

Decoding sequence-level information to predict membrane protein expression

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

Affiliation

Division of Chemistry and Chemical Engineering, California Institute of Technology, Pasadena, CA
91125

*Correspondence to: clemons@caltech.edu

1 **Abstract**

2 The expression and purification of integral membrane proteins remains a major bottleneck in the
3 characterization of these important proteins. Expression levels are currently unpredictable, which
4 renders the pursuit of these targets challenging and highly inefficient. Evidence demonstrates that small
5 changes in the nucleotide or amino-acid sequence can dramatically affect membrane protein biogenesis;
6 yet these observations have not resulted in generalizable approaches to improve expression. In this
7 study, we develop a data-driven statistical model that predicts membrane protein expression in *E. coli*
8 directly from sequence. The model, trained on experimental data, combines a set of sequence-derived
9 variables resulting in a score that predicts the likelihood of expression. We test the model against
10 various independent datasets from the literature that contain a variety of scales and experimental
11 outcomes demonstrating that the model significantly enriches expressed proteins. The model is then
12 used to score expression for membrane proteomes and protein families highlighting areas where the
13 model excels. Surprisingly, analysis of the underlying features reveals an importance in nucleotide
14 sequence-derived parameters for expression. This computational model, as illustrated here, can
15 immediately be used to identify favorable targets for characterization.

1 **Introduction**

2 The central role of integral membrane proteins motivates structural and biophysical studies that
3 require large amounts of purified protein, often at considerable cost of both material and labor. Only a
4 small percentage can be produced at high-levels resulting in membrane protein structural
5 characterization lagging roughly 20 years behind that of soluble proteins¹. To increase the pace of
6 structure determination, the scientific community created large government-funded structural genomics
7 consortia facilities, like the NIH-funded New York Consortium on Membrane Protein Structure
8 (NYCOMPS)². For this representative example, more than 8000 genes, chosen based on characteristics
9 hypothetically related to success, yielded only 600 (7.1%) highly expressing proteins³ resulting to date
10 in 34 (0.4%) unique structures (based on annotation in the RCSB PDB⁴). Despite considerable
11 investment on many scales, the lack of expressed targets has hampered membrane protein structural
12 studies⁵.

13 Tools for improving the number of expressed membrane proteins are needed. While significant work
14 has shown promise on a case-by-case basis, *e.g.* growth at lower temperatures, codon optimization⁶, and
15 regulating transcription⁷, a generalizable solution remains elusive. Currently, each target must be
16 addressed individually as the conditions that were successful for a previous target seldom carry over to
17 other proteins, even amongst closely related homologs^{8,9}. For individual cases, simple changes can have
18 dramatic effects on the amount of expressed proteins^{10,11}. Considering the scientific value of membrane
19 protein studies and absence of systematic methods to improve expression, it is surprising that no method
20 can provide solutions with broad applicability across protein families and genomes nor describe the
21 variation in expression levels seen between closely related sequences.

22 While there are no approaches to broadly decode sequence-level information precluding its use for
23 predicting membrane protein expression, the concept that sequence variation can measurably influence

1 membrane protein biogenesis is commonplace. For example, positive-charges on cytoplasmic loops are
2 important determinants of membrane protein topology^{12,13}; yet introduction of mutations presumed to
3 enhance certain properties, such as the positive inside rule, has not proven generalizable for improving
4 expression⁹. This reasons for this likely lie in the complex underpinnings of membrane protein
5 biogenesis, where the interplay between sequence parameters at the protein and nucleotide levels must
6 be considered. Optimizing for a single sequence-level feature likely diminishes the beneficial effect of
7 other features (*e.g.* increasing positive residues on internal loops might diminish favorable mRNA
8 properties). Without accounting for the broad set of features related to membrane protein expression, it
9 is impossible to predict differences in expression.

10 To connect sequence to prediction, we develop a statistical model that maps a set of sequences to
11 experimental expression levels via calculated features—thereby simultaneously accounting for the many
12 determinants of expression. The resulting model allows ranking of any arbitrary set of membrane protein
13 sequences in order of their relative likelihood of successful expression. In doing so, we leverage the
14 corpus of work that shows that sequence-level characteristics are important determinants of protein
15 biogenesis, *e.g.* RNA secondary structure^{14,15}, transmembrane segment hydrophobicity¹⁶, the positive
16 inside rule¹⁷, and loop disorder¹⁸. 105 of these sequence-derived parameters were calculated for
17 individual proteins within datasets of interest (Table S1). As the first report to predict expression, we
18 train a linear equation that provides a score based on calculating the sum of weighted features where the
19 weights are derived from fitting to experimental expression data, a “training set.” This model can be
20 used broadly to score any membrane protein based on its calculated features. We validate the model
21 against a variety of independent datasets demonstrating its generalizability. To support further
22 experimental efforts, we broadly score the membrane proteome from a variety of important genomes

1 and showcase the performance of the model across protein families. This approach and resulting model
2 provides an exciting example for connecting sequence space to complex experimental outcomes.

3

4 **Results**

5 For this study, we focus on heterologous expression in *E. coli*, due to its ubiquitous use as a tool for
6 membrane protein expression. While the benefits derived from low cost and low barriers for adoption
7 are obvious, the applicability to the spectrum of the membrane proteome are becoming clearer. Of note,
8 43 of the 216 unique eukaryotic membrane protein structures were solved using protein expressed in *E.*
9 *coli* (based on annotation in the RCSB PDB⁴). This demonstrates the utility of *E. coli* as a broad tool and
10 its potential if the expression problem can be overcome.

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

12 A key component of any machine learning model is the choice of dataset used for training. Having
13 searched the literature, we identified two publications that contained quantitative datasets on the
14 heterologous expression of *E. coli* membrane proteins in *E. coli*. The first set, Daley, Rapp *et al.*,
15 contained activity measures, proxies for expression level, from C-terminal tags of either GFP or PhoA
16 (alkaline phosphatase)¹⁹. The second set, Fluman *et al.*, contained a more detailed analysis of a subset
17 from the first utilizing in-gel fluorescence to measure folded protein²⁰ (see Methods 4c). The expression
18 results strongly correlated between the two datasets notably in that normalized GFP activity was a good
19 measure of the amount of folded membrane protein (Figure 1a, also ²¹). The experimental set-up
20 employed multiple 96-well plates over multiple days resulting in pronounced variability in the absolute
21 expression level of a given protein between trials. Daley, Rapp, *et al.* calculated average expression
22 levels by dividing the raw expression level of each protein by that of a control construct (Inverse LepB-
23 GFP or LepB-PhoA) on the corresponding plate. While the resulting values were useful for the relevant

1 question of identifying topology, we were unable to successfully fit a linear regression or a standard
2 linear-SVM on either the raw data compiled from all plates or averaged outcomes of each gene. This
3 unexpected outcome suggested that the measurements required a more complex analysis.

4 We hypothesized that measurements could be more accurately compared within an individual plate
5 then across the entire dataset. To account for this, a preference-ranking linear SVM algorithm (SVM^{rank}
6 ²²) was chosen (see Methods 4b). Simply put, the SVM^{rank} algorithm determines the optimal weight for
7 each feature to best rank the order of expression outcomes within each plate over all plates, which
8 results in a model where higher expressing proteins have higher scores. The outcome is identical in
9 structure to a multiple linear regression, but instead of minimizing the sum of squared residuals, the
10 SVM cost function is used accounting for the plate-wise constraint specified above. In practice, the
11 process optimizes as a training metric the correlation coefficient Kendall's τ to converge upon a set of
12 weights. Kendall's τ measures the agreement between ordinal quantities by calculating correctly-ordered
13 and swapped pairs.

14 Various metrics related to the training data can be derived to assess the accuracy with which the
15 model fits the input data (see Methods 4C). The SVM^{rank} training metric shows varying agreement for
16 all groups (*i.e.*, $\tau_{\text{kendall}} > 0$) (Figure 1b). For individual genes, activity values normalized and averaged
17 across trials were not directly used for the training procedure (see Methods 4a); yet one would anticipate
18 that scores for each gene should broadly correlate with expression. Indeed, the observed normalized
19 activities positively correlate with the SVM^{rank} score output by the model (Figure 1c).

20 For a more quantitative approach to assessing the models success within the training data, we turn to
21 the Receiver Operating Characteristic (ROC). ROC curves quantify the tradeoff between true positive
22 and false positive predictions across the numerical scores output from a predictor. This is a more reliable
23 assessment of prediction than simply calculating accuracy and precision from a single, arbitrary score

1 threshold ²³. The figure of merit that quantifies an ROC curve is the Area Under the Curve (AUC).
2 Given that the AUC for a perfect predictor corresponds to 100% and that of a random predictor is 50%
3 (Figure 1d, grey dashed line), an AUC greater than 50% indicates predictive performance of the model
4 (percentage signs hereafter omitted) (see Methods 5 and ²³). Here, the ROC framework will be used to
5 quantitatively assess the ability of our model to predict the outcomes within the various datasets.

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

18 *Demonstration of prediction against an independent large expression dataset*

19 While the above analyses show that the model successfully fits the training data, we assess the
20 broader applicability of the model based on its success at predicting the outcomes of independent large-
21 and small-scale expression trials. The first test considers results from NYCOMPS, where 8444
22 membrane protein genes entered expression trials, in up to eight conditions, resulting in 17114
23 expression outcomes². The majority of genes were attempted in only one condition (Figure 2a), and

1 outcomes were non-quantitative (binary: expressed or not expressed) as indicated by the presence of a
2 band by Coomassie staining of an SDS-PAGE gel after small-scale expression, solubilization, and
3 purification³. Therefore, for this analysis, we consider the experimental results in various ways: either
4 outcomes per gene (if at least one trial is positive, the gene is considered positive for expression), all
5 conditions (each expression trial considered independently), or based on defined expression conditions.
6 For the first, several metrics demonstrate prediction (Figure 2b-d).

7 A major aim of this work is to enrich the likelihood of choosing positively expressing proteins. The
8 positive predictive value (PPV, true positives \div predicted positives) becomes a useful metric for positive
9 enrichment as it conveys the degree of improvement over the experimental baseline of the dataset. The
10 PPV of the model is plotted as a function of the percentile of the SVM^{rank} score threshold for the
11 definition of predicted positives (Figure 2b). In the figure, the overall positive percentage (~24%), an
12 experimental baseline, is represented by a grey dashed line; therefore, a relative increase reflects the
13 increased predictive power of the algorithm. For example, considering the top fourth of genes by
14 SVM^{rank} score (75th percentile) shows that the algorithm enriches for positive outcomes by 8.4% over
15 baseline. Seen another way, a histogram of the SVM^{rank} score for each protein is plotted separated by
16 positive versus negative outcomes (Figure 2c). Visually, the distribution of the scores for the positive
17 group is shifted to a higher score relative to the negative group, which is substantiated quantitatively by
18 the ROC and its corresponding AUC (Figure 2d). Interestingly, considering the predictive power against
19 all conditions as opposed to by gene shows slightly better statistics (AUC=62.6) reflective of the fact
20 that many genes have mixed outcomes (Figure 2e). Importantly, the model shows consistent
21 performance throughout each of the eight possible conditions tested (Figure 2e, black, numerically in
22 Table S2).

1 The ability to predict the experimental data from NYCOMPS allows a return to the question of
2 alkaline phosphatase as a metric for expression. To investigate the trend that the expression of proteins
3 with periplasmic C-termini measured by alkaline phosphatase (Figure 1, orange) show less consistent
4 fitting by the model, the NYCOMPS outcomes are split by putative C-terminal localization as predicted
5 by Phobius²⁴. No significant difference in AUC between C-terminal localizations across all conditions
6 (Figure 2e, green vs. orange) indicate that the model is applicable for all topologies.

7 *Further demonstration of prediction against small-scale independent datasets*

8 The NYCOMPS example demonstrates the predictive power of the model across the broad range of
9 sequence space encompassed by that dataset. Next, the performance of the model is tested against
10 relevant subsets of sequence space (*e.g.* a family of proteins or the proteome from a single organism),
11 which are reminiscent of laboratory-scale experiments that precede structural or biochemical analyses.
12 While a number of datasets exist^{8,25–35}, we could only identify six for which complete sequence
13 information could be obtained to calculate all the necessary sequence parameters^{25–30}.

14 The first dataset is derived from the expression of 14 archaeal transporters in *E. coli* chosen based on
15 their homology to human proteins²⁵. For each putative transporter, expression was performed in three
16 plasmids and two strains (six total conditions) with the membrane fraction quantified by both a Western
17 blot against a histidine-affinity tag and Coomassie Blue staining of an SDS-PAGE gel. Here, the
18 majority of the expressing proteins fall into the top half of the SVM^{rank} scores, 7 out of 9 of those with
19 multiple positive outcomes (Figure 3a, top). Strikingly, quantification of the Coomassie Blue staining
20 highlights a clear correlation with the SVM^{rank} score where the higher expressing proteins have the
21 highest score (Figure 3a, bottom). ROC curves are plotted for the two thresholds: expression detected at
22 least by Western blot or, for the smaller subset, by Coomassie Blue (Figure 3b). In both cases, the model
23 shows predictive power.

1 The next test considers the expression of 105 *Mycobacterium tuberculosis* proteins in *E. coli*²⁶.
2 Protein expression was measured both by Coomassie Blue staining of an SDS-PAGE gel and Western
3 blot with only outcomes from the membrane fraction considered for this analysis. The highest
4 expressing proteins (detected via Coomassie Blue) follow the trend given by the SVM^{rank} score with 7 of
5 the 9 falling within the top half of scoring proteins (Figure 3c) and is reflected in the ROC (Figure 3d).
6 In contrast, using the positive Western blot outcomes as the minimum threshold (Figure 3c) shows an
7 AUC no better than random (Figure 3d). Given that no internal standard was used and that each
8 expression trial was performed only once, proteins that were positive by Western blot may represent a
9 pool indistinguishable in expression from those not detected; alternatively, these results support that our
10 statistical model accurately captures the most highly expressing proteins.

11 A broader test considers expression trials of 101 mammalian GPCRs in bacterial and eukaryotic
12 systems²⁷. Trials in *E. coli*, measured via Western blot of an insoluble fraction, again show highly
13 expressing proteins at higher SVM^{rank} scores while the expression of the same proteins in *P. pastoris*,
14 measured via dot blot, fail to show broad agreement (Figure 3e,f). The lack of predictive performance in
15 *P. pastoris* suggests that the parameterization of the model, calibrated for broadly characterizing *E. coli*
16 expression, requires retraining to generate a different model that captures the distinct interplay of
17 sequence parameters in yeast.

18 Further expression trials of membrane proteins from *H. pylori*, *T. maritima* as well as microbial
19 secondary transporters continues to show the same broad agreement²⁸⁻³⁰ (Figure S1). *H. pylori*
20 membrane proteins showed that as the threshold for positive expressing proteins increases, the
21 performance of the model improves (using the highest threshold n=46 and AUC=67.7) (Figure S1a,b).
22 For *T. maritima* expression, the model weakly captures outcomes for two defined thresholds (n=5 and
23 19, AUC=61.7 and 58.7), but due to the small number of successful outcomes, the confidence intervals

1 are broad (Figure S1c,d). The expression of microbial secondary transporters shows varied agreement
2 with the model. Taking proteins at the lower defined expression threshold shows predictive performance
3 (n=59, AUC=60.5); however, considering the defined high-expressing proteins is less conclusive (n=26,
4 AUC=52.0) (Figure S1e,f).

5 *Forward predictions on genomes of interest*

6 The model successfully enriches for heterologous expression of membrane proteins in *E. coli*
7 strikingly across scales, laboratories, quantification methods, and protein families supporting its broad
8 generalizability. While few genes express in every condition tested (Figure 2a and 3a), the model
9 predicts the likelihood that a gene will express within a set of conditions and enriches for those that will
10 work in any condition (Figure 2e, numerically in Table S2). Notably, had this model been implemented
11 during the NYCOMPS target selection process, only testing targets with an SVM^{rank} score greater than
12 0.5 (90th percentile or above), based on known outcomes, the percentage of successful genes would have
13 increased from 25% to 37% (Figure 2b). For perspective, testing the same number of genes would have
14 resulted in an additional 912 expressed proteins, representing a significant improvement in the return on
15 investment.

16 To expand on the utility of this model, SVM^{rank} scores were calculated for membrane proteins from a
17 variety of metazoan and microbial genomes (Figure 4a and Figure 2a). Many genomes have a significant
18 proportion of proteins with high scores particularly evidenced by portions of the distributions ahead of
19 the median in *E. coli* given by the vertical dashed line (Figure 4a). The likelihood for successful
20 expression may be inferred by equating SVM^{rank} score with the PPV from the most prevalent
21 NYCOMPS expression condition which rises dramatically at scores above zero (Figure 4b). The range
22 of scores spans those representative of high-expressing membrane proteins in both *E. coli* (Figure 1c) as

1 well as in the NYCOMPS dataset (Figure 2c) and provides suggested targets for future biophysical
2 studies (Table S4).

3 The predictions present several surprises at the biological level. One such is that the distribution of
4 membrane proteins from representative thermophilic bacterial genomes have generally lower relative
5 SVM^{rank} scores than other genomes, which implies that these proteins, on average, are harder to express
6 in *E. coli*. This contrasts the many empirical examples of proteins from thermophiles being used for
7 biophysical characterization. In the case of the malarial parasite *P. falciparum*, the inverse trend is true
8 with higher than expected relative SVM^{rank} scores despite the expectation that these proteins would be
9 hard to express in *E. coli*. A possible cause for the unexpected distribution of scores may lie in the
10 differences in the parameters that define the proteins in these particular groups. As the training set
11 consists only of native *E. coli* sequences, the range of values for each parameter in the training set may
12 not represent the full range of possible values for the parameter. For the special cases highlighted,
13 perhaps the underlying sequence parameters fall into a poorly characterized subset of sequence space
14 bringing into question the applicability of the model for these cases.

15 To address the utility of the model relative to differences in the sampling of sequence parameters, we
16 measure the overlap of the distributions of sequence parameters for a given subset (see Methods 7)
17 (Figure S3b). Simply put, if two subsets contain the same distribution of sequence parameters the
18 expectation is that a given parameter should approach 100%. In the simplest case, comparing the
19 distribution of sequences parameters in all *E. coli* membrane proteins against the subset used in the
20 training set shows that the majority of parameters have overlap values over 75% (Figure 4c), which
21 provides a lower threshold for similarity of sequence parameter range. For NYCOMPS sequences, most
22 of the overlap values relative to the training set are above the threshold. As this set shows predictive
23 performance, comparison to the training set provides a baseline to assess the reliability of predictions

1 within other subsets (Figure 4d-f, x-axis). In the first case (Figure 4d), there is a strong correlation
2 between all the forward predictions and NYCOMPS, *i.e.* values are near the diagonal (quantified by a
3 Mean Absolute Deviation (MAD) = 11.6), suggesting that differences in parameter space do not
4 significantly affect the predictive power of the model. For the thermophiles subset (Figure 4e), the
5 values again are close to the diagonal (*i.e.* low MAD = 10.6) implying that the predictions are credible.
6 *P. falciparum* (Figure 4f), on the other hand, clearly shows stark differences as most parameters fall
7 below the 75% cut-off (MAD = 29.0) bringing into question the reliability of these predictions. A
8 training set with broader coverage of the parameter space may generate a better predictor for all
9 genomes.

10 *Performance of the model across protein families*

11 To provide a clear path forward for experiment, we consider the performance of the model with
12 regards to protein homology families where protein family definitions are based on Pfam
13 classifications³⁶. For outcomes from NYCOMPS (Figure 5a), there are no significant difference in the
14 predictive performance of the model between groups of genes whether or not they are part of a protein
15 family found in the training set.

16 The scale of NYCOMPS allows us to investigate whether there are protein families for which the
17 model does better or worse than the aggregate. For this, an AUC is calculated for each protein family
18 that has minimally five total outcomes (including at least one positive and one negative). Figure 5B plots
19 the AUC for each protein family in increasing order as a cumulative distribution function. The breadth
20 of the AUC values highlights the variability in predictive power across families. Most families can be
21 predicted by the model (115 of 159 have an AUC > 0.5, visually blue and purple) though some not at
22 95% confidence (57 of 115, blue), likely due to an insufficient number tested.

1 For the protein families that are well-predicted within the NYCOMPS set, the model gives accurate
2 insight into the likelihood of expression of a given protein. We demonstrate the utility of this prediction
3 by looking at protein families that have yet to be characterized structurally. While there are a number of
4 choices, a first example is the protein family annotated as short-chain fatty-acid transporters (PF02667),
5 characterized by AtoE in *E. coli*, that typically contains 10 transmembrane domains with an overall
6 length of ~450 amino acids. A second example is the protein family annotated as copper resistance
7 proteins (PF05425), characterized by CopD in *E. coli*, that typically contains eight transmembrane
8 domains with an overall length of ~315 amino acids. In both cases, as indicated by the AUC values, the
9 model provides a clear score cut-off for consideration for expression. For example, considering CopD
10 homologs, one would expect that those with SVM^{rank} scores above -1 will express.

11 *Biological importance of various sequence parameters*

12 Using a simple proof-of-concept linear model allowed for a straightforward and robust predictor;
13 however, intrinsically this complicates determination of biological underpinnings due to the unequal
14 distribution of weight across correlating features. For example, the average $\Delta G_{\text{insertion}}$ of transmembrane
15 segments has a positive weight whereas average hydrophobicity, a correlating parameter, has a negative
16 weight (Table S1, Figure S3). As many parameters, such as those related to hydrophobicity, are highly
17 correlated; conclusive information cannot be obtained simply using weights of individual features to
18 interpret the relative importance of their underlying biological phenomena. An alternative is to collapse
19 related features into biologically meaningful categories reducing correlation (Figure 6a), thereby
20 providing a mechanism to interpret information from the model. For example, the hydrophobicity group
21 incorporates sequence features such as average hydrophobicity, maximum hydrophobicity, $\Delta G_{\text{insertion}}$,
22 etc. The full list of groupings are provided in Table S1 and Figure S3.

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

9 To explore the role of each category in prediction, we calculate the performance of the model by
10 either only using weights from features within a single category or excluding weights from within a
11 single category. We assess predictive performances by calculating ROC curves across genes and
12 expression trials from NYCOMPS dataset for each case (Figure 6c,d). Feature categories that are
13 sufficient for prediction will have an AUC > 0.5 when used alone (Figure 6c) and those necessary for
14 the model will show an AUC < 0.5 when excluded from prediction (Figure 6d). Notably, when only
15 considering all genes independent of condition, most individual categories cannot predict expression
16 (*i.e.* AUC with 95% CI straddling 0.5) (Figure 6c, red). A notable exception is tRNA Adaptation Index,
17 where the per gene AUC is slightly higher than the performance of the model. However, since the model
18 demonstrates predictive performance at 95% confidence across all experimental conditions (Figure 2e),
19 feature categories that are sufficient for prediction must also perform across these conditions. In this
20 case, tRNA Adaptation Index performs poorly against a number of the experimental subsets, so it is not
21 sufficient for prediction. On the other hand, while Codon Pair Score alone shows predictive power
22 across experimental conditions, when excluded from the model, the category alone cannot explain the
23 model (only a single 95% confidence interval crossing AUC = 50, Figure 6d).

1 Importantly, no feature category independently drives the predictor as excluding each individually
2 does not significantly affect the overall predictive performance, except for Length/pI (isoelectric point)
3 (Figure 6d). Sequence length composes the majority of the weight within this category and is one of the
4 highest weighted features in the model. This is consistent with the anecdotal observation that larger
5 membrane proteins are typically harder to express. However, this parameter alone would not be useful
6 for predicting within a smaller subset, like a single protein family, where there is little variance in length
7 (*e.g.* Figure 3,4). One might develop a predictor that was better for a given protein family under certain
8 conditions with a subset of the entire features considered here; yet this would require *a priori* knowledge
9 of the system, *i.e.* which sequence features were truly most important, and would preclude broad
10 generalizability as shown for the predictor presented here.

11 A coarser view of the weights is a comparison of the features derived from either protein or
12 nucleotide sequence. The summed weight for protein features is around zero, whereas for nucleotide
13 features the summed weight is slightly positive suggesting that in comparison these features may be
14 more important to the predictive performance of the model (Figure 6e). Comparison of the predictive
15 performance of the two subsets of weights shows that the nucleotide features alone can give similar
16 performance to the full model (Figure 6f). It is important to note that this does not suggest that protein
17 features are not important for membrane protein expression. Instead, within the context of the trained
18 model, nucleotide features are critical for predictive performance for a large dataset such as NYCOMPS.
19 This finding corroborates growing literature that the nucleotide sequence holds significant determinants
20 of biological processes^{14,20,37–39}.

21 *Sequence optimization for expression*

22 The predictive performance of the model implies that the parameters defined here provide a coarse
23 approximation of the fitness landscape for membrane protein expression. Attempting to optimize a

1 single feature by modifying the sequence will likely affect the resulting score and expression due to
2 changes in other features. Fluman, *et al.*²⁰ provides an illustrative experiment. They hypothesized that
3 altering the number of Shine-Dalgarno (SD)-like sites in the coding sequence of a membrane protein
4 would affect expression. To test this, silent mutations were engineered within the first 200 bases of three
5 proteins (genes *ygdD*, *brnQ*, and *ybjJ* from *E. coli*) to increase the number of SD-like sites with the goal
6 of improving expression. Expression trials demonstrated that only one of the proteins (BrnQ) had
7 improved expression of folded protein (Figure 6g). However, the resulting changes in the SVM^{rank} score
8 correspond with the changes in measured expression as the model considers changes to other nucleotide
9 features. Capture of the outcomes in this small test case by the model illustrates the utility of integrating
10 the contribution of the numerous parameters involved in membrane protein biogenesis.

11

12 **Discussion**

13 The model developed here provides a robust predictor for membrane protein expression. The current
14 best practice for characterization of a membrane protein target begins with the identification and testing
15 of many homologs or variants for expression. The model presented here will allow for prioritization of
16 targets to test for expression thereby making more optimal use of limited human and material resources.
17 In addition, due to the scale of NYCOMPS, protein families that were extensively tested provide ranges
18 of scores (*e.g.* Figure 5c) where the score of an individual target directly indicates its likelihood of
19 expression relative to known experimental results. We provide the current predictor as web service
20 where scores can be calculated, and the method, associated data, and suggested analyses are publically
21 available to catalyze progress across the community (<http://clemonslab.caltech.edu>).

22 The generalizability of the model is remarkable despite several known limitations. Using data from a
23 single study for training precludes including certain variables that empirically influence expression such

1 as the parameters corresponding to fusion tags and the context of the protein in an expression plasmid,
2 *e.g.* the 5' untranslated region, for which there was no variation in the Daley, Rapp, *et al.* dataset.
3 Moreover, using a simple proof-of-concept linear model allowed for a straightforward and robust
4 predictor; however, intrinsically it cannot be directly related to the biological underpinnings. While we
5 can extract some biological inference, a linear combination of sequence parameters does not explicitly
6 reflect the reality of physical limits for host cells. To some extent, constraint information is likely
7 encoded in the complex architecture of the underlying sequence space (*e.g.* through the genetic code,
8 TM prediction, RNA secondary structure analyses). Future statistical models that improve on these
9 limitations will likely hone predictive power and more intricately characterize the interplay of variables
10 that underlie membrane protein expression in *E. coli* and other systems.

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

23

1 **Acknowledgements**

2 We thank Daniel Daley and Thomas Miller's group for discussion, Yaser Abu-Mostafa and Yisong
3 Yue for guidance regarding machine learning, Niles Pierce for providing NUPACK source code⁴⁰, and
4 Welison Floriano and Naveed Near-Ansari for maintaining local computing resources. We thank James
5 Bowie, Michiel Niesen, Stephen Marshall, Thomas Miller, Reid van Lehn, and Tom Rapoport for
6 critical reading of the manuscript. Models and analyses are possible thanks to raw experimental data
7 provided by Daniel Daley and Mikaela Rapp¹⁹; Nir Fluman²⁰; Edda Kloppmann, Brian Kloss, and
8 Marco Punta from NYCOMPS^{2,3}; Pikyee Ma²⁵; Renaud Wagner²⁷; and Florent Bernaudat³¹. We
9 acknowledge funding from an NIH Pioneer Award to WMC (5DP1GM105385); a Benjamin M. Rosen
10 graduate fellowship, a NIH/NRSA training grant (5T32GM07616), and a NSF Graduate Research
11 fellowship to SMS; and an Arthur A. Noyes Summer Undergraduate Research Fellowship to NJ.
12 Computational time was provided by Stephen Mayo and Douglas Rees. This work used the Extreme
13 Science and Engineering Discovery Environment (XSEDE), which is supported by National Science
14 Foundation grant number ACI-1053575⁴¹.

15

16 **Author Contributions**

17 S.M.S., A.M., and W.M.C. conceived the project. S.M.S. developed the approach. S.M.S., A.M., and
18 N.J. compiled sequence and experimental data. N.J. created code to demonstrate feasibility. S.M.S.
19 performed all published calculations. S.M.S. and WMC wrote the manuscript.

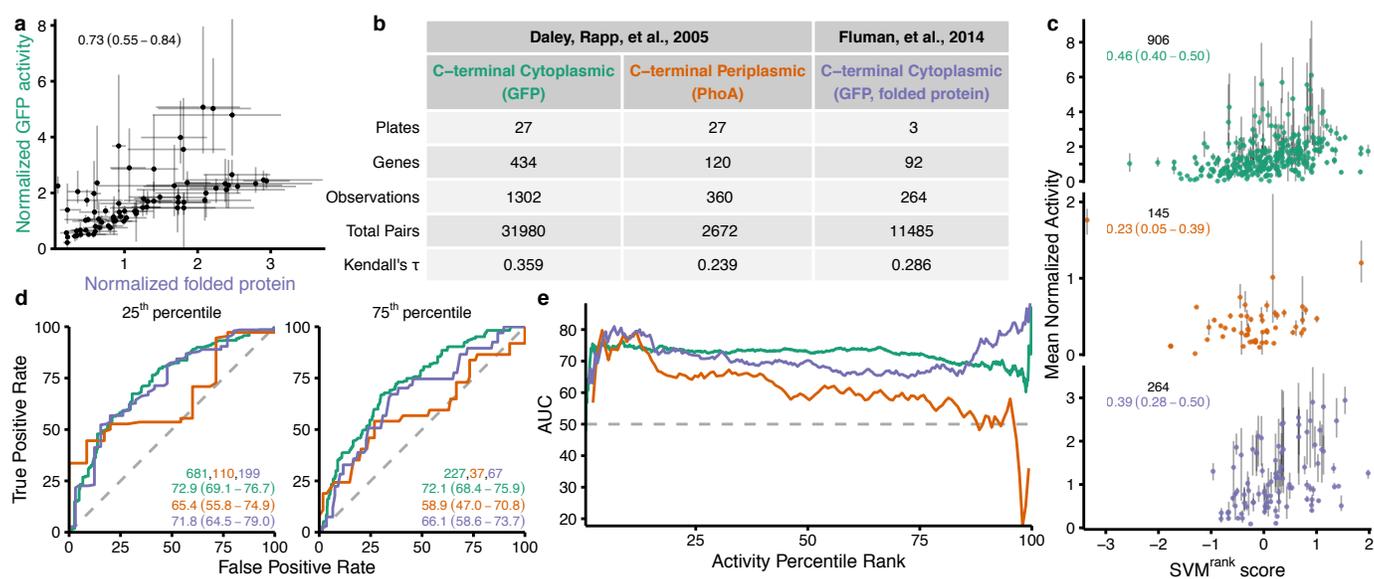
20

21 **Author Information**

22 The authors declare no competing financial interests.

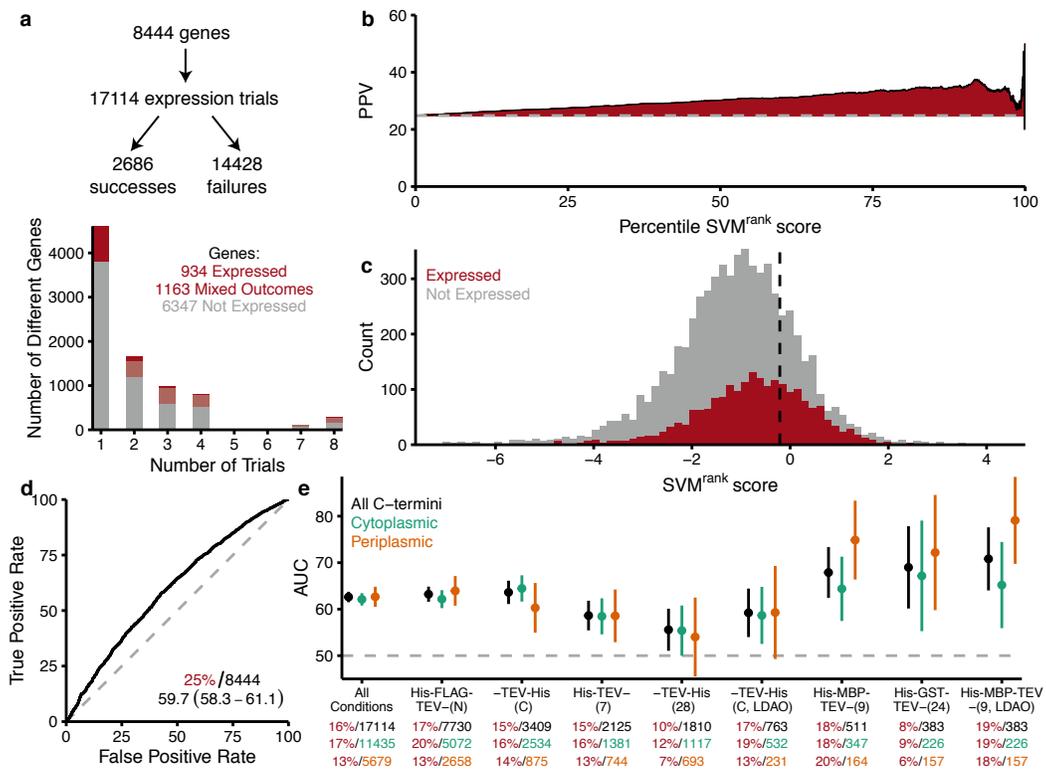
23 Correspondence and requests for materials should be addressed to clemons@caltech.edu.

Figure 1



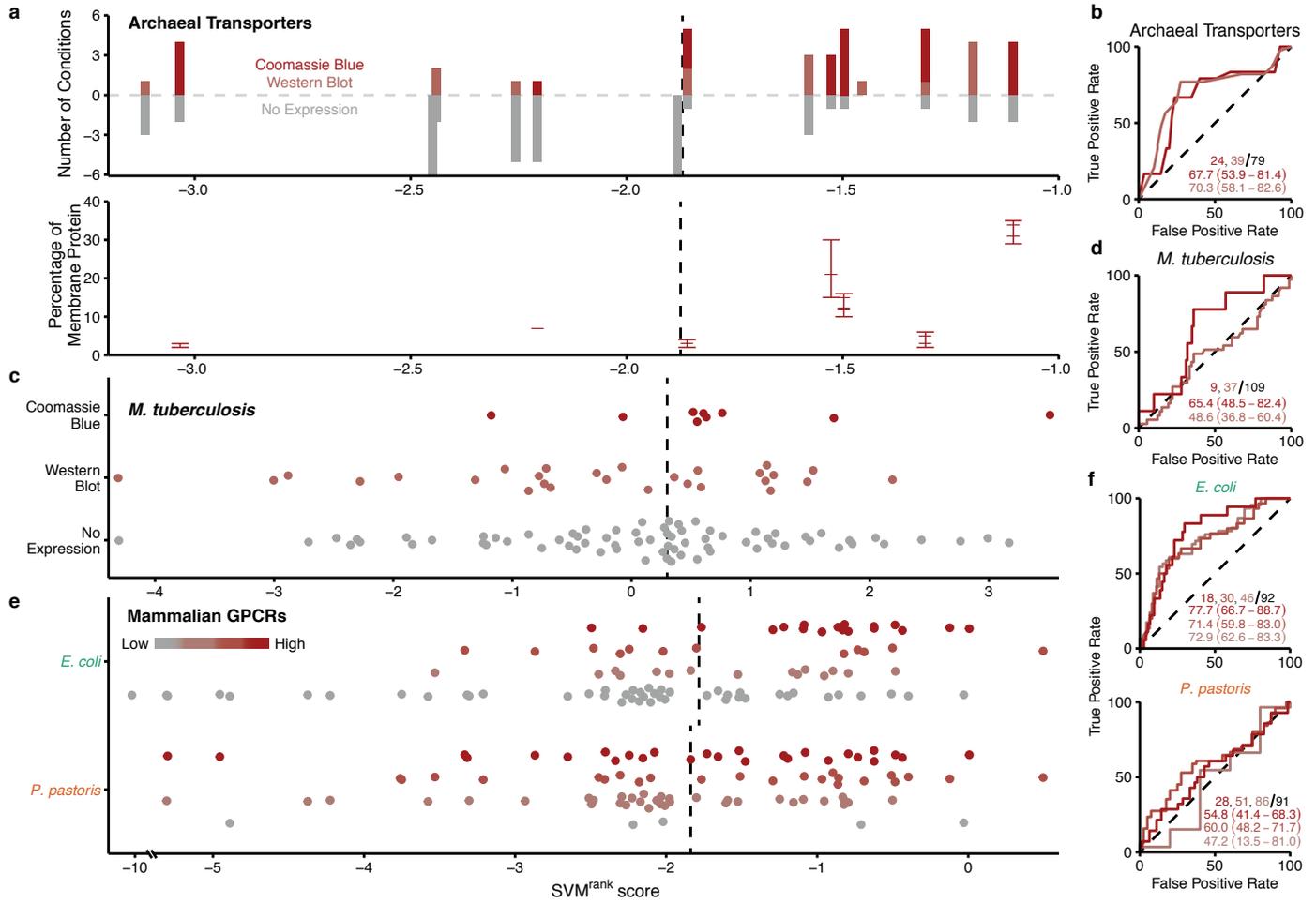
1 **Figure 1.** Training performance. **a**, A comparison of GFP activity¹⁹ with folded protein²⁰ where each
2 point represents the mean for a given gene tested in both works, and error bars plot the extrema.
3 Spearman's rank correlation coefficient ρ and 95% confidence interval (CI)⁴² are shown. **b**, Plates are
4 the number of independent sets of measurements within which expression levels can be reliably
5 compared. Genes are the number of proteins for which the C-terminus was reliably ascertained¹⁹.
6 Observations are the total number of expression data points accessible. Total pairs are the number of
7 comparable expression measurements (*i.e.* those within a single plate). Kendall's τ is the metric
8 maximized by the training process (See Methods 4b). The color of the column heading identifying each
9 experimental set is retained throughout the figure. **c**, Agreement against the normalized outcomes
10 plotted as the mean activity (see Methods 5 for definition) versus the score with error bars providing the
11 extent of observed activities (Spearman's ρ and 95% CI noted). **d**, Illustrative ROCs for thresholds at
12 25th and 75th percentile in activity with the number of positive outcomes at that threshold, the AUC, and
13 95% CI⁴³ indicated. **e**, The AUC of the ROC at every possible activity threshold.

Figure 2



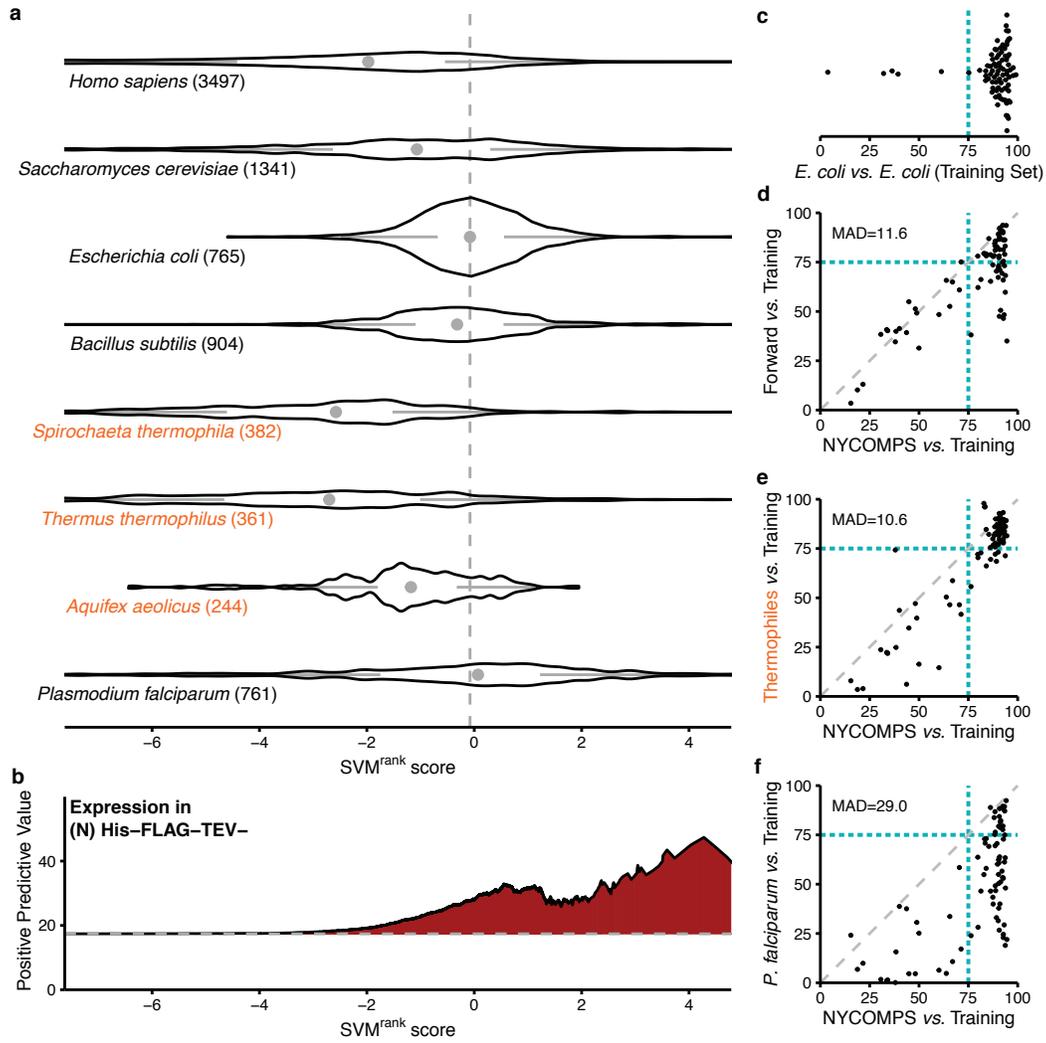
1 **Figure 2.** Success of the model against outcomes from NYCOMPS. **a**, An overview of the NYCOMPS
2 outcomes and a plot of the number of conditions tested per gene with outcomes highlighted. **b**, The PPV
3 plotted for each percentile SVM^{rank} score, *e.g.* 75 on the x-axis indicates the PPV for the top 25% of
4 genes based on score. The grey dashed line shows the ~25% overall success rate of the NYCOMPS
5 experimental outcomes. **c**, Histograms of the total count of proteins at a given SVM^{rank} score colored by
6 NYCOMPS-determined outcomes. **d**, ROC curve, positive (red) and total (black) counts, and AUC
7 values with 95% CI. The grey dashed line shows the performance of a completely random predictor
8 (AUC=50). **e**, The AUCs for all trials together (left) followed by outcomes in individual plasmid and
9 solubilization conditions (DDM except LDAO where noted) along with 95% CI (numerically in Table
10 S2). Performances are also split by predicted C-terminal localization²⁴. Overall positive percentage (red)
11 and total number of outcomes within each group is noted below the axis.

Figure 3



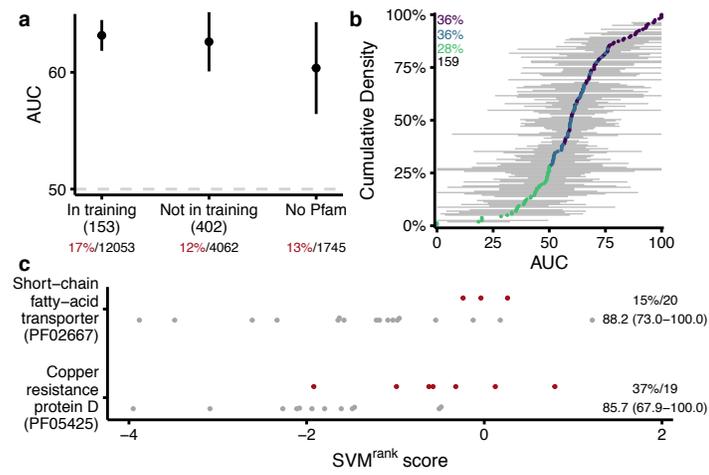
1 **Figure 3.** Success of the model against a variety of small scale outcomes. For each set, vertical lines
2 indicate the median SVM^{rank} score. ROCs along with AUCs and 95% CI as well as the total number of
3 positives for the given threshold (red hues) along with the total outcomes (black) are presented. In each
4 curve, increasing expression thresholds as defined by the original publication are displayed as deeper
5 red. **a**, The expression of archaeal transporters in up to 6 trials. Top, positive expression count is plotted
6 above the dashed line and negative outcomes below the line. Bottom, from the same work, the
7 expression of proteins detected by Coomassie Blue²⁵. **b**, ROC curves for each positive threshold (*i.e.*
8 Coomassie Blue or Western Blot) from trials in **a**. **c**, Experimental expression of *M. tuberculosis*
9 membrane proteins plotted based on outcomes. **d**, ROC curves for each possible threshold from trials in
10 **c**. **e**, Mammalian GPCR expression in either *E. coli* (top) or *P. pastoris* (bottom). **f**, ROC curves for each
11 possible threshold from trials in **e**.

Figure 4



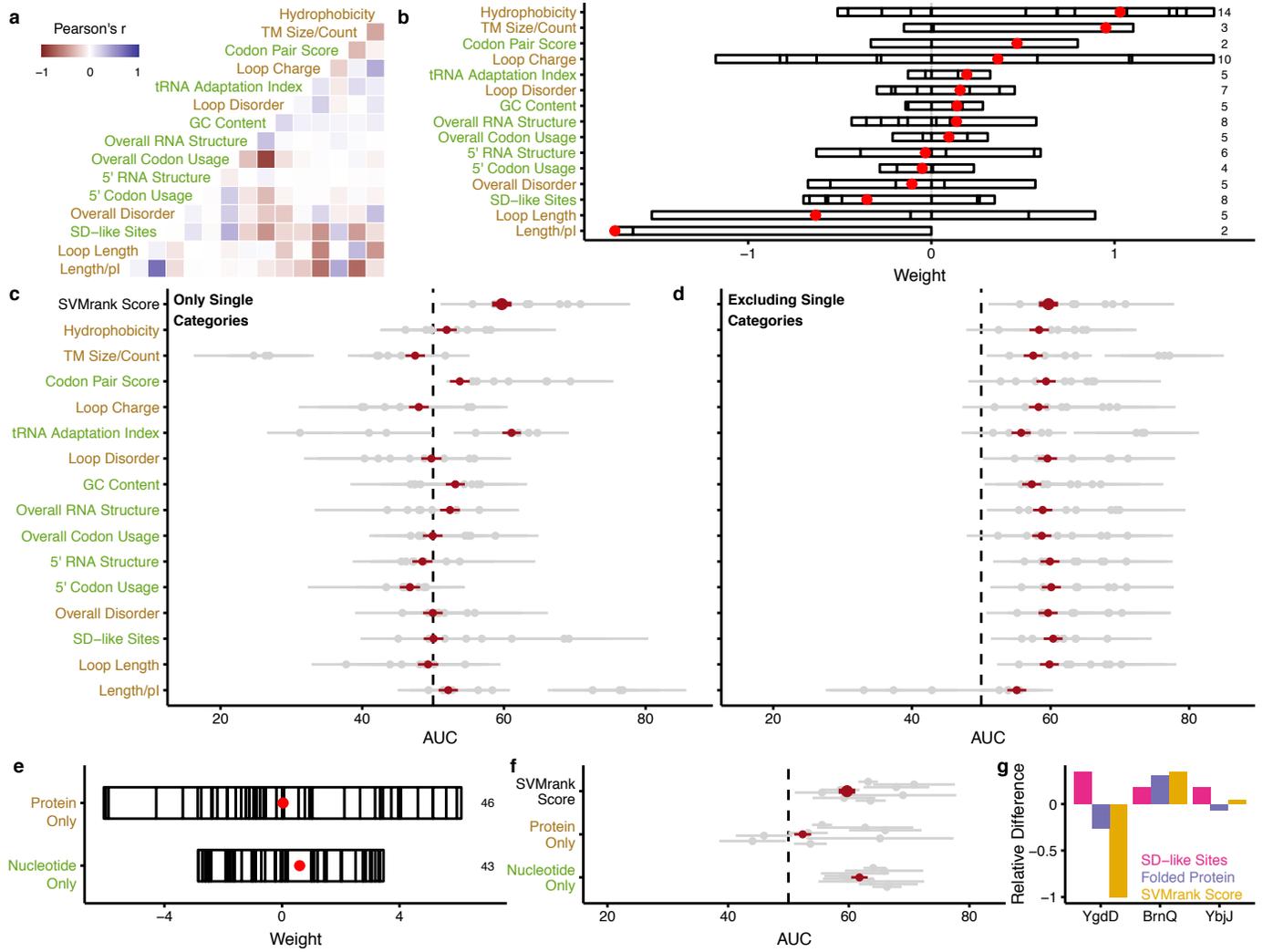
1 **Figure 4.** Forward predictions of membrane protein expression for various genomes. **a**, Calculated
2 scores for proteins from a variety of genomes (count in parentheses; complete set provided in Figure
3 S2a) plotted as contours of kernel density estimates of the number of proteins at a given score.
4 Amplitude is only relative within a genome. The dot indicate the median, and the lines depict quantities
5 of an analogous Tukey boxplot^{44,45}. The vertical line shows the median score in *E. coli* to provide
6 context for other distributions. **b**, PPV of the model within the most tested NYCOMPS condition. **c**,
7 Distribution of overlap coefficients (see Methods 7) for each sequence parameter comparing the entire
8 *E. coli* membrane proteome vs. the training set from *E. coli*. The dashed line provides a threshold
9 separating the cluster of highly-related features from those with lower overlap. **d-f**, A comparison of
10 overlap coefficients with the training set between NYCOMPS and **d**, all forward predictions (Figure
11 S2a), **e**, thermophilic genomes (orange), or **f**, *P. falciparum*. Mean Absolute Deviation is indicated for
12 each plot.

Figure 5



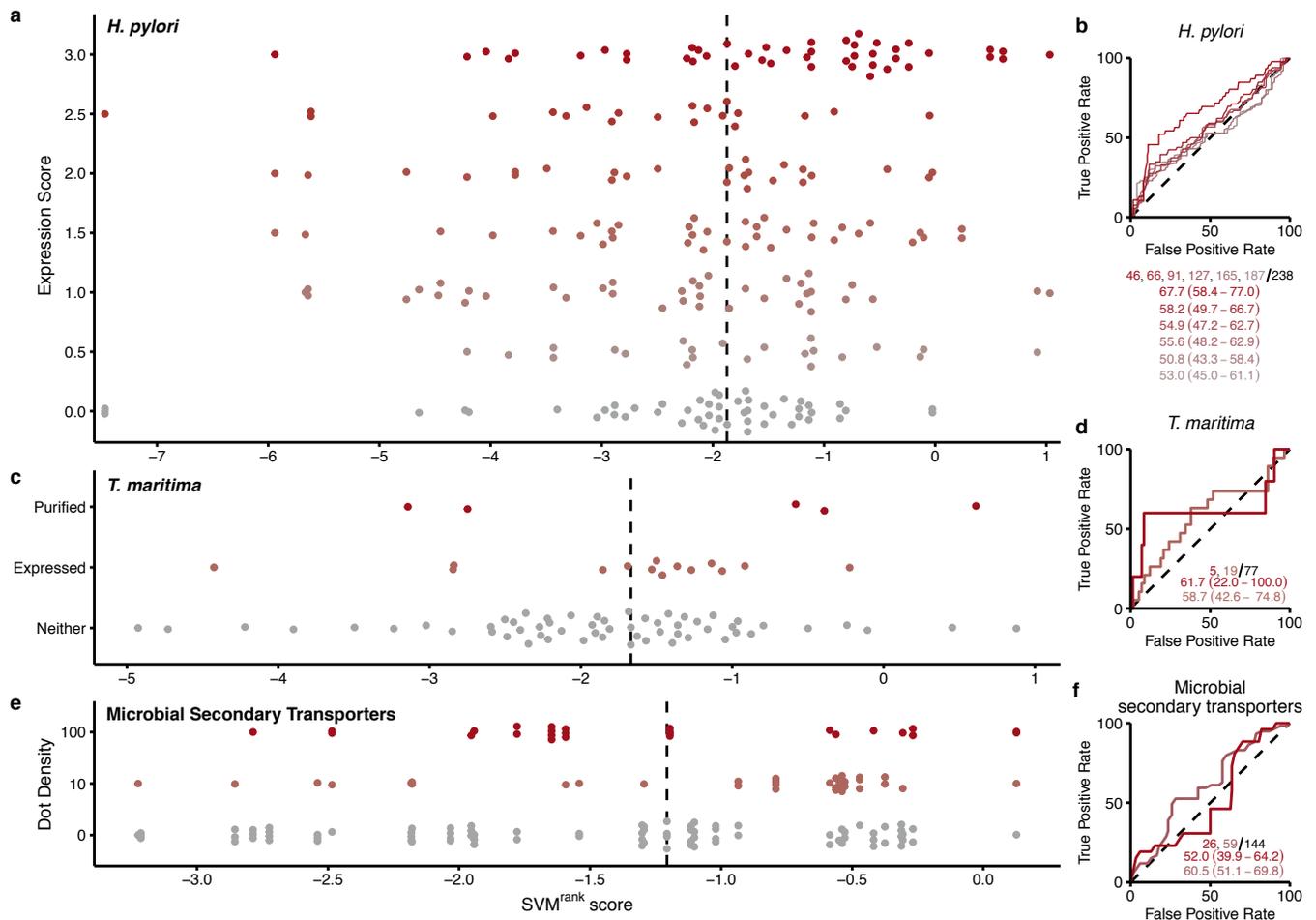
1 **Figure 5.** Model performance across protein families **a**, The NYCOMPS dataset is split by those
2 proteins with a Pfam found either in the training set, not in the training set or those without a Pfam
3 (Pfam counts in parentheses). The AUC and 95% confidence interval for each set is plotted with the
4 positive rate (red) and number of trials (black) indicated below the x-axis. **b**, For each family, the AUC
5 across all outcomes is plotted arranged in order of the value, an empirical cumulative distribution
6 function, with horizontal bars indicating the 95% confidence interval. The color indicates the
7 significance of the prediction within the family: purple, predictive at 95% confidence, blue, predictive
8 but not at 95% confidence, green, not predictive. The size of each significance group and total number
9 of families (black) are indicated on the plot. **c**, Outcomes for specific protein families. Each was only
10 tested in a single condition (N). The overall positive percentage within the group, total number of
11 outcomes, and AUC with 95% CI is labelled to the right.

Figure 6



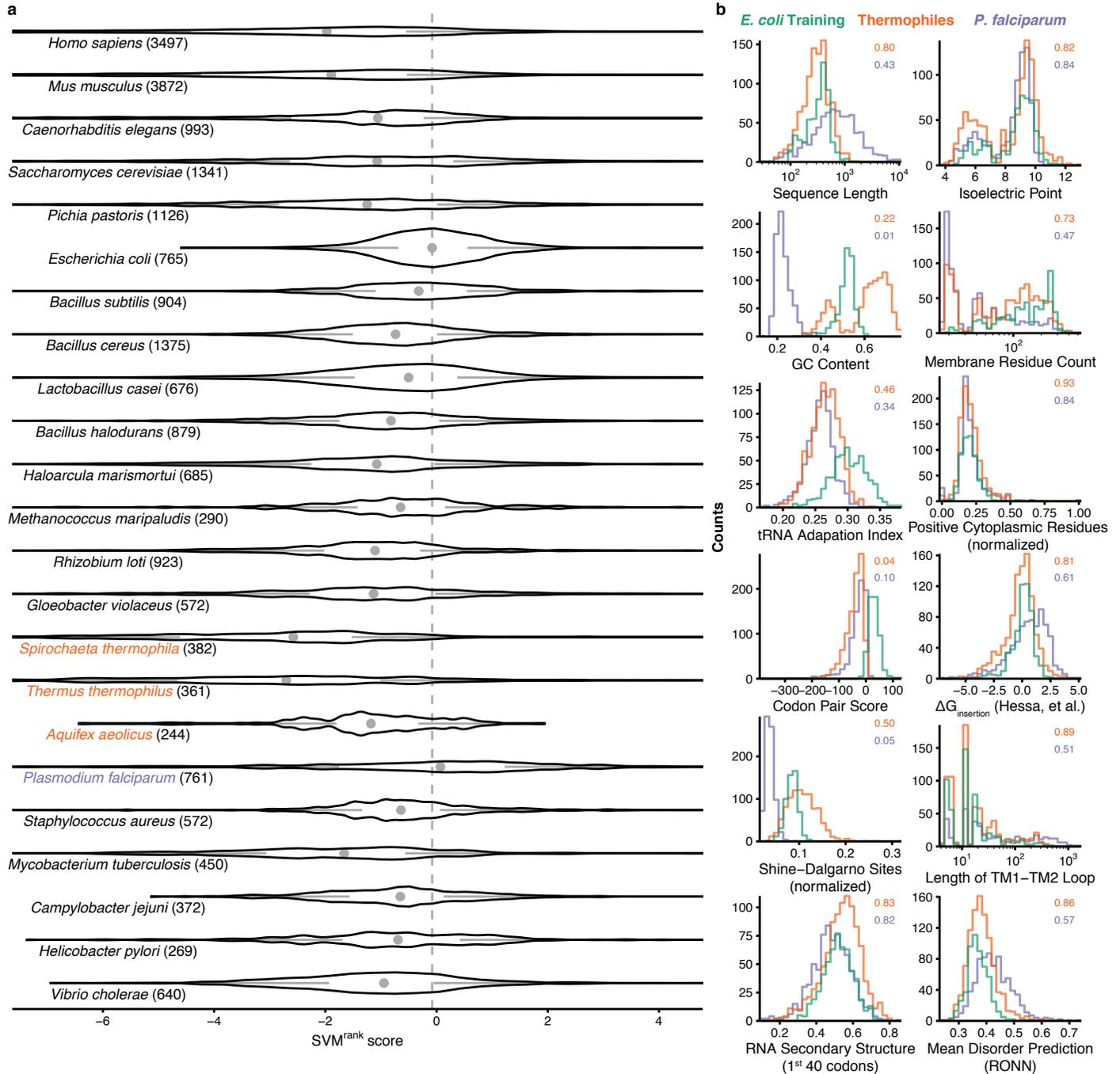
1 **Figure 6.** Feature contributions to the model **a**, Pearson correlation coefficients between feature
2 categories are shown. Feature labels are green for protein-sequence derived and brown for nucleotide-
3 sequence derived. **b**, Total weight for each category is represented as a bar. The contribution of each
4 feature to the category is shown by partitioning the bar. The red dot indicates the total sum of weights
5 within the category. **c**, The AUC and 95% confidence interval when predicting with the entire model
6 (SVM^{rank} score) or single category specified on the NYCOMPS dataset. Red shows the outcome of
7 predicting at the level of individual genes (Figure 2b-d) and grey show the outcome within each vector
8 individually (as in Figure 2e). **d**, The AUC and 95% confidence interval when predicting with the entire
9 model (SVM^{rank} score) or by excluding single category specified on the NYCOMPS dataset. **e**, As **b**, but
10 classifying features by the type of sequence they are calculated from. **f**, The AUC and 95% confidence
11 intervals using only protein or nucleotide features. **g**, Relative difference in SD-like sites (green),
12 expression (purple), and SVM^{rank} score (yellow) between wild-type and mutants with silent mutations.
13 See Methods 7 for further detail.

Supplementary Figure 1



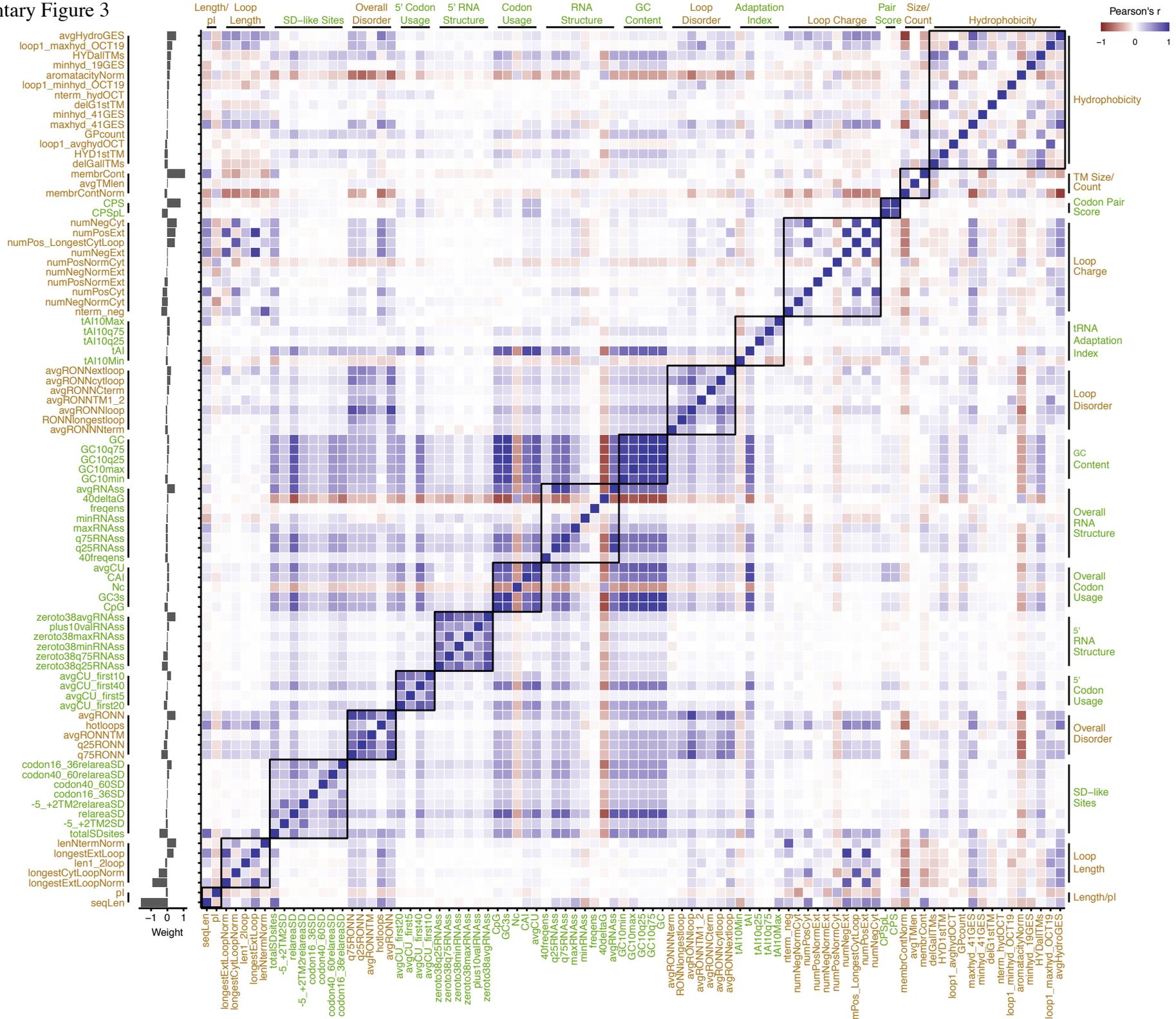
1 **Supplementary Figure 1.** Additional small-scale predictions and outcomes. **a**, Experimental expression
2 of 116 *H. pylori* membrane proteins in *E. coli* in at most 3 vectors (238 trials) scored as either a 1, 2, or
3 3 from the outcome of a dot blot as well as Coomassie Staining of an SDS-PAGE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

Supplementary Figure 2



1 **Supplementary Figure 2.** Complete set of forward predictions. **a**, Extended from Figure 4c, the full
2 complement of score distributions calculated by genome is plotted and arranged to accentuate similar
3 features by physiology, *e.g.* growth condition, or scientific interest, *e.g.* pathogenic. Raw scores along
4 with sequence identifiers are available in the Table S3. **b**, Histograms of representative sequence
5 parameters between the training data set (green), thermophiles (orange), and *P. falciparum* (purple).
6 Values for sequence parameter overlap coefficients derived from kernel density estimates (Methods 7)
7 versus the *E. coli* training data are included. See Table S1 for parameter descriptions.

Supplementary Figure 3



1 **Supplementary Figure 3.** Complete set of feature correlations and their individual contributions to the
2 model. Features are ordered first by category (as in Figure 5) and then by weight (grey bars). Labels are
3 green for protein-sequence derived and brown for nucleotide-sequence derived features. Pearson
4 correlation coefficient between each pair of features across the NYCOMPS dataset is plotted (right). See
5 Table S1 for a detailed description of each feature. Feature categories are overlaid as square boxes and
6 indicated by black bars on the top, left, and right of the correlation matrix.

1 **Table S1 (separate file)**

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

4

5 **Table S2 (separate file)**

6 AUC values for the NYCOMPS dataset. AUC values and 95% confidence intervals are presented in
7 summary, by expression condition, and by predicted C-terminal localization as well as for SVM^{rank}
8 scores calculated without the most computationally expensive RNA secondary structure calculation (as
9 in Figure 4).

10

11 **Table S3 (separate file)**

12 Full list of predicted membrane proteins from the various genomes in Supplementary Figure 2 with
13 corresponding identifiers, descriptions, Pfam families, coding sequences, and SVM^{rank} scores. This data
14 is available in an interactive format at clemonslab.caltech.edu.

15

16 **Table S4 (separate file)**

17 Predictive performances of the model across protein families tested by NYCOMPS summarized in
18 Figure 5. This data is available in an interactive format at clemonslab.caltech.edu.

1 **Materials and Methods**

2 Sequence mapping & retrieval and feature calculation was performed in Python 2.7⁴⁶ using
3 BioPython⁴⁷ and NumPy⁴⁸; executed and consolidated using Bash (shell) scripts; and parallelized where
4 possible using GNU Parallel⁴⁹. Data analysis and presentation was done in R⁵⁰ within RStudio⁵¹ using
5 magrittr⁵², plyr⁵³, dplyr⁵⁴, asbio⁵⁵, and datamart⁵⁶ for data handling; ggplot2⁵⁷, ggbeeswarm⁵⁸, GGally⁵⁹,
6 gridExtra⁶⁰, cowplot⁶¹, scales⁶², viridis⁶³, and RColorBrewer^{64,65} for plotting; multidplyr⁶⁶ with parallel⁵⁰
7 and foreach⁶⁷ with iterators⁶⁸ and doMC⁶⁹/doParallel⁷⁰ for parallel processing; and roxygen2⁷¹ for code
8 organization and documentation as well as other packages as referenced.

9

10 1. Collection of data necessary for learning and evaluation

11 *E. coli* Sequence Data – The nucleotide sequences from ¹⁹ were deduced by reconstructing forward and
12 reverse primers (*i.e.* ~20 nucleotide stretches) from each gene in Colibri (based on EcoGene 11), the
13 original source cited and later verified these primers against an archival spreadsheet provided directly by
14 Daniel Daley (personal communication). To account for sequence and annotation corrections made to
15 the genome after Daley, Rapp, *et al.*'s work, these primers were directly used to reconstruct the
16 amplified product from the most recent release of the *E. coli* K-12 substr. MG1655 genome⁷² (EcoGene
17 3.0; U00096.3). Although Daniel Daley mentioned that raw reads from the Sanger sequencing runs may
18 be available within his own archives, it was decided that the additional labor to retrieve this data and
19 parse these reads would not significantly impact the model. The deduced nucleotide sequences were
20 verified against the protein lengths given in Table S1 from ¹⁹. The plasmid library tested in ²⁰ was
21 provided by Daniel Daley, and those sequences are taken to be the same.

22

1 *E. coli Training Data* – The preliminary results using the mean-normalized activities echoed the
2 findings of ¹⁹ that these do not correlate with sequence features either in the univariate sense (many
3 simple linear regressions, Table S1¹⁹) or a multivariate sense (multiple linear regression, data not
4 shown). This is presumably due to the loss of information regarding variability in expression level for
5 given genes or due to the increase in variance of the normalized quantity (See Methods 4a) due to the
6 normalization and averaging procedure. Daniel Daley and Mikaela Rapp provided spreadsheets of the
7 outcomes from the 96-well plates used for their expression trials and sent scanned copies of the readouts
8 from archival laboratory notebooks where the digital data was no longer accessible (personal
9 communication). Those proteins without a reliable C-terminal localization (as given in the original
10 work) or without raw expression outcomes were not included in further analyses.

11 Similarly, Nir Fluman also provided spreadsheets of the raw data from the set of three expression
12 trials performed in ²⁰.

13
14 *New York Consortium on Membrane Protein Structure (NYCOMPS) Data* – Brian Kloss, Marco Punta,
15 and Edda Kloppman provided a dataset of actions performed by the NYCOMPS center including
16 expression outcomes in various conditions.^{2,3} The protein sequences were mapped to NCBI GenInfo
17 Identifier (GI) numbers either via the Entrez system⁷³ or the Uniprot mapping service⁷⁴. Each GI number
18 was mapped to its nucleotide sequence via a combination of the NCBI Elink mapping service and the
19 “coded_by” or “locus” tags of Coding Sequence (CDS) features within GenBank entries. Though a
20 custom script was created, a script from Peter Cock on the BioPython listserv to do the same task via a
21 similar mapping mechanism was found⁷⁵. To confirm all the sequences, the TargetTrack⁷⁶ XML file was
22 parsed for the internal NYCOMPS identifiers and compared for sequence identity to those that had been

1 mapped using the custom script; 20 (less than 1%) of the sequences had minor inconsistencies and were
2 manually replaced.

3

4 *Archaeal transporters Data* – The locus tags (“Gene Name” in Table 1) were mapped directly to the
5 sequences and retrieved from NCBI²⁵. Pikyee Ma and Margarida Archer clarified questions regarding
6 their work to inform the analysis.

7

8 *GPCR Expression Data* – Nucleotide sequences were collected by mapping the protein identifiers given
9 in Table 1 from ²⁷ to protein GIs via the Uniprot mapping service⁷⁴ and subsequently to their nucleotide
10 sequences via the custom mapping script described above (see NYCOMPS). The sequence length and pI
11 were validated against those provided. Renaud Wagner assisted in providing the nucleotide sequences
12 for genes whose listed identifiers were unable to be mapped and/or did not pass the validation criteria as
13 the MeProtDB (the sponsor of the GPCR project) does not provide a public archive.

14

15 *Helicobacter pylori Data* – Nucleotide sequences were retrieved by mapping the locus tags given in
16 Supplemental Table 1 from ²⁸ to locus tags in the Jan 31, 2014 release of the *H. pylori* 26695 genome
17 (AE000511.1). To verify sequence accuracy, sequences whose molecular weight matched that given by
18 the authors were accepted. Those that did not match, in addition to the one locus tag that could not be
19 mapped to the Jan 31, 2014 genome version, were retrieved from the Apr 9, 2015 release of the genome
20 (NC_000915.1). Both releases are derived from the original sequencing project⁷⁷. After this curation, all
21 mapped sequences matched the reported molecular weight.

22 In this data set, expression tests were performed in three expression vectors and scored as 1, 2, or 3.
23 Two vectors were scored via two methods. For these two vectors, the two scores were averaged to give a

1 single number for the condition making them comparable to the third vector while yielding 2 additional
2 thresholds (1.5 and 2.5) result in the 5 total curves shown (Figure S1b).

3

4 *Mycobacterium tuberculosis* Data – The authors note using TubercuList through GenoList⁷⁸, therefore,
5 nucleotide sequences were retrieved from the archival website based on the original sequencing
6 project⁷⁹. The sequences corresponding to the identifiers and outcomes in Table 1 from ²⁶ were validated
7 against the provided molecular weight .

8

9 *Secondary Transporter Data* – GI Numbers given in Table 1 from ³⁰ were matched to their CDS entries
10 using the custom mapping script described above (see NYCOMPS). Only expression in *E. coli* with
11 IPTG-inducible vectors was considered.

12

13 *Thermotoga maritima* Data – Gene names given in Table 1⁸⁰ were matched to CDS entries in the Jan
14 31, 2014 release of the *Thermotoga maritima* MSB8 genome (AE000512.1), a revised annotation of the
15 original release⁸¹. The sequence length and molecular weight were validated against those provided.

16

17 2. Calculation of sequence features

18 Based on experimental analyses and anecdotal evidence, approximately 105 different protein and
19 nucleotide sequence features thought to be relevant to expression were identified and calculated for each
20 protein using custom code together with published software (codonW⁸², tAI⁸³, NUPACK⁴⁰, Vienna
21 RNA⁸⁴, Codon Pair Bias⁸⁵, Disembl¹⁸, and RONN⁸⁶). Relative metrics (*e.g.* codon adaptation index) are
22 calculated with respect to the *E. coli* K-12 substr. MG1655⁷² quantity. The octanol-water partitioning⁸⁷,
23 GES hydrophobicity⁸⁸, ΔG of insertion¹⁶ scales were employed as well. Transmembrane segment

1 topology was predicted using Phobius Constrained for the training data and Phobius for all other
2 datasets²⁴. Two RNA secondary structure metrics were prompted in part by ¹⁴. Several features were
3 obtained by averaging per-site metrics (*e.g.* per-residue RONN3.2 disorder predictions) in windows of a
4 specified length. Windowed tAI metrics are calculated over *all* 30 base windows (not solely over 10
5 codon windows). Table S1 lists a description of each feature. Features are calculated solely from a gene
6 of interest excluding portions of the ORFs such as linkers and tags derived from the plasmid backbone
7 employed (future work will explore contributions of these elements).

8

9 3. Preparation for model learning

10 Calculated sequence features for the membrane proteins in the *E. coli* dataset as well as raw activity
11 measurements, *i.e.* each 96-well plate, were loaded into R. As is best practice in using Support Vector
12 Machines, each feature was “centered” and “scaled” where the mean value of a given feature was
13 subtracted from each data point and then divided by the standard deviation of that feature using
14 `preprocess`⁸⁹. As is standard practice, the resulting set was then culled for those features of near
15 zero-variance, over 95% correlation (Pearson’s r), and linear dependence (`nearZeroVar`,
16 `findCorrelation`, `findLinearCombos`)⁸⁹. In particular this procedure removed extraneous
17 degrees of freedom during the training process which carry little to no additional information with
18 respect to the feature space and which may over represent certain redundant features. Features and
19 outcomes for each list (“query”) were written into the SVM^{light} format using a modified
20 `svmlight.write`⁹⁰.

21 The final features were calculated for each sequence in the test datasets, prepared for scoring by
22 “centering” and “scaling” by the training set parameters via `preprocess`⁸⁹, and then written into
23 SVM^{light} format again using a modified `svmlight.write`.

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

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

8 The quantitative 96-well plate results¹⁹ that comprise our training data do not offer an absolute
9 expression metric valid over all plates—the top expressing proteins in one plate would not necessarily
10 be the best expressing within another. As such, this problem is suited for preference-ranking methods.
11 As a ranking problem, the response variable is the ordinal rank for each protein derived from its
12 overexpression relative to the other members of the same plate of expression trials. In other words, the
13 aim is to rank highly expressed proteins (based on numerous trials) at higher scores than lower
14 expressed proteins by fitting against the order of expression outcomes from each constituent 96-well
15 plate.

16 *b.* As the first work of this kind, the aim was to employ the simplest framework necessary taking in
17 account the considerations above. The method chosen computes all valid pairwise classifications (*i.e.*
18 within a single plate) transforming the original ranking problem into a binary classification problem.
19 The algorithm outputs a score for each input by minimizing the number of swapped pairs thereby
20 maximizing Kendall's τ ⁹¹. For example, consider the following data generated via context A
21 $(X_{A,1}, Y_{A,1}), (X_{A,2}, Y_{A,2})$ and B $(X_{B,1}, Y_{B,1}), (X_{B,2}, Y_{B,2})$ where observed response follows as index i , *i.e.*
22 $Y_n < Y_{n+1}$. Binary classifier $f(X_i, X_j)$ gives a score of 1 if an input pair matches its ordering criteria and
23 -1 if not, *i.e.* $Y_i < Y_j$:

1
$$f(X_{A,1}, X_{A,2}) = 1; f(X_{A,2}, X_{A,1}) = -1$$

2
$$f(X_{B,1}, X_{B,2}) = 1; f(X_{B,2}, X_{B,1}) = -1$$

3
$$f(X_{A,1}, X_{B,2}), f(X_{A,2}, X_{B,1}) \text{ are invalid}$$

4 Free parameters describing f are calculated such that those calculated orderings
5 $f(X_{A,1}), f(X_{A,2}) \dots; f(X_{B,1}), f(X_{B,2}) \dots$ most closely agree (overall Kendall's τ) with the observed
6 ordering Y_n, Y_{n+1}, \dots . In this sense, f is a pairwise Learning to Rank method.

7 Within this class of models, a linear preference-ranking Support Vector Machine was employed⁹².
8 To be clear, as an algorithm a preference-ranking SVM operates similarly to the canonical SVM binary
9 classifier. In the traditional binary classification problem, a linear SVM seeks the maximally separating
10 hyper-plane in the feature space between two classes, where class membership is determined by which
11 side of the hyper-plane points reside. For some n linear separable training examples $D =$
12 $\{(x_i) | x_i \in \mathbb{R}^d\}^n$ and two classes $y_i \in \{-1, 1\}$, a linear SVM seeks a mapping from the d -dimensional
13 feature space $\mathbb{R}^d \rightarrow \{-1, 1\}$ by finding two maximally separated hyperplanes $w \cdot x - b = 1$ and $w \cdot$
14 $x - b = -1$ 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
15 x_i with $y_i \in \{-1\}$. The feature weights correspond to the vector w , which is the vector perpendicular to
16 the separating hyperplanes, and are computable in $O(n \log n)$ implemented as part of the SVM^{rank}
17 software package, though in $O(n^2)$ ²². See ⁹² for an in-depth, technical discussion.

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

1 model is composed solely of a single weight for each feature; the tradeoff parameter C is only part of the
2 training process.

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

7

8 5. Quantitative Assessment of Predictive Performance

9 In generating a predictive model, one aims to enrich for positive outcomes while ensuring they do
10 not come at the cost of increased false positive diagnoses. This is formalized in Receiver Operating
11 Characteristic (ROC) theory (for a primer see ²³), where the true positive rate is plotted against the false
12 positive rate for all classification thresholds (score cutoffs in the ranked list). In this framework, the
13 overall ability of the model to resolve positive from negative outcomes is evaluated by analyzing the
14 area under an ROC curve (AUC) where $AUC_{\text{perfect}}=100\%$ and $AUC_{\text{random}}=50\%$ (percentage signs are
15 omitted throughout the text and figures). All ROCs are calculated through pROC⁹³ using the analytic
16 Delong method for AUC confidence intervals⁴³. Bootstrapped AUC CIs ($N = 10^6$) were precise to 4
17 decimal places suggesting that analytic CIs are valid for the NYCOMPS dataset.

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

21 *Training data* – The outcomes are quantitative (activity level), so each ROC is calculated by
22 normalizing within each dataset to the standard well subject to the discussion in 4a above (LepB for
23 PhoA, and InvLepB for GFP) (examples in Figure 1D) for each possible threshold, *i.e.* each normalized

1 expression value with each AUC plotted in Figure 1e. 95% confidence intervals of Spearman's ρ are
2 given by 10^6 iterations of a bias-corrected and accelerated (BCa) bootstrap of the data (Figure 1a,c)⁴².

3 *Large-scale* – ROCs were calculated for each of the expression classes (Figure 2e). Regardless of the
4 split, predictive performance is noted. The binwidth for the histogram was determined using the
5 Freedman-Diaconis rule, and scores outside the plotted range comprising <0.6% of the density were
6 implicitly hidden.

7 *Small-scale* – Classes can be defined in many different ways. To be principled about the matter, ROCs
8 for each possible cutoff are presented based on definitions from each publication (Figure 3b,d,f, Figure
9 S1b,d,f). See Methods 1 for any necessary details about outcome classifications for each dataset.

10

11 6. Feature Weights

12 Weights for the learned SVM are pulled directly from the model file produced by SVM^{light} and are given
13 in Table S1 after normalizing to the mean value.

14

15 7. Forward Predictions

16 *Data collection* – We selected several genomes for comparison as shown in Figure 4, Figure S2a, and
17 Table S3. Coding sequences of membrane proteins from human and mouse genomes were gathered by
18 mapping Uniprot identifiers of proteins noted to have at least one transmembrane segment by Uniprot⁷⁴
19 to Ensembl (release 82) coding sequences⁹⁴ via Biomart.⁹⁵ *C. elegans* coding sequences were similarly
20 mapped via Uniprot but to WormBase coding sequences⁹⁶ also via Biomart. *S. cerevisiae* strain S288C
21 coding sequences⁹⁷ were retrieved from the Saccharomyces Genome Database. *P. pastoris* strain GS115
22 coding sequences⁹⁸ were retrieved from the DOE Joint Genome Institute (JGI) Genome Portal⁹⁹. Those
23 sequences without predicted²⁴ TMs were excluded from subsequent analyses. Microbial sequences were

1 gathered via a custom, in-house database populated with data compiled primarily from Pfam³⁶, DOE JGI
2 Integrated Microbial Genomes¹⁰⁰, and the Microbial Genome Database¹⁰¹.

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

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

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

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

15 As written f_i and g_i are probability densities, they need to be approximated before calculating Δ_i and
16 are done so via kernel density estimates (KDE) of the observed samples $[x_1^f, \dots, x_n^f]$ and $[x_1^g, \dots, x_n^g]$
17 using a nonparametric, locally adaptive method allowing for variable bandwidth smoothing
18 implemented in LocFit¹⁰² (adpen= $2\sigma^2$) providing \hat{f}_i and \hat{g}_i . The distribution overlap Δ_i is evaluated
19 over a grid of 2^{13} equally spaced points over the range of f_i and g_i .

20 *Shine-Dalgarno-like mutagenesis* – Folded protein is quantified by densitometry measurement^{103,104} of
21 the relevant band in Figure 6 of ²⁰. Relative difference is calculated as is standard:

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

1

2 8. Availability

3 All analysis is documented in a series of R notebooks¹⁰⁵ available openly at [github.com/clemlab/ml-](https://github.com/clemlab/ml-ecoli-svmrank)
4 [ecoli-svmrank](https://github.com/clemlab/ml-ecoli-svmrank). These notebooks provide fully executable instructions for the reproduction of the
5 analyses and the generation of figures and statistics in this study. The ranking engine is available as a
6 web service at clemonslab.caltech.edu. Additional code is available upon request.

References

1. White, S. H. Biophysical dissection of membrane proteins. *Nature* **459**, 344–346 (2009).
2. Punta, M. *et al.* Structural genomics target selection for the New York consortium on membrane protein structure. *J. Struct. Funct. Genomics* **10**, 255–268 (2009).
3. Love, J. *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* **11**, 191–199 (2010).
4. Berman, H. M. *et al.* The Protein Data Bank. *Nucleic Acids Res.* **28**, 235–242 (2000).
5. Bill, R. M. *et al.* Overcoming barriers to membrane protein structure determination. *Nat. Biotechnol.* **29**, 335–340 (2011).
6. Nørholm, M. H. H. *et al.* Manipulating the genetic code for membrane protein production: what have we learnt so far? *Biochim. Biophys. Acta* **1818**, 1091–1096 (2012).
7. Wagner, S. *et al.* Tuning Escherichia coli for membrane protein overexpression. *Proc. Natl. Acad. Sci. U. S. A.* **105**, 14371–14376 (2008).
8. Lewinson, O., Lee, A. T. & Rees, D. C. The funnel approach to the precrystallization production of membrane proteins. *J. Mol. Biol.* **377**, 62–73 (2008).
9. Marshall, S. S. *et al.* A Link between Integral Membrane Protein Expression and Simulated Integration Efficiency. *Cell Rep.* **16**, 2169–2177 (2016).
10. Sarkar, C. A. *et al.* Directed evolution of a G protein-coupled receptor for expression, stability, and binding selectivity. *Proc. Natl. Acad. Sci. U. S. A.* **105**, 14808–14813 (2008).
11. Schlinkmann, K. M. *et al.* 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.* **109**, 9810–9815 (2012).
12. Seppälä, S., Slusky, J. S., Lloris-Garcerá, P., Rapp, M. & von Heijne, G. Control of membrane protein topology by a single C-terminal residue. *Science* **328**, 1698–1700 (2010).
13. Van Lehn, R. C., Zhang, B. & Miller, T. F. Regulation of multispansing membrane protein topology via post-translational annealing. *eLife* **4**, (2015).
14. Goodman, D. B., Church, G. M. & Kosuri, S. Causes and effects of N-terminal codon bias in bacterial genes. *Science* **342**, 475–479 (2013).
15. Mirzadeh, K. *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
16. Hessa, T. *et al.* Molecular code for transmembrane-helix recognition by the Sec61 translocon. *Nature* **450**, 1026–1030 (2007).
17. Heijne, G. The distribution of positively charged residues in bacterial inner membrane proteins correlates with the trans-membrane topology. *EMBO J.* **5**, 3021–3027 (1986).
18. Linding, R. *et al.* Protein disorder prediction: implications for structural proteomics. *Structure* **11**, 1453–1459 (2003).
19. Daley, D. O. *et al.* Global topology analysis of the Escherichia coli inner membrane proteome. *Science* **308**, 1321–1323 (2005).
20. Fluman, N., Navon, S., Bibi, E. & Pilpel, Y. mRNA-programmed translation pauses in the targeting of E. coli membrane proteins. *eLife* **3**, (2014).
21. Geertsma, E. R., Groeneveld, M., Slotboom, D.-J. & Poolman, B. Quality control of overexpressed membrane proteins. *Proc. Natl. Acad. Sci. U. S. A.* **105**, 5722–5727 (2008).
22. Tsochantaridis, I., Joachims, T., Hofmann, T. & Altun, Y. Large Margin Methods for Structured and Interdependent Output Variables. *J. Mach. Learn. Res.* **6**, 1453–1484 (2005).

- 1 23. Swets, J. A., Dawes, R. M. & Monahan, J. Better decisions through science. *Sci. Am.* **283**, 82–87
2 (2000).
- 3 24. Käll, L., Krogh, A. & Sonnhammer, E. L. L. A combined transmembrane topology and signal
4 peptide prediction method. *J. Mol. Biol.* **338**, 1027–1036 (2004).
- 5 25. Ma, P. *et al.* An efficient strategy for small-scale screening and production of archaeal membrane
6 transport proteins in *Escherichia coli*. *PloS One* **8**, e76913 (2013).
- 7 26. Korepanova, A. *et al.* Cloning and expression of multiple integral membrane proteins from
8 *Mycobacterium tuberculosis* in *Escherichia coli*. *Protein Sci.* **14**, 148–158 (2005).
- 9 27. Lundstrom, K. *et al.* Structural genomics on membrane proteins: comparison of more than 100
10 GPCRs in 3 expression systems. *J. Struct. Funct. Genomics* **7**, 77–91 (2006).
- 11 28. Psakis, G. *et al.* Expression screening of integral membrane proteins from *Helicobacter pylori*
12 26695. *Protein Sci.* **16**, 2667–2676 (2007).
- 13 29. Dobrovetsky, E. *et al.* High-throughput production of prokaryotic membrane proteins. *J. Struct.*
14 *Funct. Genomics* **6**, 33–50 (2005).
- 15 30. Surade, S. *et al.* Comparative analysis and ‘expression space’ coverage of the production of
16 prokaryotic membrane proteins for structural genomics. *Protein Sci.* **15**, 2178–2189 (2006).
- 17 31. Bernaudat, F. *et al.* Heterologous expression of membrane proteins: choosing the appropriate host.
18 *PloS One* **6**, e29191 (2011).
- 19 32. Eshaghi, S. *et al.* An efficient strategy for high-throughput expression screening of recombinant
20 integral membrane proteins. *Protein Sci.* **14**, 676–683 (2005).
- 21 33. Gordon, E. *et al.* Effective high-throughput overproduction of membrane proteins in *Escherichia*
22 *coli*. *Protein Expr. Purif.* **62**, 1–8 (2008).
- 23 34. Petrovskaya, L. E. *et al.* Expression of G-protein coupled receptors in *Escherichia coli* for structural
24 studies. *Biochem. Mosc.* **75**, 881–891 (2010).
- 25 35. Szakonyi, G. *et al.* A genomic strategy for cloning, expressing and purifying efflux proteins of the
26 major facilitator superfamily. *J. Antimicrob. Chemother.* **59**, 1265–1270 (2007).
- 27 36. Finn, R. D. *et al.* Pfam: the protein families database. *Nucleic Acids Res.* **42**, D222–230 (2014).
- 28 37. Li, G.-W., Oh, E. & Weissman, J. S. The anti-Shine-Dalgarno sequence drives translational pausing
29 and codon choice in bacteria. *Nature* **484**, 538–541 (2012).
- 30 38. Gamble, C. E., Brule, C. E., Dean, K. M., Fields, S. & Grayhack, E. J. Adjacent Codons Act in
31 Concert to Modulate Translation Efficiency in Yeast. *Cell* **166**, 679–690 (2016).
- 32 39. Chartron, J. W., Hunt, K. C. L. & Frydman, J. Cotranslational signal-independent SRP preloading
33 during membrane targeting. *Nature* **536**, 224–228 (2016).
- 34 40. Zadeh, J. N. *et al.* NUPACK: Analysis and design of nucleic acid systems. *J. Comput. Chem.* **32**,
35 170–173 (2011).
- 36 41. Towns, J. *et al.* XSEDE: Accelerating Scientific Discovery. *Comput. Sci. Eng.* **16**, 62–74 (2014).
- 37 42. Canty, A. & Ripley, B. D. *boot: Bootstrap R (S-Plus) Functions.* (2015).
- 38 43. DeLong, E. R., DeLong, D. M. & Clarke-Pearson, D. L. Comparing the areas under two or more
39 correlated receiver operating characteristic curves: a nonparametric approach. *Biometrics* **44**, 837–
40 845 (1988).
- 41 44. Tukey, J. W. *Exploratory data analysis.* (Addison-Wesley Pub. Co, 1977).
- 42 45. Tufte, E. R. *The visual display of quantitative information.* (Graphics Press, 2001).
- 43 46. Van Rossum, G. & Drake Jr, F. L. *Python reference manual.* (Centrum voor Wiskunde en
44 Informatica Amsterdam, 1995).
- 45 47. Cock, P. J. A. *et al.* Biopython: freely available Python tools for computational molecular biology
46 and bioinformatics. *Bioinformatics* **25**, 1422–1423 (2009).

- 1 48. van der Walt, S., Colbert, S. C. & Varoquaux, G. The NumPy Array: A Structure for Efficient
2 Numerical Computation. *Comput. Sci. Eng.* **13**, 22–30 (2011).
- 3 49. Tange, O. GNU Parallel - The Command-Line Power Tool. *Login USENIX Mag.* **36**, 42–47 (2011).
- 4 50. R Core Team. *R: A Language and Environment for Statistical Computing*. (R Foundation for
5 Statistical Computing, 2015).
- 6 51. RStudio Team. *RStudio: Integrated Development Environment for R*. (RStudio, Inc., 2015).
- 7 52. Bache, S. M. & Wickham, H. *magrittr: A Forward-Pipe Operator for R*. (2014).
- 8 53. Wickham, H. The Split-Apply-Combine Strategy for Data Analysis. *J. Stat. Softw.* **40**, 1–29 (2011).
- 9 54. Wickham, H. & Francois, R. *dplyr: A Grammar of Data Manipulation*. (2015).
- 10 55. Aho, K. *asbio: A Collection of Statistical Tools for Biologists*. (2015).
- 11 56. Weinert, K. *datamart: Unified access to your data sources*. (2014).
- 12 57. Wickham, H. *ggplot2: elegant graphics for data analysis*. (Springer New York, 2009).
- 13 58. Clarke, E. & Sherrill-Mix, S. *ggbeeswarm: Categorical Scatter (Violin Point) Plots*. (2015).
- 14 59. Schloerke, B. *et al. GGally: Extension to 'ggplot2'*. (2016).
- 15 60. Auguie, B. *gridExtra: Miscellaneous Functions for 'Grid' Graphics*. (2015).
- 16 61. Wilke, C. O. *cowplot: Streamlined Plot Theme and Plot Annotations for 'ggplot2'*. (2015).
- 17 62. Wickham, H. *scales: Scale Functions for Visualization*. (2015).
- 18 63. Garnier, S. *viridis: Default Color Maps from 'matplotlib'*. (2016).
- 19 64. Neuwirth, E. *RColorBrewer: ColorBrewer Palettes*. (2014).
- 20 65. Harrower, M. & Brewer, C. A. ColorBrewer.org: an online tool for selecting colour schemes for
21 maps. *Cartogr. J.* **40**, 27–37 (2003).
- 22 66. Wickham, H. *multidplyr: Partitioned data frames for 'dplyr'*.
- 23 67. Revolution Analytics & Weston, S. *foreach: Provides Foreach Looping Construct for R*. (2015).
- 24 68. Revolution Analytics & Weston, S. *iterators: Provides Iterator Construct for R*. (2015).
- 25 69. Revolution Analytics & Weston, S. *doMC: Foreach Parallel Adaptor for 'parallel'*. (2015).
- 26 70. Revolution Analytics & Weston, S. *doParallel: Foreach Parallel Adaptor for the 'parallel'*
27 *Package*. (2015).
- 28 71. Wickham, H., Danenberg, P. & Eugster, M. *roxygen2: In-Source Documentation for R*. (2015).
- 29 72. Zhou, J. & Rudd, K. E. EcoGene 3.0. *Nucleic Acids Res.* **41**, D613–624 (2013).
- 30 73. Schuler, G. D., Epstein, J. A., Ohkawa, H. & Kans, J. A. Entrez: molecular biology database and
31 retrieval system. *Methods Enzymol.* **266**, 141–162 (1996).
- 32 74. UniProt Consortium. Reorganizing the protein space at the Universal Protein Resource (UniProt).
33 *Nucleic Acids Res.* **40**, D71–75 (2012).
- 34 75. Cock, P. [BioPython] Downloading CDS sequences. (2009).
- 35 76. Chen, L., Oughtred, R., Berman, H. M. & Westbrook, J. TargetDB: a target registration database for
36 structural genomics projects. *Bioinformatics* **20**, 2860–2862 (2004).
- 37 77. Tomb, J. F. *et al.* The complete genome sequence of the gastric pathogen *Helicobacter pylori*.
38 *Nature* **388**, 539–547 (1997).
- 39 78. Lechat, P., Hummel, L., Rousseau, S. & Moszer, I. GenoList: an integrated environment for
40 comparative analysis of microbial genomes. *Nucleic Acids Res.* **36**, D469–474 (2008).
- 41 79. Cole, S. T. *et al.* Deciphering the biology of *Mycobacterium tuberculosis* from the complete genome
42 sequence. *Nature* **393**, 537–544 (1998).
- 43 80. Dobrovetsky, E. *et al.* High-throughput production of prokaryotic membrane proteins. *J. Struct.*
44 *Funct. Genomics* **6**, 33–50 (2005).
- 45 81. Nelson, K. E. *et al.* Evidence for lateral gene transfer between Archaea and bacteria from genome
46 sequence of *Thermotoga maritima*. *Nature* **399**, 323–329 (1999).

- 1 82. Peden, J. F. Analysis of codon usage. (University of Nottingham, 2000).
- 2 83. dos Reis, M., Wernisch, L. & Savva, R. Unexpected correlations between gene expression and
3 codon usage bias from microarray data for the whole Escherichia coli K-12 genome. *Nucleic Acids*
4 *Res.* **31**, 6976–6985 (2003).
- 5 84. Lorenz, R. *et al.* ViennaRNA Package 2.0. *Algorithms Mol. Biol. AMB* **6**, 26 (2011).
- 6 85. Coleman, J. R. *et al.* Virus attenuation by genome-scale changes in codon pair bias. *Science* **320**,
7 1784–1787 (2008).
- 8 86. Yang, Z. R., Thomson, R., McNeil, P. & Esnouf, R. M. RONN: the bio-basis function neural
9 network technique applied to the detection of natively disordered regions in proteins.
10 *Bioinformatics* **21**, 3369–3376 (2005).
- 11 87. Wimley, W. C., Creamer, T. P. & White, S. H. Solvation energies of amino acid side chains and
12 backbone in a family of host-guest pentapeptides. *Biochemistry (Mosc.)* **35**, 5109–5124 (1996).
- 13 88. Engelman, D. M., Steitz, T. A. & Goldman, A. Identifying nonpolar transbilayer helices in amino
14 acid sequences of membrane proteins. *Annu. Rev. Biophys. Biophys. Chem.* **15**, 321–353 (1986).
- 15 89. Kuhn, M. Building predictive models in R using the caret package. *J Stat Soft* (2008).
- 16 90. Weihs, C., Ligges, U., Luebke, K. & Raabe, N. in *Data Analysis and Decision Support* (eds. Baier,
17 D., Decker, R. & Schmidt-Thieme, L.) 335–343 (Springer-Verlag, 2005).
- 18 91. Kendall, M. G. A New Measure of Rank Correlation. *Biometrika* **30**, 81 (1938).
- 19 92. Joachims, T. Optimizing search engines using clickthrough data. in 133 (ACM Press, 2002).
20 doi:10.1145/775047.775067
- 21 93. Robin, X. *et al.* pROC: an open-source package for R and S+ to analyze and compare ROC curves.
22 *BMC Bioinformatics* **12**, 77 (2011).
- 23 94. Cunningham, F. *et al.* Ensembl 2015. *Nucleic Acids Res.* **43**, D662–669 (2015).
- 24 95. Smedley, D. *et al.* The BioMart community portal: an innovative alternative to large, centralized
25 data repositories. *Nucleic Acids Res.* **43**, W589–598 (2015).
- 26 96. Harris, T. W. *et al.* WormBase 2014: new views of curated biology. *Nucleic Acids Res.* **42**, D789–
27 793 (2014).
- 28 97. Engel, S. R. *et al.* The reference genome sequence of *Saccharomyces cerevisiae*: then and now. *G3*
29 *Bethesda Md* **4**, 389–398 (2014).
- 30 98. De Schutter, K. *et al.* Genome sequence of the recombinant protein production host *Pichia pastoris*.
31 *Nat. Biotechnol.* **27**, 561–566 (2009).
- 32 99. Nordberg, H. *et al.* The genome portal of the Department of Energy Joint Genome Institute: 2014
33 updates. *Nucleic Acids Res.* **42**, D26–31 (2014).
- 34 100. Markowitz, V. M. *et al.* IMG 4 version of the integrated microbial genomes comparative analysis
35 system. *Nucleic Acids Res.* **42**, D560–567 (2014).
- 36 101. Uchiyama, I., Mihara, M., Nishide, H. & Chiba, H. MBGD update 2015: microbial genome
37 database for flexible ortholog analysis utilizing a diverse set of genomic data. *Nucleic Acids Res.*
38 **43**, D270–276 (2015).
- 39 102. Loader, C. *locfit: Local Regression, Likelihood and Density Estimation.* (2013).
- 40 103. Schindelin, J. *et al.* Fiji: an open-source platform for biological-image analysis. *Nat. Methods* **9**,
41 676–682 (2012).
- 42 104. Schneider, C. A., Rasband, W. S. & Eliceiri, K. W. NIH Image to ImageJ: 25 years of image
43 analysis. *Nat. Methods* **9**, 671–675 (2012).
- 44 105. Xie, Y. in *Implementing Reproducible Computational Research* (eds. Stodden, V., Leisch, F. &
45 Peng, R. D.) (Chapman and Hall/CRC, 2014).