CRAN.R-project.org/package=roxygen2

93. Zhou J, Rudd KE. EcoGene 3.0. Nucleic Acids Res. 2013;41: D613-624. doi:10.1093/nar/gks1235

94. Schuler GD, Epstein JA, Ohkawa H, Kans JA. Entrez: molecular biology database and retrieval system. Methods Enzymol. 1996;266: 141–162.

95. UniProt Consortium. Reorganizing the protein space at the Universal Protein Resource (UniProt). Nucleic Acids Res. 2012;40: D71-75. doi:10.1093/nar/gkr981

96. Cock P. [BioPython] Downloading CDS sequences [Internet]. 2009. Available: http://biopython.org/pipermail/biopython/2009-January/004886.html

97. Tomb JF, White O, Kerlavage AR, Clayton RA, Sutton GG, Fleischmann RD, et al. The complete genome sequence of the gastric pathogen Helicobacter pylori. Nature. 1997;388: 539–547. doi:10.1038/41483

98. Lechat P, Hummel L, Rousseau S, Moszer I. GenoList: an integrated environment for comparative analysis of microbial genomes. Nucleic Acids Res. 2008;36: D469-474. doi:10.1093/nar/gkm1042

99. Cole ST, Brosch R, Parkhill J, Garnier T, Churcher C, Harris D, et al. Deciphering the biology of Mycobacterium tuberculosis from the complete genome sequence. Nature. 1998;393: 537–544. doi:10.1038/31159

100. Dobrovetsky E, Lu ML, Andorn-Broza R, Khutoreskaya G, Bray JE, Savchenko A, et al. High-throughput production of prokaryotic membrane proteins. J Struct Funct Genomics. 2005;6: 33–50. doi:10.1007/s10969-005-1363-5

101. Nelson KE, Clayton RA, Gill SR, Gwinn ML, Dodson RJ, Haft DH, et al. Evidence for lateral gene transfer between Archaea and bacteria from genome sequence of Thermotoga maritima. Nature. 1999;399: 323–329. doi:10.1038/20601

102. Peden JF. Analysis of codon usage. University of Nottingham. 2000.

103. dos Reis M, Wernisch L, Savva R. Unexpected correlations between gene expression and codon usage bias from microarray data for the whole Escherichia coli K-12 genome. Nucleic Acids Res. 2003;31: 6976–6985.

104. Zadeh JN, Steenberg CD, Bois JS, Wolfe BR, Pierce MB, Khan AR, et al. NUPACK: Analysis and design of nucleic acid systems. J Comput Chem. 2011;32: 170–173. doi:10.1002/jcc.21596

105. Lorenz R, Bernhart SH, Höner Zu Siederdissen C, Tafer H, Flamm C, Stadler PF, et al. ViennaRNA Package 2.0. Algorithms Mol Biol AMB. 2011;6: 26. doi:10.1186/1748-7188-6-26

106. Coleman JR, Papamichail D, Skiena S, Futcher B, Wimmer E, Mueller S. Virus attenuation by genome-scale changes in codon pair bias. Science. 2008;320: 1784–1787. doi:10.1126/science.1155761

107. Yang ZR, Thomson R, McNeil P, Esnouf RM. RONN: the bio-basis function neural network technique applied to the detection of natively disordered regions in proteins. Bioinformatics. 2005;21: 3369–3376. doi:10.1093/bioinformatics/bti534

108. Kuhn M. Building predictive models in R using the caret package. J Stat Soft. 2008;

109. Weihs C, Ligges U, Luebke K, Raabe N. klaR Analyzing German Business Cycles. In: Baier D, Decker R, Schmidt-Thieme L, editors. Data Analysis and Decision Support. Berlin/Heidelberg: Springer-Verlag; 2005. pp. 335–343. Available: http://link.springer.com/10.1007/3-540-28397-8_36

110. Kendall MG. A New Measure of Rank Correlation. Biometrika. 1938;30: 81. doi:10.2307/2332226

111. Joachims T. Optimizing search engines using clickthrough data. ACM Press; 2002. p. 133. doi:10.1145/775047.775067

112. Robin X, Turck N, Hainard A, Tiberti N, Lisacek F, Sanchez J-C, et al. pROC: an open-source package for R and S+ to analyze and compare ROC curves. BMC Bioinformatics. 2011;12: 77. doi:10.1186/1471-2105-12-77

113. DeLong ER, DeLong DM, Clarke-Pearson DL. Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. Biometrics. 1988;44: 837–845.

114. Freedman D, Diaconis P. On the histogram as a density estimator:L 2 theory. Z F�r Wahrscheinlichkeitstheorie Verwandte Geb. 1981;57: 453–476. doi:10.1007/BF01025868

115. Cunningham F, Amode MR, Barrell D, Beal K, Billis K, Brent S, et al. Ensembl 2015. Nucleic Acids Res. 2015;43: D662-669. doi:10.1093/nar/gku1010

116. Smedley D, Haider S, Durinck S, Pandini L, Provero P, Allen J, et al. The BioMart community portal: an innovative alternative to large, centralized data repositories. Nucleic Acids Res. 2015;43: W589-598. doi:10.1093/nar/gkv350

117. Harris TW, Baran J, Bieri T, Cabunoc A, Chan J, Chen WJ, et al. WormBase 2014: new views of curated biology. Nucleic Acids Res. 2014;42: D789-793. doi:10.1093/nar/gkt1063

118. Engel SR, Dietrich FS, Fisk DG, Binkley G, Balakrishnan R, Costanzo MC, et al. The reference genome sequence of Saccharomyces cerevisiae: then and now. G3 Bethesda Md. 2014;4: 389–398. doi:10.1534/g3.113.008995

119. De Schutter K, Lin Y-C, Tiels P, Van Hecke A, Glinka S, Weber-Lehmann J, et al. Genome sequence of the recombinant protein production host Pichia pastoris. Nat Biotechnol. 2009;27: 561–566. doi:10.1038/nbt.1544

120. Nordberg H, Cantor M, Dusheyko S, Hua S, Poliakov A, Shabalov I, et al. The genome portal of the Department of Energy Joint Genome Institute: 2014 updates. Nucleic Acids Res. 2014;42: D26-31. doi:10.1093/nar/gkt1069

121. Markowitz VM, Chen I-MA, Palaniappan K, Chu K, Szeto E, Pillay M, et al. IMG 4 version of the integrated microbial genomes comparative analysis system. Nucleic Acids Res. 2014;42: D560-567. doi:10.1093/nar/gkt963

122. Uchiyama I, Mihara M, Nishide H, Chiba H. MBGD update 2015: microbial genome database for flexible ortholog analysis utilizing a diverse set of genomic data. Nucleic Acids Res. 2015;43: D270-276. doi:10.1093/nar/gku1152

123. Loader C. locfit: Local Regression, Likelihood and Density Estimation. [Internet]. 2013. Available: https://CRAN.R-project.org/package=locfit

124. Schindelin J, Arganda-Carreras I, Frise E, Kaynig V, Longair M, Pietzsch T, et al. Fiji: an open-source platform for biological-image analysis. Nat Methods. 2012;9: 676–682. doi:10.1038/nmeth.2019

125. Schneider CA, Rasband WS, Eliceiri KW. NIH Image to ImageJ: 25 years of image analysis. Nat Methods. 2012;9: 671–675.

126. Xie Y. knitr: A Comprehensive Tool for Reproducible Research in R. In: Stodden V, Leisch F, Peng RD, editors. Implementing Reproducible Computational Research. Chapman and Hall/CRC; 2014. Available: http://www.crcpress.com/product/isbn/9781466561595

127. Towns J, Cockerill T, Dahan M, Foster I, Gaither K, Grimshaw A, et al. XSEDE: Accelerating Scientific Discovery. Comput Sci Eng. 2014;16: 62–74. doi:10.1109/MCSE.2014.80

Supporting Information

S1 Fig. Additional small-scale predictions and outcomes. (A) Experimental expression of 116 *H. pylori* membrane proteins in *E. coli* in at most 3 vectors (238 trials) scored as either a 1, 2, or 3 from the outcome of a dot blot as well as Coomassie Staining of an SDS-PAGE gel for two of the vectors. To compare the three vectors with a single set of scores, the two scores were averaged to give a single number for a condition making them comparable to the third vector while yielding 2 additional thresholds (1.5 and 2.5) and the 6 total levels shown. **(B)** The Reciever Operating Characteristic (ROC) with each cutoff is plotted, where a higher cutoff is represented by a deeper red, followed by the Area Under the Curves (directly below) in colors that correspond to the respective curve. **(C)** Expression of 77 *T. maritima* membrane proteins in *E. coli* noted as purified (5), not purified but expressed (14), or neither. **(D)** ROC curve for each threshold. **(E)** Expression of 37 microbial secondary transporters in 4 IPTG-inducible vectors (144 trials) in *E. coli* quantified as 10 ng/mL (pink) or 100 ng/mL (red) via dot blot. **(F)** ROC curve for each threshold.

S2 Fig. Complete set of forward predictions. (A) Extended from Fig 5C, the full complement of score distributions calculated by genome is plotted and arranged to accentuate similar features by physiology, *e.g.* growth condition, or scientific interest, *e.g.* pathogenic. Raw scores along with sequence identifiers are available in the S4 Table. **(B)** Histograms of representative sequence features between the training data set (green), thermophiles (orange), and *P. falciparum* (purple). Values for sequence parameter overlap coefficients derived from kernel density estimates (Methods 7) versus the *E. coli* training data are included. See S1 Table for parameter descriptions.

S3 Fig. Complete set of feature correlations and their individual contributions to the model. Features are ordered first by category (as in Fig 5) and then by weight (grey bars). Labels are green for protein-sequence derived and brown for nucleotide-sequence derived features. Pearson correlation coefficient between each pair of features across the NYCOMPS dataset is plotted (right). See S1 Table for a detailed description of each feature. Feature categories are overlaid as square boxes and indicated by black bars on the top, left, and right of the correlation matrix.

S4 Fig. Feature contributions to the model across datasets used for training and validation. (A) Total weight for each category is represented as a bar. The contribution of each feature to the category is shown by partitioning the bar. The red dot indicates the total sum of weights within the category. **(B)** Pearson correlation coefficients between feature categories are shown. Feature labels are green for protein-sequence derived and brown for nucleotide-sequence derived. **(C)** Feature category dependence within the training set is shown by Spearman's ρ and 95% CI between the normalized outcomes versus the feature subset. **(D)** Considering the NYCOMPS data set (as in Fig 2), the Area Under the Curve (AUC) of a Receiver Operating Characteristic and 95% confidence interval when predicting solely by features from the specified category against the NYCOMPS dataset. Red, using positive only as the cut-off for individual genes (Fig 2C); grey, using positive outcomes within each plasmid and solubilization condition (as in Fig 2E).

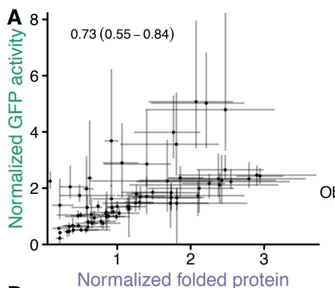
S1 Table. Sequence parameter weights and descriptions. Weights are presented after normalizing to the mean value for clarity. Features that were calculated but removed in pre-processing are noted (Methods 3).

S2 Table. AUC values for the NYCOMPS dataset. AUC values and 95% confidence intervals are presented in summary, by expression condition, and by predicted C-terminal localization as well as for IMProve scores calculated without the most computationally expensive RNA secondary structure calculation (as in Fig 5).

S3 Table. Predictive performances of the model across protein families. The proteins and performances are with respect to those tested by NYCOMPS as summarized in Fig 5. This data is available in an interactive format at clemonslab.caltech.edu.

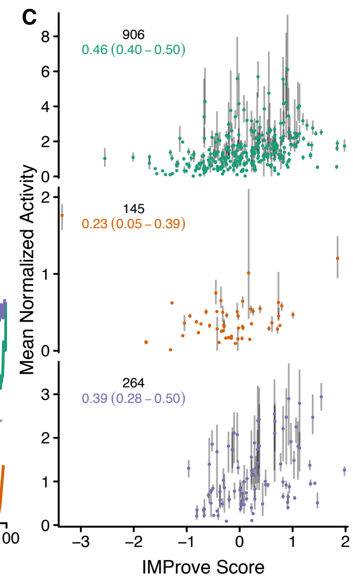
S4 Table. Full list of predicted membrane proteins. This includes corresponding identifiers, descriptions, Pfam families, coding sequences, and IMProve scores. This data is available in an interactive format at clemonslab.caltech.edu.

Figure 1



B

	Daley, Rapp, et al., 2005		Fluman, et al., 2014
	C–terminal Cytoplasmic (GFP)	C–terminal Periplasmic (PhoA)	C–terminal Cytoplasmic (GFP, folded protein)
Plates	27	27	3
Genes	434	120	92
Observations	1302	360	264
Total Pairs	31980	2672	11485
Kendall's τ	0.359	0.239	0.286



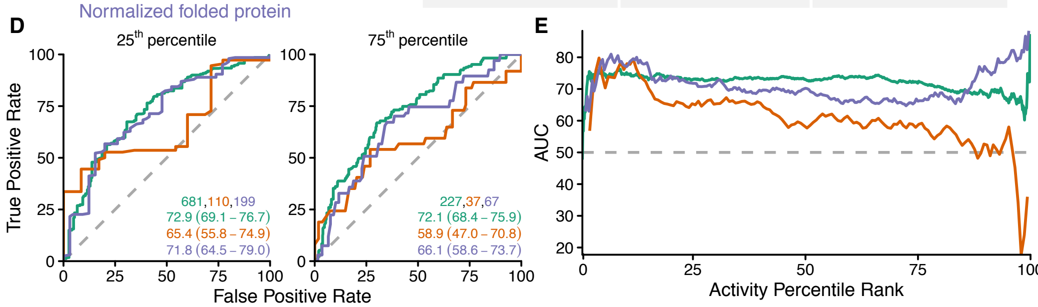


Figure 2

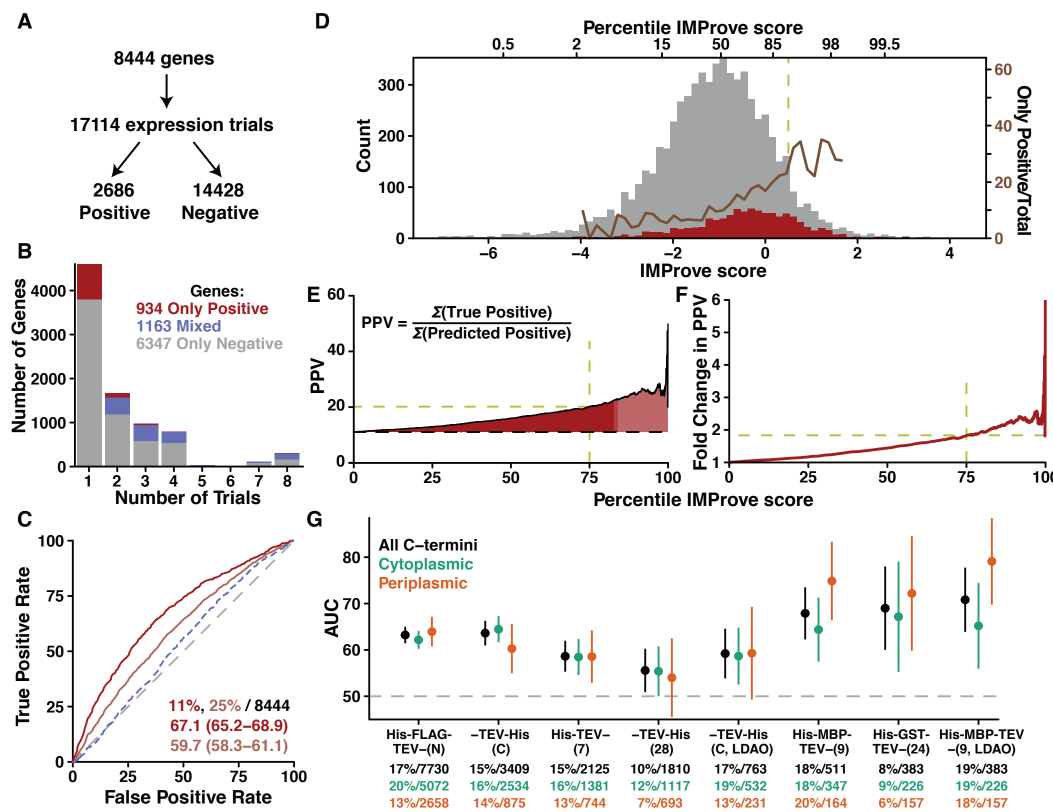

A
8444 genes
17114 expression trials
2686 Positive
14428 Negative
B
Number of Genes
Number of Trials
Genes:
934 Only Positive
1163 Mixed
6347 Only Negative
C
True Positive Rate
False Positive Rate
11%, 25% / 8444
67.1 (65.2–68.9)
59.7 (58.3–61.1)
D
Percentile IMProve score
Count
Only Positive/Total
IMProve score
E
PPV = Σ(True Positive) / Σ(Predicted Positive)
PPV
F
Fold Change in PPV
Percentile IMProve score
G
All C–termini
Cytoplasmic
Periplasmic
AUC
His-FLAG-TEV–(N)
–TEV-His (C)
His-TEV– (7)
–TEV-His (28)
–TEV-His (C, LDAO)
His-MBP-TEV–(9)
His-GST-TEV–(24)
His-MBP-TEV –(9, LDAO)
17%/7730
15%/3409
15%/2125
10%/1810
17%/763
18%/511
8%/383
19%/383
20%/5072
16%/2534
16%/1381
12%/1117
19%/532
18%/347
9%/226
19%/226
13%/2658
14%/875
13%/744
7%/693
13%/231
20%/164
6%/157
18%/157

Figure 3

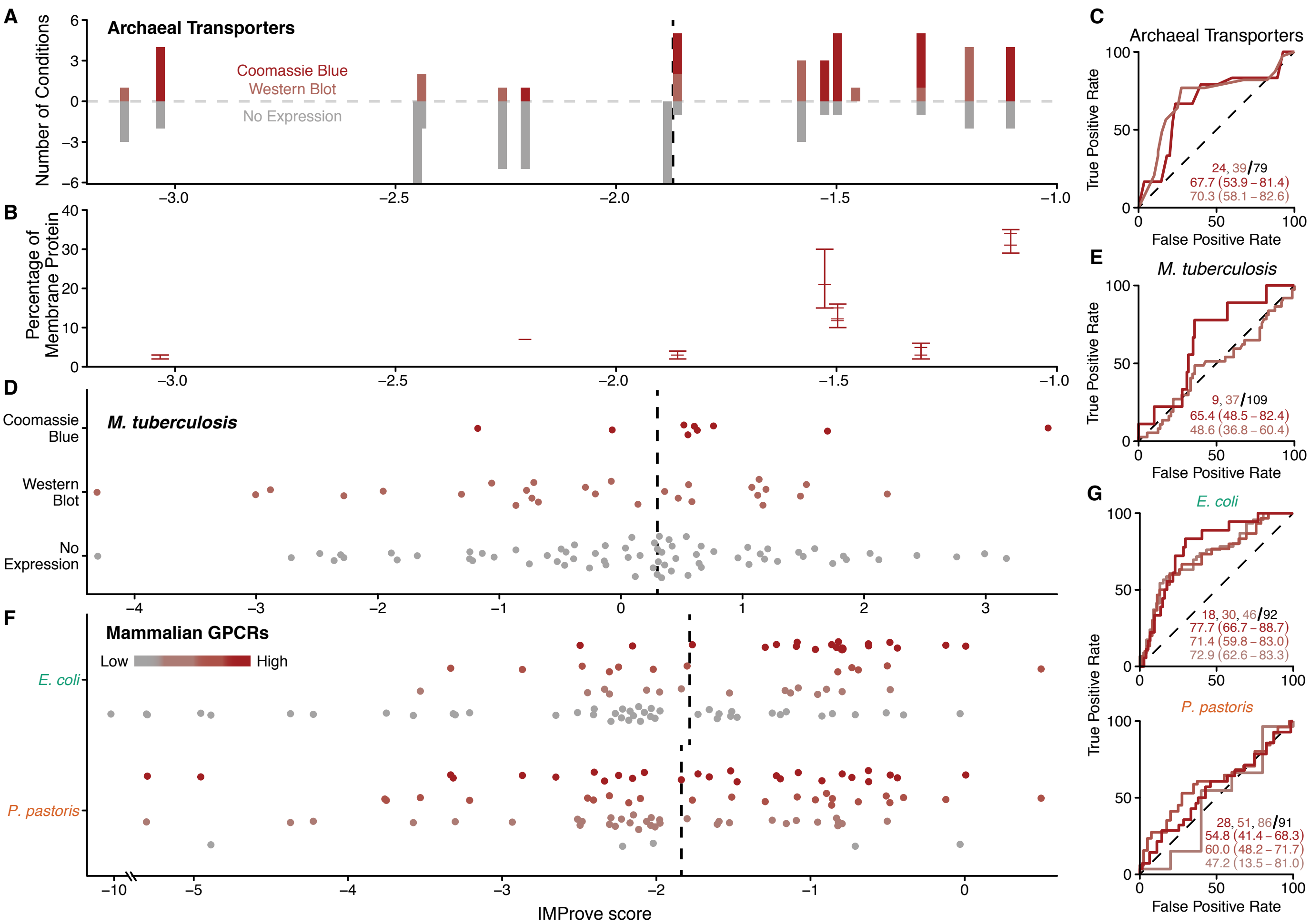
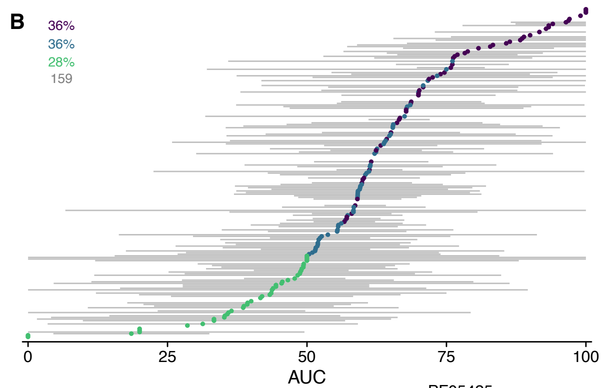

A
Archaeal Transporters
Coomassie Blue
Western Blot
No Expression
Number of Conditions
B
Percentage of Membrane Protein
C
Archaeal Transporters
True Positive Rate
False Positive Rate
24, 39/79
67.7 (53.9 – 81.4)
70.3 (58.1 – 82.6)
D
M. tuberculosis
Coomassie Blue
Western Blot
No Expression
E
M. tuberculosis
9, 37/109
65.4 (48.5 – 82.4)
48.6 (36.8 – 60.4)
F
Mammalian GPCRs
Low
High
E. coli
P. pastoris
IMProve score
G
E. coli
18, 30, 46/92
77.7 (66.7 – 88.7)
71.4 (59.8 – 83.0)
72.9 (62.6 – 83.3)
P. pastoris
28, 51, 86/91
54.8 (41.4 – 68.3)
60.0 (48.2 – 71.7)
47.2 (13.5 – 81.0)

Figure 4

A

	In training	Not in training	No Pfam
Pfam Count	153	402	NA
Positive/Gene Count	11%/5536	9%/1695	12%/1278
AUC (95% CI)	68.2 (65.9–70.4)	67.2 (63.1–71.3)	63.8 (59.2–68.4)

B



C **D**

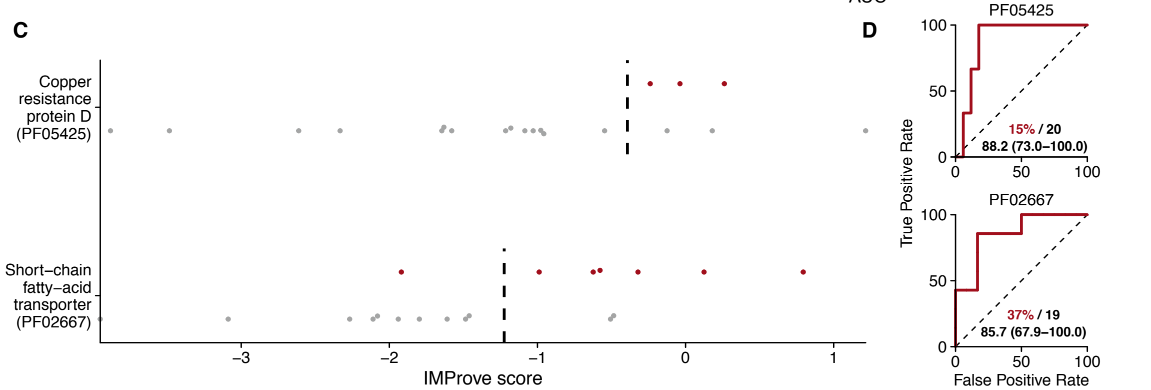


Figure 5

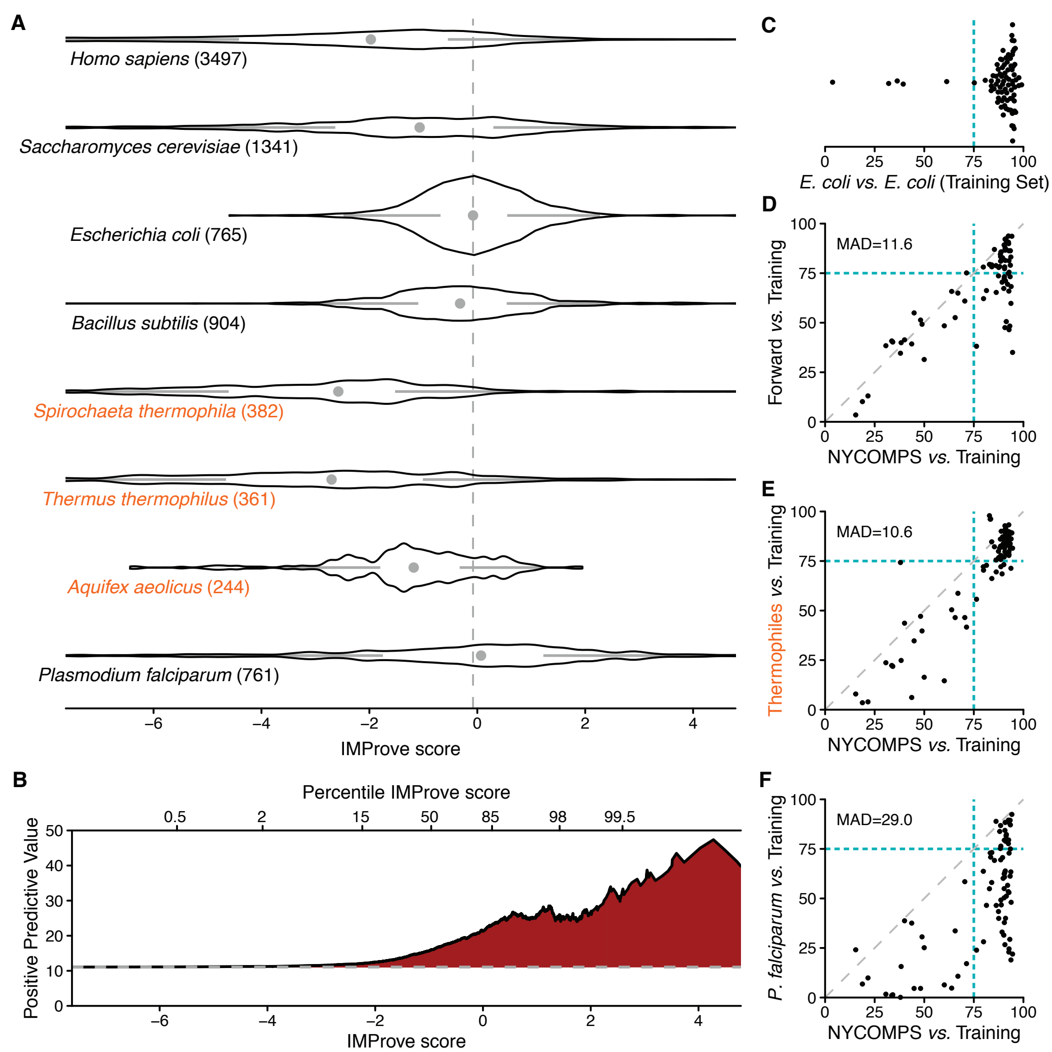

A
Homo sapiens (3497)
Saccharomyces cerevisiae (1341)
Escherichia coli (765)
Bacillus subtilis (904)
Spirochaeta thermophila (382)
Thermus thermophilus (361)
Aquifex aeolicus (244)
Plasmodium falciparum (761)
−6
−4
−2
0
2
4
IMProve score
B
Percentile IMProve score
0.5
2
15
50
85
98
99.5
Positive Predictive Value
IMProve score
C
E. coli vs. E. coli (Training Set)
D
MAD=11.6
Forward vs. Training
NYCOMPS vs. Training
E
MAD=10.6
Thermophiles vs. Training
F
MAD=29.0
P. falciparum vs. Training

Figure 6

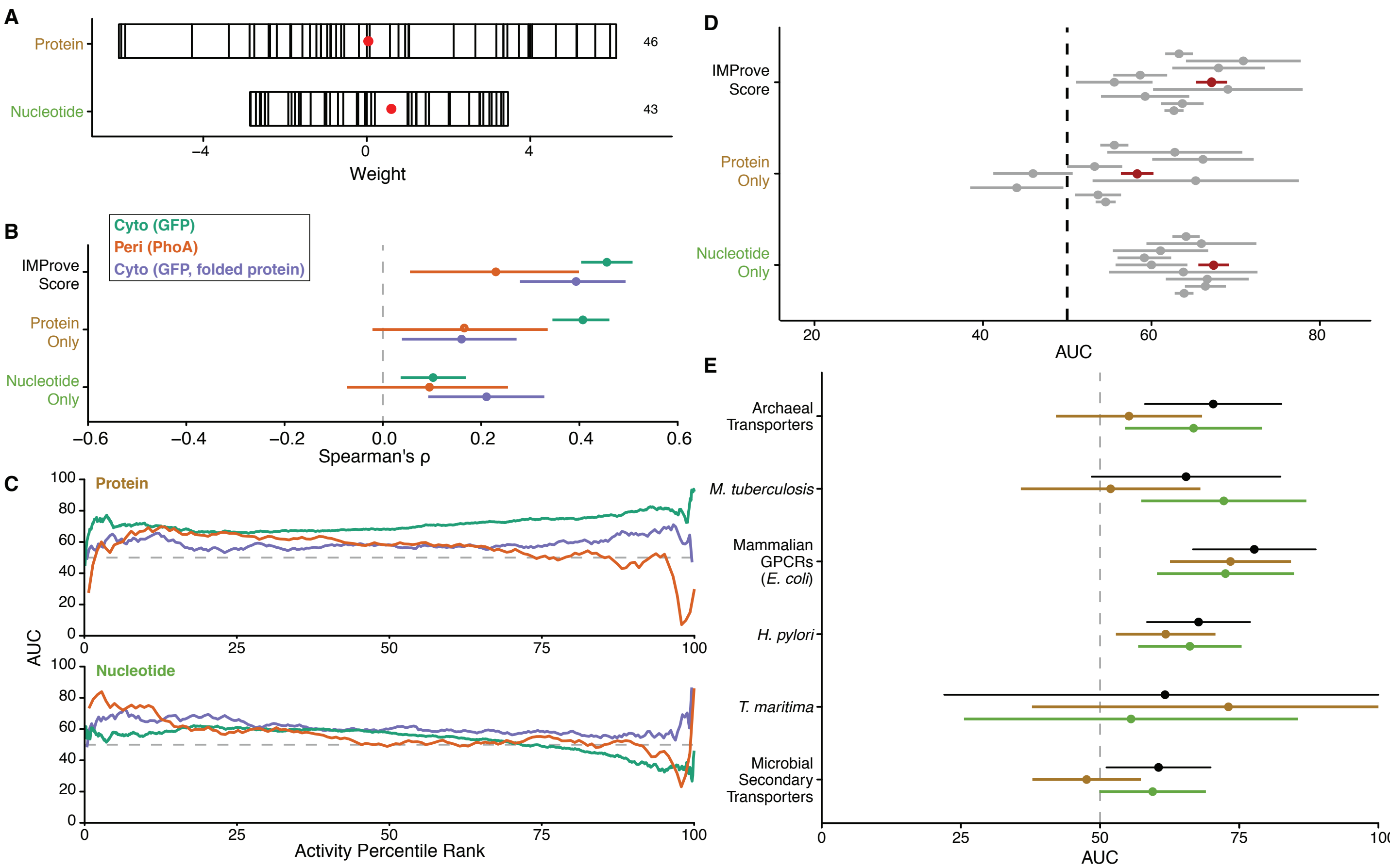
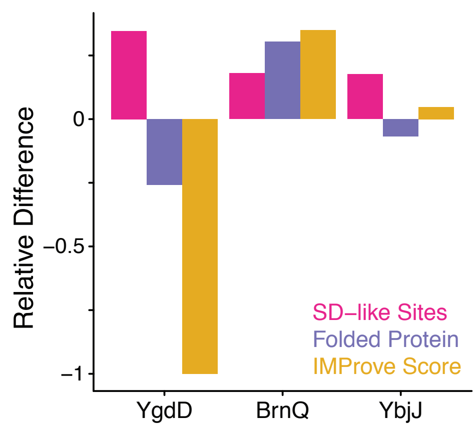
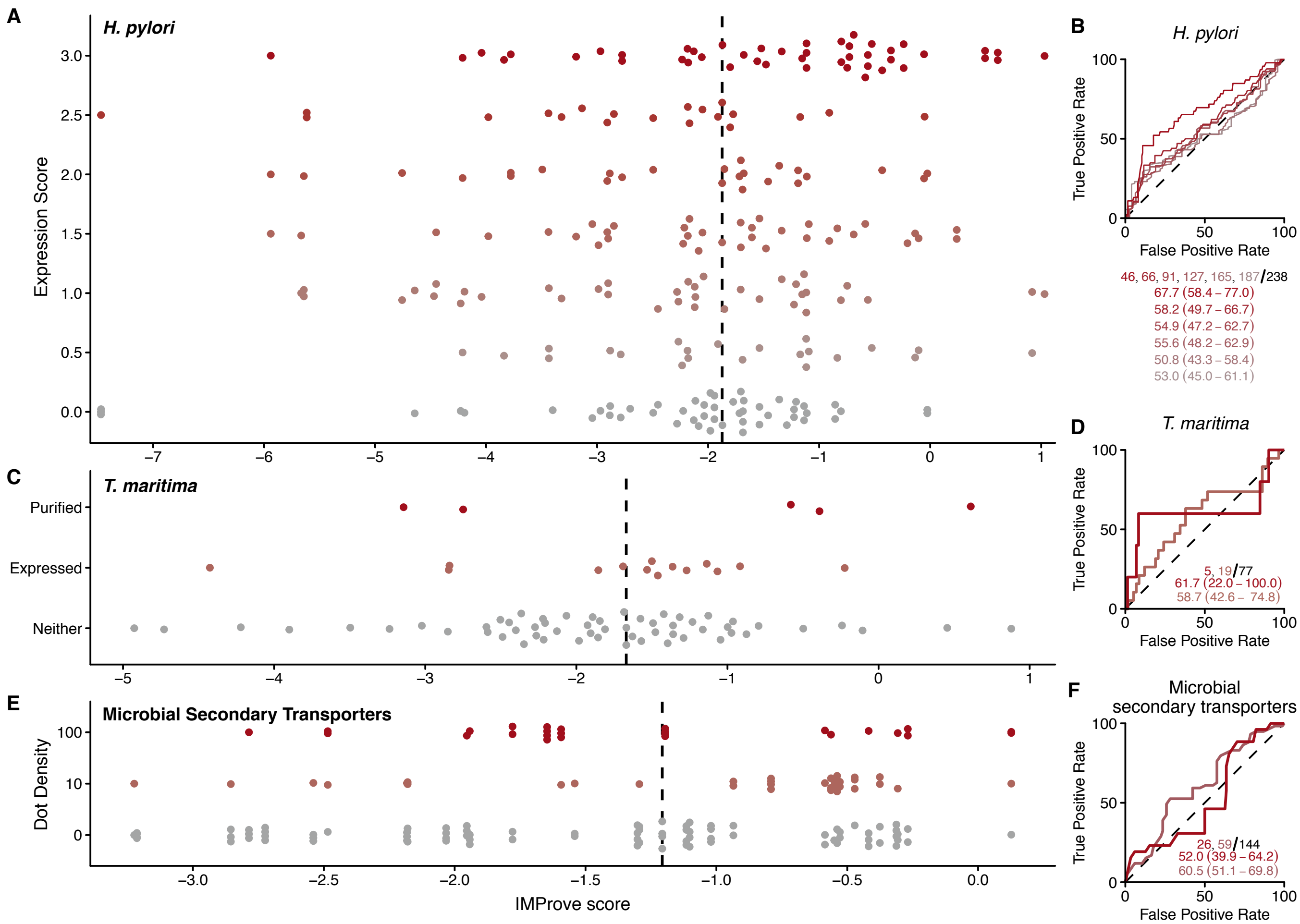

A
Protein
Nucleotide
46
43
-4
0
4
Weight
B
Cyto (GFP)
Peri (PhoA)
Cyto (GFP, folded protein)
IMProve Score
Protein Only
Nucleotide Only
-0.6
-0.4
-0.2
0.0
0.2
0.4
0.6
Spearman's ρ
C
Protein
Nucleotide
AUC
Activity Percentile Rank
D
IMProve Score
Protein Only
Nucleotide Only
AUC
E
Archaeal Transporters
M. tuberculosis
Mammalian GPCRs (E. coli)
H. pylori
T. maritima
Microbial Secondary Transporters
AUC

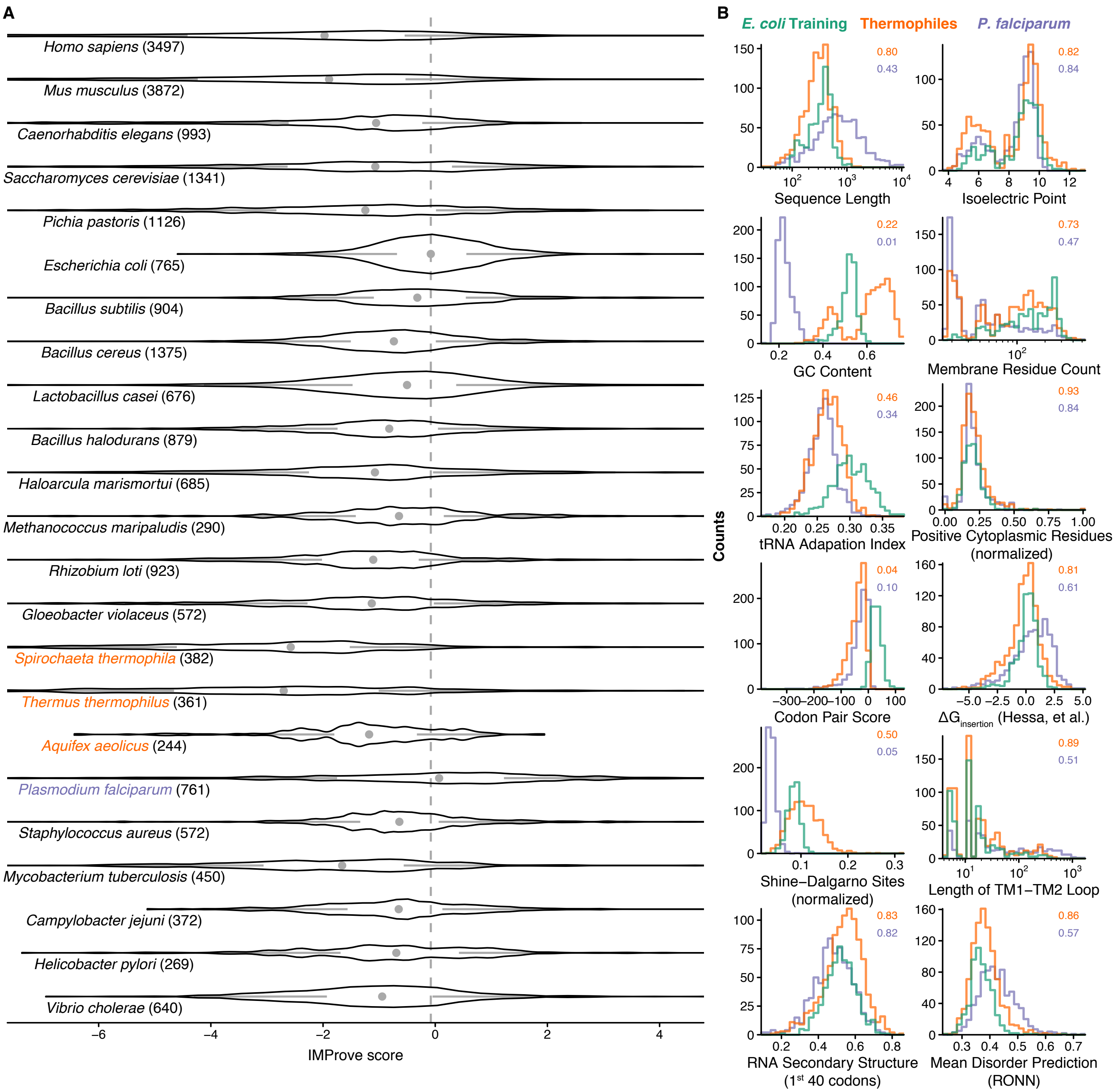
Figure 7


Relative Difference
0
−0.5
−1
YgdD
BrnQ
YbjJ
SD−like Sites
Folded Protein
IMProve Score

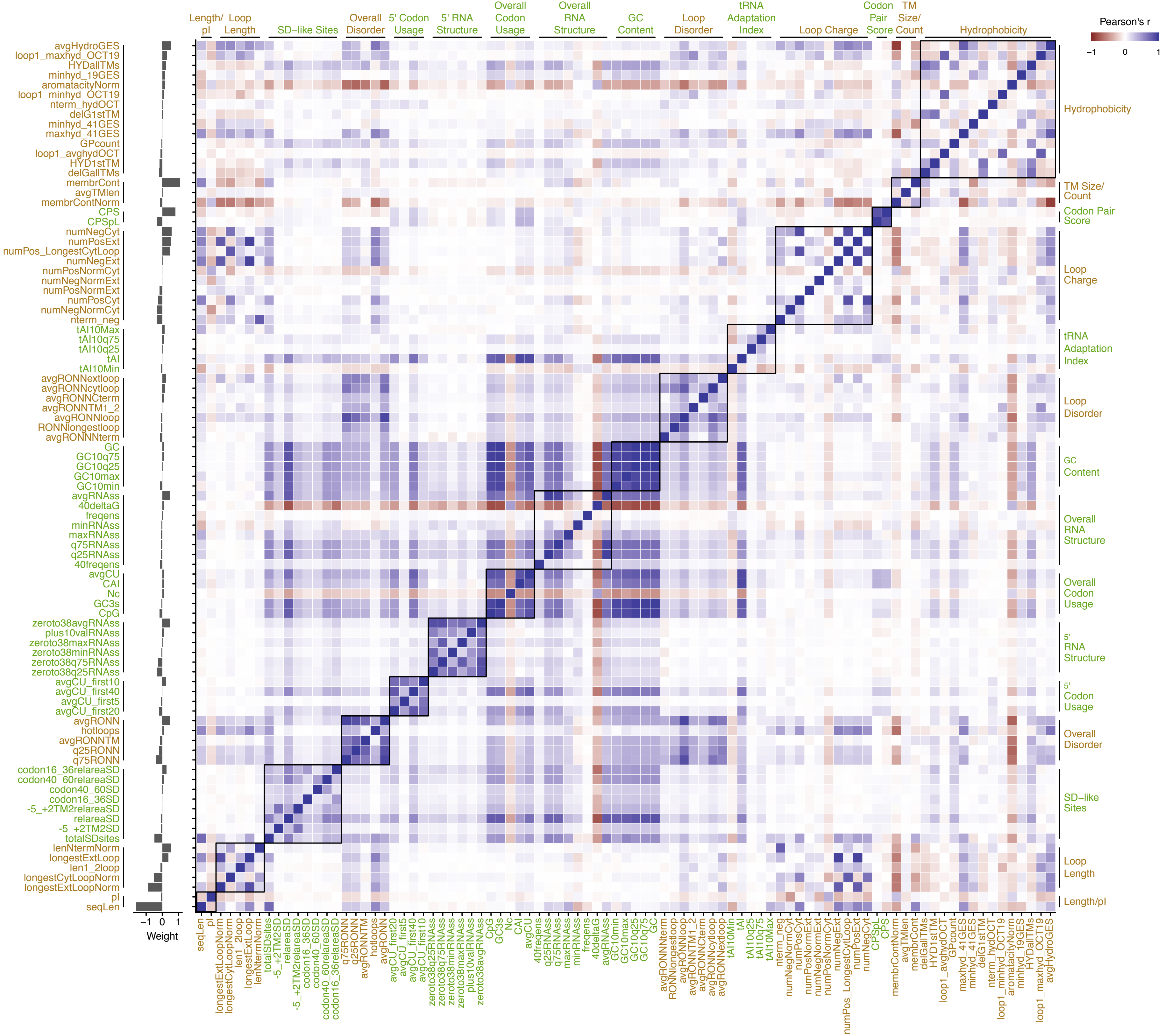
S1 Figure


A
H. pylori
Expression Score
3.0
2.5
2.0
1.5
1.0
0.5
0.0
−7
−6
−5
−4
−3
−2
−1
0
1
B
H. pylori
True Positive Rate
False Positive Rate
46, 66, 91, 127, 165, 187/238
67.7 (58.4 – 77.0)
58.2 (49.7 – 66.7)
54.9 (47.2 – 62.7)
55.6 (48.2 – 62.9)
50.8 (43.3 – 58.4)
53.0 (45.0 – 61.1)
C
T. maritima
Purified
Expressed
Neither
D
T. maritima
5, 19/77
61.7 (22.0 – 100.0)
58.7 (42.6 – 74.8)
E
Microbial Secondary Transporters
Dot Density
100
10
0
−3.0
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
IMProve score
F
Microbial secondary transporters
26, 59/144
52.0 (39.9 – 64.2)
60.5 (51.1 – 69.8)

S2 Figure

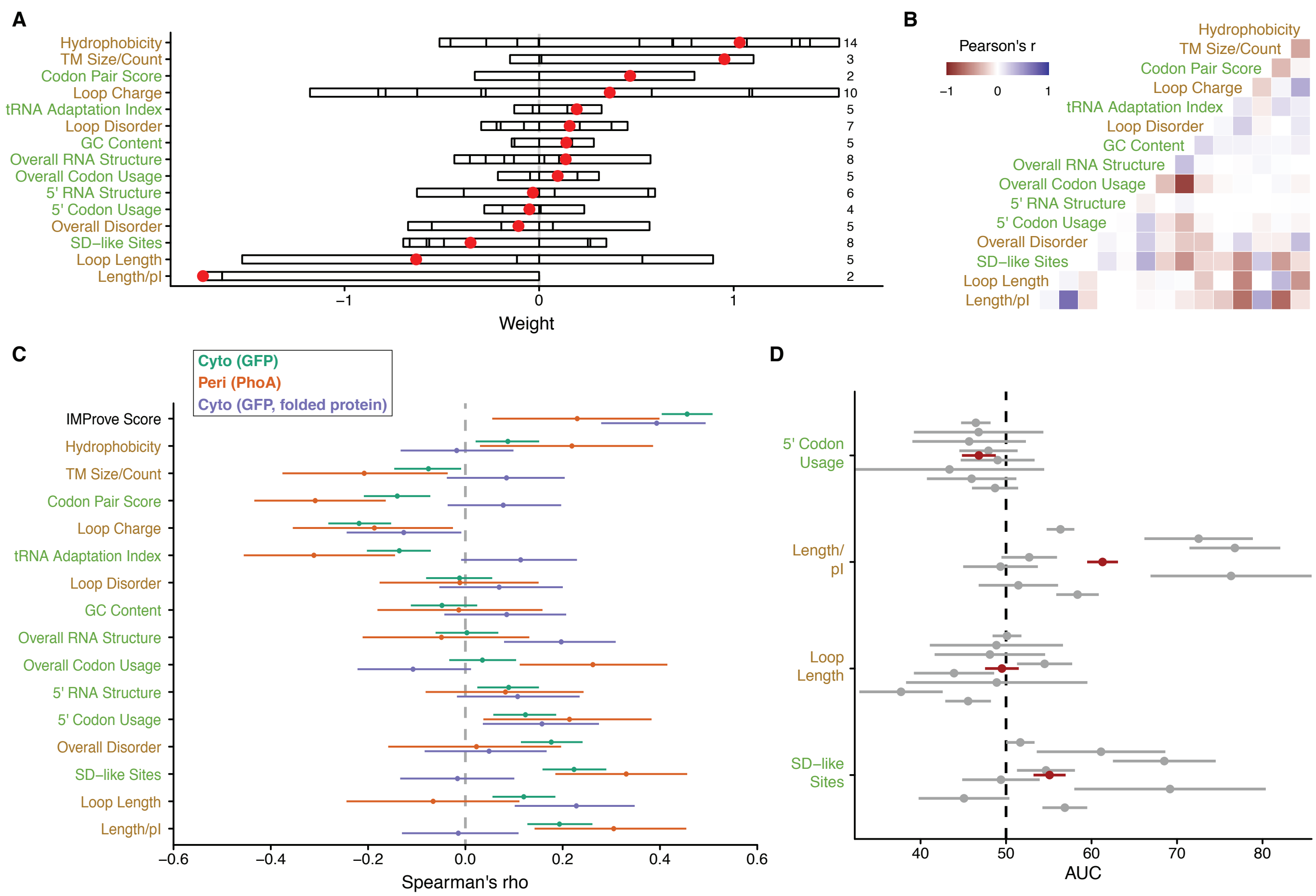

A
Homo sapiens (3497)
Mus musculus (3872)
Caenorhabditis elegans (993)
Saccharomyces cerevisiae (1341)
Pichia pastoris (1126)
Escherichia coli (765)
Bacillus subtilis (904)
Bacillus cereus (1375)
Lactobacillus casei (676)
Bacillus halodurans (879)
Haloarcula marismortui (685)
Methanococcus maripaludis (290)
Rhizobium loti (923)
Gloeobacter violaceus (572)
Spirochaeta thermophila (382)
Thermus thermophilus (361)
Aquifex aeolicus (244)
Plasmodium falciparum (761)
Staphylococcus aureus (572)
Mycobacterium tuberculosis (450)
Campylobacter jejuni (372)
Helicobacter pylori (269)
Vibrio cholerae (640)
IMProve score
B
E. coli Training
Thermophiles
P. falciparum
Counts
Sequence Length
Isoelectric Point
GC Content
Membrane Residue Count
tRNA Adapation Index
Positive Cytoplasmic Residues (normalized)
Codon Pair Score
ΔG insertion (Hessa, et al.)
Shine–Dalgarno Sites (normalized)
Length of TM1–TM2 Loop
RNA Secondary Structure (1st 40 codons)
Mean Disorder Prediction (RONN)

S3 Figure



Pearson's r
−1
0
1
Length/pI
Loop Length
SD−like Sites
Overall Disorder
5' Codon Usage
5' RNA Structure
Overall Codon Usage
Overall RNA Structure
GC Content
Loop Disorder
tRNA Adaptation Index
Loop Charge
Codon Pair Score
TM Size/Count
Hydrophobicity
avgHydroGES
loop1_maxhyd_OCT19
HYDallTMs
minhyd_19GES
aromatacityNorm
loop1_minhyd_OCT19
nterm_hydOCT
delG1stTM
minhyd_41GES
maxhyd_41GES
GPcount
loop1_avghydOCT
HYD1stTM
delGallTMs
membrCont
avgTMlen
membrContNorm
CPS
CPSpL
numNegCyt
numPosExt
numPos_LongestCytLoop
numNegExt
numPosNormCyt
numNegNormExt
numPosNormExt
numPosCyt
numNegNormCyt
nterm_neg
tAI10Max
tAI10q75
tAI10q25
tAI
tAI10Min
avgRONNextloop
avgRONNcytloop
avgRONNCterm
avgRONNTM1_2
avgRONNloop
RONNlongestloop
avgRONNNterm
GC
GC10q75
GC10q25
GC10max
GC10min
avgRNAss
40deltaG
freqens
minRNAss
maxRNAss
q75RNAss
q25RNAss
40freqens
avgCU
CAI
Nc
GC3s
CpG
zeroto38avgRNAss
plus10valRNAss
zeroto38maxRNAss
zeroto38minRNAss
zeroto38q75RNAss
zeroto38q25RNAss
avgCU_first10
avgCU_first40
avgCU_first5
avgCU_first20
avgRONN
hotloops
avgRONNTM
q25RONN
q75RONN
codon16_36relareaSD
codon40_60relareaSD
codon40_60SD
codon16_36SD
-5_+2TM2relareaSD
relareaSD
-5_+2TM2SD
totalSDsites
lenNtermNorm
longestExtLoop
len1_2loop
longestCytLoopNorm
longestExtLoopNorm
pI
seqLen
−1 0 1
Weight

S4 Figure


A
Hydrophobicity
TM Size/Count
Codon Pair Score
Loop Charge
tRNA Adaptation Index
Loop Disorder
GC Content
Overall RNA Structure
Overall Codon Usage
5' RNA Structure
5' Codon Usage
Overall Disorder
SD–like Sites
Loop Length
Length/pI
14
3
2
10
5
7
5
8
5
6
4
5
8
5
2
–1
0
1
Weight
B
Pearson's r
–1
0
1
C
Cyto (GFP)
Peri (PhoA)
Cyto (GFP, folded protein)
IMProve Score
–0.6
–0.4
–0.2
0.0
0.2
0.4
0.6
Spearman's rho
D
5' Codon Usage
Length/ pI
Loop Length
SD–like Sites
40
50
60
70
80
AUC