Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

OXFORD

Gene Expression

Insight to gene expression from promoter libraries with the machine learning workflow *Exp2lpynb*

Ulf W. Liebal [1,*], Sebastian Köbbing [1], Linus Netze [2], Artur M. Schweidtmann [3], Alexander Mitsos [2] and Lars M. Blank [1]

[1] iAMB - Institute of Applied Microbiology, ABBT, RWTH Aachen University, 52074 Aachen, Germany.
[2] AVT - Process Systems Engineering, RWTH Aachen University, 52074 Aachen, Germany.
[3] Department of Chemical Engineering, Delft University of Technology, Van der Maasweg 9, Delft 2629 HZ, The Netherlands.

* To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Metabolic engineering relies on modifying gene expression to regulate protein concentrations and reaction activities. The gene expression is controlled by the promoter sequence and sequence libraries are used to scan expression activities and to identify predictive sequence positions. We present a computational workflow called *Exp2lpynb* to analyze promoter libraries maximizing information retrieval and sequences prediction with defined activity. We applied *Exp2lpynb* to seven prokaryotic expression libraries to identify optimal experimental design principles.
Results: The workflow is open source, available as Jupyter Notebooks and covers the steps to (i) generate a statistical overview to sequence and activity, (ii) train machine-learning algorithms, such as random forest, gradient boosting trees and support vector machines for prediction and extraction of feature importance, (iii) evaluate the performance of the estimator, and (iv) to predict new sequences with defined activity using optimization methods. The workflow can perform regression or classification on multiple promoter libraries, across species or reporter proteins. The most accurate predictions in the sample libraries were achieved when the promoters in the library were recognized by a single sigma factor and a unique reporter system. The prediction confidence mostly depends on samples size and sequence diversity, and we present a relationship to estimate their respective effects. The workflow can be adapted to process sequence libraries from other expression-related problems and contributes to increase insight to the growing application of high-throughput experiments, providing support for efficient strain engineering.
Availability: Freely available on GitHub: https://github.com/iAMB-RWTH-Aachen/Exp2lpynb and licensed under the terms of GPLv3.
Contact: ulf.liebal@rwth-aachen.de
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Metabolic engineering aims at optimizing metabolite production by adjusting the activity of native and heterologous enzymes. A frequently manipulated factor of activity is the enzyme concentration, which can be regulated on the transcriptional, translational, and post-translational level. In bacteria, enzyme concentrations are set largely at the transcriptional level (Balakrishnan *et al.*, 2021). Among the most import transcriptional element is the promoter sequence. The promoter sequence is the primary target for metabolic engineering by allowing fine-tuning and modular replacement. Both, the composition and regulation of expression by promoters have been intensively studied (Kalisky *et al.*, 2007; Zaslaver *et al.*, 2009; Kochanowski *et al.*, 2017). For example, Rhodius and coworkers investigated the expression strength of 60 σ^{E} promoters and analyzed the impact of promoter boxes and upstream regulating elements on expression (Rhodius and Mutalik, 2010; Rhodius *et al.*, 2012). A modular promoter system was developed by (Mutalik *et al.*, 2013), that

reduces interference from the sequence of the gene of interest, resulting in reproducible promoter activities.

Promoter libraries are routinely constructed to generate promoters with defined activities (Jensen *et al.*, 1993; Alper *et al.*, 2005; Hammer *et al.*, 2006; Balzer *et al.*, 2013; Köbbing *et al.*, 2020) and machine learning has been applied for better understanding of transcription mechanisms and sequence prediction. For example, Meng *et al.* analyzed 98 σ^{70} promoter sequences in *E. coli* and fine-tuned heterologous expression with predicted synthetic promoters (Meng *et al.*, 2013, 2017). The machine learning analysis therein was based on artificial neural networks (ANN) (Meng *et al.*, 2013) and support vector machines (SVM) (Meng *et al.*, 2017), and both approaches performed comparably. The same system with σ^{70} driven expression in *E. coli* was tested by Zhao *et al.* with over 3,500 promoter sequences (Zhao *et al.*, 2020) and analyzed using projection to latent spaces (PLS), tree methods (gradient boosting trees, GBT), and recurrent neural networks, wherein GBT performed best. In *Bacillus subtilis* Liu *et al.* employed a synthetic promoter library with 214 sequences to adjust pathway activity for metabolite overproduction (Liu *et al.*, 2018) and used PLS for regression analysis. To facilitate metabolic engineering in *Geobacillus thermoglucosidasius*, a library with 80 sequences was tested for both the expression of GFP and mOrange (Gilman *et al.*, 2019) and trained on models of PLS and ANN. Overall, the libraries were generated with a varying sample sizes from 60 to over 3,500 and enabled model-based sequence analysis and rational promoter development.

Promoter libraries are typically analyzed by individually developed scripts, which is time intense and impedes direct comparison of performance measures. To unify the analysis and to provide a platform suitable for easy reconfiguration, we have developed a general workflow for promoter library analysis called *Exp2Ipynb*. The *Exp2Ipynb* workflow consists of a collection of Python Jupyter Notebooks to (i) generate a statistical overview to sequence and activity, (ii) train machine-learning algorithms for prediction and extraction of feature importance, (iii) evaluate the performance of the estimator, and (iv) to predict new sequences with defined activity using optimization methods. We tested the workflow on an existing promoter library for σ^{70} in *Pseudomonas putida* KT2440, expanded with new sequences. The results revealed limitations of one-factor-at-a-time experimental designs for machine learning. Subsequently, we used six published libraries to compare their composition and the resulting machine learning performance.

2 Methods

2.1 Data preparation and statistical analysis

A configuration file stores variables with global use in each Notebook assisted analysis. The data input is a comma-separated-value file (*csv*) with at least three columns: (i) an identifier column, (ii) a sequence column, and (iii) an expression value column, with header names. Optional columns are expression values of the sequences in other organisms or with a different reporter system and the standard deviation with the replicate numbers. The sequence column only accepts DNA abbreviations (A, C, G, T) with an identical length in each sample. The output file names and figure file types can be defined. The column names for data import are defined in a separate configuration file called 'config.txt' and the notebook '0-Workflow' guides through its construction.

The statistical analysis is conducted in the notebook '1-Statistical-Analysis'. In addition to a single expression value, the standard deviation and replicate number can be provided. Optionally, outliers in the original data set can be removed from further analysis. Machine learning performance is improved if replicates are available and the workflow enables the re-generation of replicates based on mean and standard deviation. The replicates are calculated from Python numpy random normal function (Harris *et al.*, 2020) and is valid for normal-distributed data while adding a reasonable prediction bias.

The statistical analysis in the Notebook '1-Statistical-Analysis.ipynb' provides an overview to the metrics with relevance for data exploration and model development. The sequence diversity represents how different the sequences are among each other. It is calculated as the normalized sum of nucleotide differences relative to a reference sequence or among all sequences and ranges between 0 (identical) to 1 (each position differs). The reference sequence can be provided with the configuration file, or it is generated automatically by finding the most common nucleotide on each position. For large libraries, *i.e.*, >1000 samples, the full pairwise distance is costly to compute and the reference sequence distance is performed by default. The position diversity informs about how many nucleotides have been sampled for each position. It is visualized with two bar plots: (i) the cumulative number of each nucleotide tested on each position, and (ii) the entropy (H_i) for any sequence position (i):

$$H_i = -\sum_{i=(A,C,G,T)} p_i \log_2 p_i \tag{1}$$

where p_i is the position-related nucleotide frequency. The expression statistics for all nucleotides and positions $\overline{Exp}$ informs how each nucleotide contributes to the expression:

$$\overline{Exp} = \frac{\sum_{n=1}^{S} \begin{pmatrix} & A & C & G & T & \\ 1 & \begin{pmatrix} 0 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 1 & 0 \end{pmatrix}_n & \cdot Exp_n \\ R & & & & & \end{pmatrix}}{S}$$

where each sequence is transformed into a one-hot matrix with the four nucleotides as columns, (A, C, G, T) and the sequence as rows with the maximum sequence length R. This sample-sequence one-hot matrix is multiplied by the associated expression strength (Exp_n) and is used to calculate the mean (standard deviation) of expression at each position-nucleotide pair over all samples (S). The expression strength is visualized with a histogram and a scatter plot for cross library expression.

2.2 Predictor training and performance

The workflow supports classification and regression. In regression, the expression values for sequences are quantitatively predicted, but require high data quality (sufficient sample size and position entropy). Classification provides qualitative predictions, (*e.g.*, low-medium-high), with more reliable predictions even for small data sets. The implemented machine learning routines are random forest (RF), gradient boosting trees (GB), and support vector machines (SVM). The input features are nucleotides on each position in one-hot encoding plus the overall sequence GC-content. The predicted target variable is the expression strength. The feature size depends on the sequence length typically ranging between 40-100 nucleotides or 160-400 features. If classification is chosen, the output is binned according a parameter provided by the user ($Response\_Value$ in *config.csv*). If $Response\_Value = 1$, the original activity values are used, if the $Response\_Value = 0$, the data is centered with zero mean and unit variance and for larger values bins are generated as equal sized buckets (python pandas 'qcut'), and the bin label is used as the target prediction. The data is split into training and test set, with a default ratio of 9:1 and a grid search on the training set identifies the optimal hyper-parameter.

Additional feature selection procedures were implemented to increase performance. During feature selection, the number of nucleotides that serve as features can be reduced to optimize training. Reasonable

predictions rely on a sufficient variety of nucleotides tested at each position (H_i), and a cut-off can be chosen in the configuration file to exclude positions below a defined value of H_i. However, note that non-canonical nucleotides in conserved promoter (box)-regions can result in drastic expression deficiencies. Thus, a low diversity can also reflect critical nucleotides associated with difficulties to sample experimentally.

The performance evaluation provides metrics for correlating the experimental and predicted outcomes and informs about important features for tree-based methods. The performance evaluation is based on cross-validation with 9:1 data separation that jointly moves identical sequences (*i.e.*, replicates) to test or training sets. Thus, the test sequences are unknown to the regressor. The performance evaluation is based on the R2- (regression) and weighted F1-score (classification) (and optionally Matthews correlation coefficient) for training set and the test set. The prediction uncertainty metric is determined by the coefficient of variance of the mean R2 and F1 prediction of the cross validated training set. The tree-based methods allow the extraction of feature importance and represent the contributions of each nucleotide-position and are visualized with Logo-plots (Tareen and Kinney, 2020). Moreover, single decision trees can be exported.

2.3 Synthetic sequence identification by optimization

The ability to predict new sequences with defined activities is required for effective strain engineering. The predictors are used to optimize sequence and expression with a genetic algorithm based on the Python framework DEAP (Fortin *et al.*, 2012). The genetic optimization algorithm is used as it can be easily applied to a broad variety of machine learning models. Notably, deterministic global methods can guarantee the identification of global solutions and have recently been applied to complex machine learning models (Schweidtmann and Mitsos, 2019; Schweidtmann *et al.*, 2020; Thebelt *et al.*, 2020). Sequence identification for a regressor is conducted with a search for the sequence with closest expression. The classification requires that the predicted expression lies within the target expression class, while the sequence distance with respect to reference sequences is minimized. The sequences of measured expression in the data are excluded from the search.

2.4 Experimental library construction

Construction of the single nucleotide polymorphism (SNP) library was based on the plasmid pBG14g as template with oligonucleotides containing single degenerate nucleotides inside the P14g promoter sequence (Zobel *et al.*, 2015; Köbbing *et al.*, 2020). The fragments and vector pBG were digested with *PacI* and *NcoI* (New England Biolabs) at 37°C. Digested backbone and promoter containing PCR fragments were ligated with T4 ligase (New England Biolabs) at room temperature for 30 minutes. Transformation into chemically competent *E. coli* PIR2 cells was done by heat shock (Hanahan, 1983). Plasmids containing different synthetic promoter sequences were sequenced (Eurofins Genomics) and genomically integrated into the *att*Tn7 site of *P. putida* KT2440 by mating (Zobel *et al.*, 2015). For promoter characterization in *P. putida* KT2440, cells were grown in minimal medium (Hartmans *et al.*, 1989), with 20 mM glucose as carbon source. The Biolector system (M2P Labs) measured optical density (620 nm) and msfGFP (excitation wavelength 488 nm, emission wavelength 520 nm). Scattered light was correlated to OD600 with a dilution series of a stationary phase culture. Promoter activity is reflecting the slope of the function of fluorescence over OD600 at the beginning of the exponential phase. More detailed information is given in Köbbing *et al.*, 2020.

Table 1. Classification quality report for random forest (RF), gradient boosting trees (GBT) and support vector machine (SVM). The training was performed with the same Train(56)-Test(7) division with cross-validation on the training set (10% hold-out). Nucleotide-position feature selection based on entropy threshold (0.2 bits) resulted in 15 positions, in addition to GC-content (input vector: $15 \times 4 + 1$). Only tree-based methods (RF, GBT) extract the feature importance (FI).

	RF	GBT	SVM
Run time	144	927	111
CV:F1-score	0.42±0.19	0.5±0.22	0.47±0.21
Train:F1-score	0.58	0.89	0.88
Test:F1-score	0.62	0.46	0.14
GC-content	2nd	1st	N.A.
Top 3 FI	$-35:T:0.24$	$-34:T:0.08$	N.A.
	$-34:T:0.10$	$-35:A:0.05$	N.A.
	$-35:A:0.10$	$-14:G:0.05$	N.A.

3 Case studies

The workflow was first applied on a newly extended *P. putida* KT2440 synthetic promoter library, followed by a cross analysis of six published libraries. The analysis of our *P. putida* library will show how the workflow can be used, and we will highlight shortcomings of the experimental design for machine-learning driven analysis. In the cross-library comparison, we investigated how the different library parameters of sample size, diversity and feature number impact the predictions.

3.1 *P. putida* single library analysis

We used the workflow to analyze a synthetic library in *P. putida* KT2440 driven by σ^{70}-dependent promoters and measured with GFP published previously (Köbbing *et al.*, 2020) but expanded by eight new sequences. The goal was to identify sequence features responsible for expression strength. Experimentally, the sequence diversity was generated by single nucleotide exchanges in 28 nucleotides upstream of the transcription start site resulting in 63 unique sequences. For the analysis we used 40 nucleotides of the promoter as input and binned the expression activity into three approximately equal classes for the output. The sequences had low information content on most positions (Figure 1 (A) and (B)) allowing only predictions of categorized expression values, a regression predicted not better than random (not shown).

A classification estimator can predict the approximate magnitude of expression. A reasonable estimation can only be performed on features with sufficient information content and positions with a higher entropy than 0.2 bits were included for prediction. Figure 1 (A) shows the entropy of the full data set, and because the cut-off is applied to the training subset, the following positions are additionally neglected [-6,-15-20,-22,-26]. Decreasing the entropy threshold had no effect on the performance (not shown). The classification details are given in Table 1, and the high F1-score variation indicates that the low nucleotide redundancy on each position affected sample separation during cross validation. The detailed prediction results of the training and test set are shown as confusion matrix in Figure 2 (A). The GC-content is among the most important features for the prediction probably because it has the highest entropy of all features (1.2 bits). Along with the GC-content, the strongest determinant of expression were the positions –35 and –34, corresponding to the fact that the -35-box is an important site of transcription factor binding (Paget, 2015) (Figure 2 (B)). New sequences with defined expression activity were identified. Sequences with close relation to the reference sequence were chosen, because the data used for training itself has a low sequence

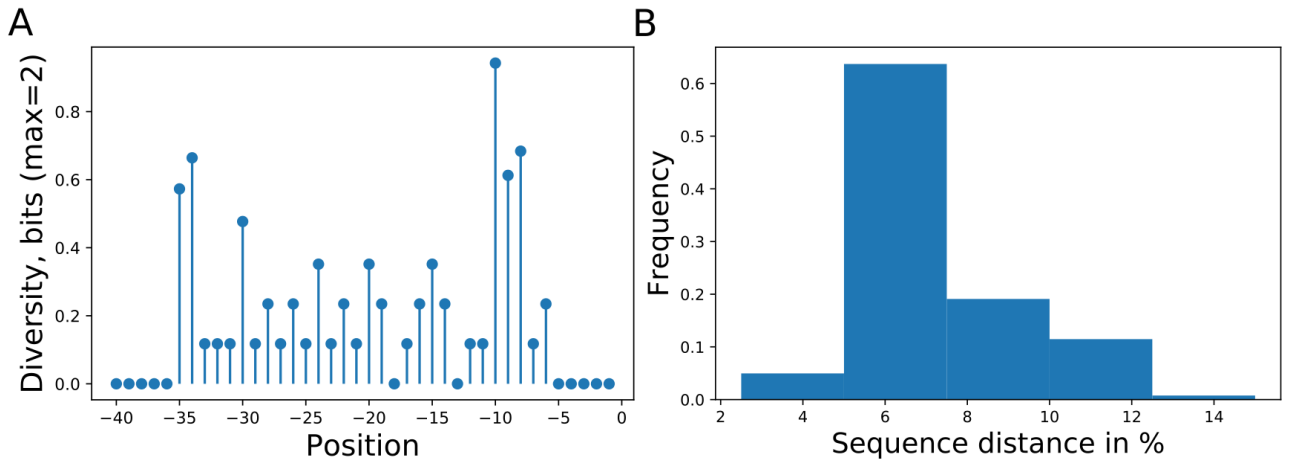

Fig. 1. (A): Promoter sampling diversity of the complete data set for nucleotide variations on each sequence position, transcription starts at '0' and (B): mutual sequence distances. The promoter library is based on our previously published library (Köbbing et al., 2020), with additional sequences totaling 63 different promoters with mutations directly upstream of the transcription start site. Two positions ([-13,-18]) were not mutated resulting in a total of 28 tested positions.

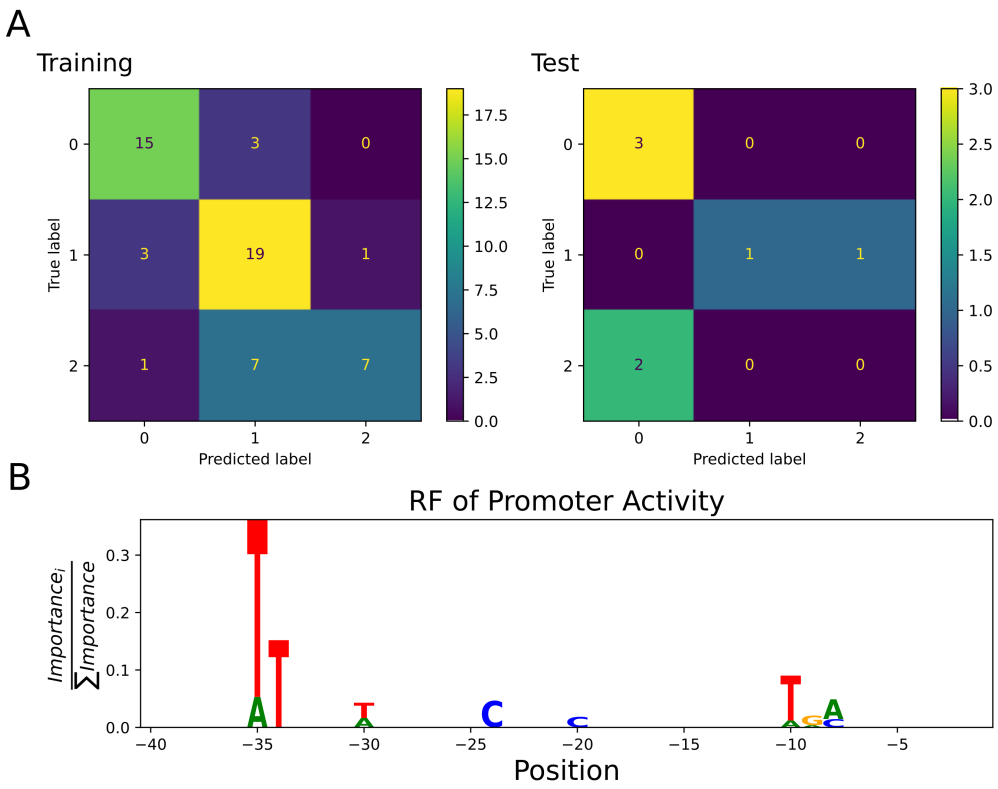

Fig. 2. (A): Confusion matrix showing classification quality for training (n=56) and test set (n=7). The classification labels are '0': low (expression 0.2 - 16.6), '1': medium (16.6 - 24.8), 2: high (24.8 - 35.2)((Köbbing et al., 2020)). (B): Sequence logo of positions used by the random forest to predict expression activity class. The -35 and -10 box positions dominated the feature importance.

diversity (Figure 1), restricting the application of the prediction based on the *P. putida* data set. Two predicted sequences with low expression and one with high expression were experimentally tested (see Table 2). While the promoters with low activity could be validated, the predicted promoter with high activity only showed medium activity experimentally.

3.2 Cross-library analysis

In the following, six published bacterial promoter libraries were analyzed with the aim to highlight the effect of the library design on the estimation quality. The data is available online in the *Exp2Ipynb* package at GitHub. The libraries were measured in *E. coli*, *Geobacillus thermoglucosidasius*, and *B. subtilis* and were targeted to specific or unknown sigma factors. All

Table 2. Performance of newly predicted promoters in comparison to reference promoters. The reference promoter is shown with the expression measured originally and during reevaluation.

ID	Seq.Diff.	Predicted	Experiment
Ref:mod4_1	GGTATAAT	9 ± 2	6.5
pred:0-1	GGTA**C**AAT	$0.25 - 16.5$	7 ± 0.9
pred:0-2	GGTAT**T**AT	$0.25 - 16.5$	7 ± 0.9
Ref:SPL14g_14_g	GGTATAAT	29	26 ± 1.5
pred:2-1	**C**GTATAAT	$25 - 35$	16 ± 0.4

Table 3. Details of the six published libraries used to compare estimation quality in response to library experimental design. The columns 'n' represent total sample size, 'Feat.' the size of the input vector, 'Avg.Seq.Distance' the average distance of all sequences to a reference sequence, composed of the most common nucleotide at each position. Full table with numerical values of Figure 3 in appendix.

Reference	Organism	Transcr. Factor	Reporter	n	Feat.	Avg.Seq. Distance
a:This Work	*P. putida*	σ^{70}	GFP	63	61	0.06
b:Rhodius *et al.*	*E. coli*	σ^{E}	various	59	121	0.61
c:Meng *et al.*	*E. coli*	σ^{70}	GFP	98	709	0.61
d:Zhao *et al.*	*E. coli*	σ^{E}	GFP	3543	285	0.09
e:Gilman *et al.*	*G. therm.*	N.A.	GFP	81	397	0.69
f:Gilman *et al.*	*G. therm.*	N.A.	mOrange	81	397	0.69
g:Liu *et al.*	*B. subtilis*	σ^{A}	GFP	206	105	0.35
h:Meng *et al.*	*E. coli*	σ^{70}	GFP	84	133	0.61
i:Meng *et al.*	*E. coli*	σ^{70}	GFP	84	129	0.61
k:Zhao *et al.*	*E. coli*	σ^{E}	GFP	896	161	0.09

libraries differ from each other in terms of combinations of library sample size, tested sequence length and sequence diversity (Table 3).

The analysis was based on a classification task with a random forest and three classes: low ('1'), medium ('2'), and high expression ('3'). The original data of the promoter libraries was used with minor corrections regarding sequence length homogenization. For quality assessment, we generated three synthetic data sets based on partitioned sequence regions from Meng *et al.* (2013) and Zhao *et al.* (2020). The original sequence from Meng *et al.* (2013), is 224 nucleotides (nt) long, and we extracted the starting 40 nt ('h') and last 40 nt ('i'), assuming that few predictive positions are contained in the new sequences. Zhao *et al.* (2020), tested 113 nt ranging from upstream regulating elements down to the coding sequence. Again, the first 40 nt ('k') were extracted, corresponding to the upstream regulating element. Three values are important for analysis of promoter libraries: (i) the fidelity of gene expression prediction, computed by the F1-score (see methods), (ii) the confidence of the prediction, computed by the coefficient of variation, and (iii) important sequence features for expression, computed by the feature importance of random forest and gradient boosting strategies.

Over all libraries, the prediction qualities, evaluated by the correlation coefficient, ranged between 0.3-0.6. The best predictive quality (F1-score) and confidence (coefficient of variation of F1-score) was achieved by the promoter library with more than 3,500 samples (Figure 3, (A), library 'd'). A higher sample size was associated with lower sequence diversity and consequently better prediction confidence, with coefficient of variations below 0.2 ('d', 'g'). Notably, the F1-score was independent of the number of tested input features (Figure 3 (A)). The effect of the training sample size on the coefficient of variation is hyperbolic decreasing (Figure 3 (C)): below 200 samples represented by library 'g', the coefficient of variation rises thus decreasing prediction confidence. Two scenarios are visible in Figure 3 (C), 'a' and 'c' display already a very low sequence diversity and focus should be on more diverse sequences. In contrast, libraries 'b', 'e', and 'f' are diverse, and increasing the sample size even with related sequences would be most beneficial. An important information is the feature importance with nucleotide positions of outstanding predictability. Figure 3 (B) indicates how the number of features affects the sum of the three top important features derived from the RF. The top-three feature importance sum decreases linearly with increasing numbers of features. However, for library 'c', the top three nucleotide positions are much more predictive than expected assuming that more features lead to an larger distribution of feature importances and thus less proportional share for the top three features available.

We were interested to examine how the performance is affected when subsequences were extracted from libraries. We chose the libraries from Meng *et al.* (2013) and Zhao *et al.* (2020) because they report a high feature number with low sequence diversity. Subsequences from Meng *et al.* (2013) were extracted with 40 nt sequences at the start ('h') and end ('i') because these contain features with low importance. The rational to use the starting 40 nt from Zhao *et al.* (2020) ('k') is that they contained the upstream regulating element and thus are important expression features. The sequence extraction reduces the sample size because more redundant sequences are generated and thus increase the nucleotide diversity along with shorter sequence length. Figure 3 shows that the partitioned data set 'k' (Zhao *et al.*, 2020) maintained the prediction confidence from the original data set: the F1-CoV and top three feature contribution increased proportionally to the original data (Figure 3 B, C, and D). This proportional movement was not displayed by the extracted sequences ('h', 'i') from Meng *et al.* (2013). Thus, sequence positions impacting expression were identified in the upstream regulating elements of Zhao *et al.* (2020), but not in the first and last 40 nt of the sequences of Meng *et al.* (2013). The synthetic data set 'k' is in line with the sample size correlation to the coefficient of variance of the libraries 'd' and 'g'. These results indicate that sample size in the range of 200 to 3,500 is not critical for the coefficient of variance.

4 Discussion

Synthetic promoter libraries are increasingly constructed to facilitate promoter selection with defined activities. The *Exp2Ipynb* workflow supports the analysis of promoter libraries to identify the sequence information that determines expression strength. The identification is based on statistical analysis of the average nucleotide-position associated expression, as well as the training of different machine-learning algorithms. Moreover, the analysis of more complex libraries with multiple readouts, like two reporter proteins, transcript, and protein level, or cross host expression can be analyzed. The workflow facilitates data exploration, regressor training and performance evaluation, as well as testing of novel sequences within the DNA sequence exploration space. The implementation in a Jupyter notebook facilitates rapid implementation and the low-level scripting allows for direct adaptation of the workflow to specific needs. In the following, we summarize the applicability of *Exp2Ipynb* to a *P. putida* promoter library with sub-optimal data quality, followed by an discussion of data properties for optimal sequence analysis.

We analyzed a promoter expression library in *P. putida* KT2440 previously published (Köbbing *et al.*, 2020) and amended with additional data points and used the classification model for prediction of new sequences. The different samples in the library were generated based on a one-factor-at-a-time approach to identify nucleotide-sequence effects on expression (Czitrom, 1999). Our results of the strong impact of the positions –35 and –34 and the lower impact of the last half of the -10 box

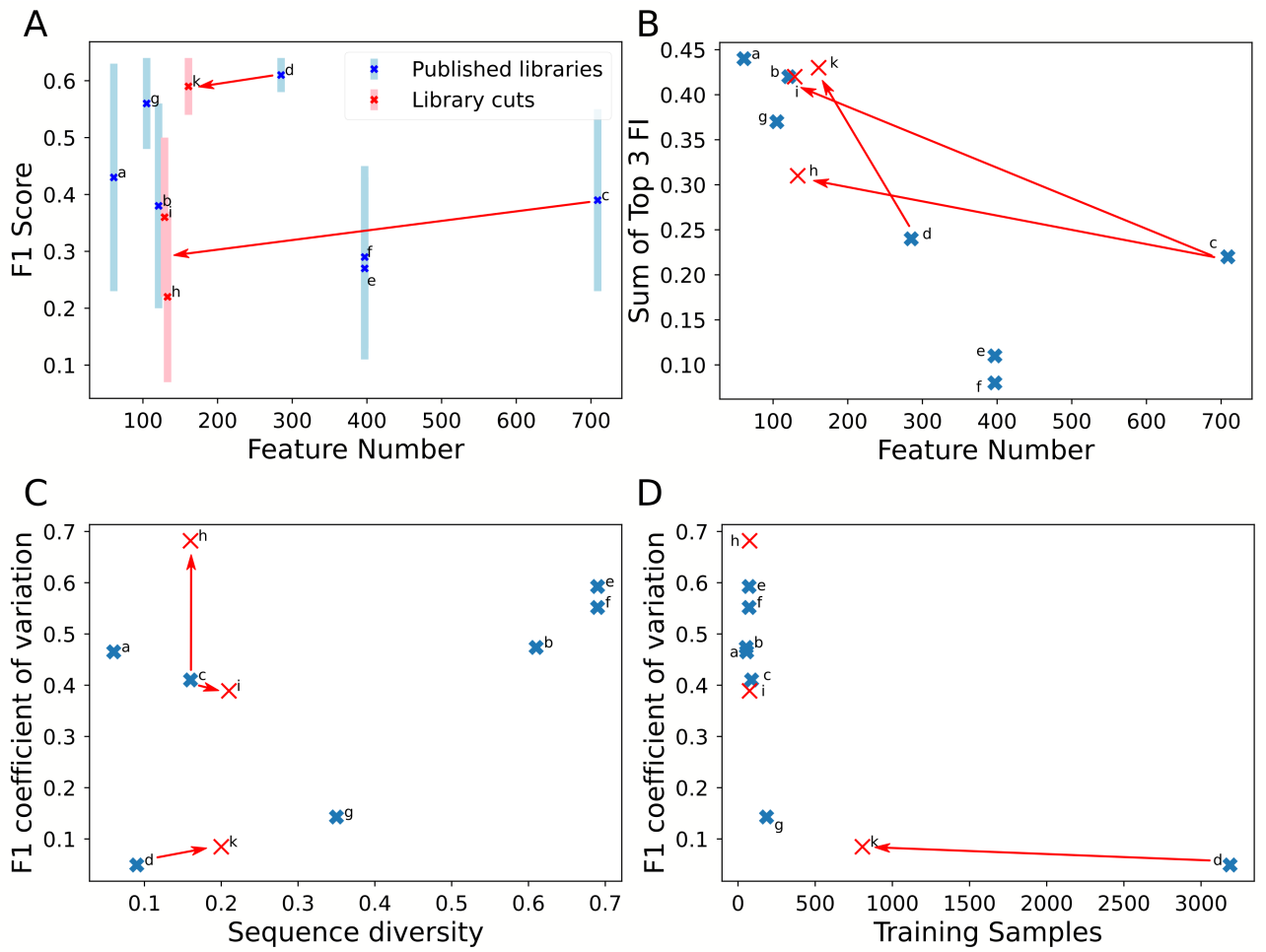

Fig. 3. Comparison of library properties on classification quality parameters for an estimation using RF. The estimations were performed on the training set of each library with 9:1 cross-validation. The seven full length promoter libraries are listed in Table 3. Moreover, sub-sequences of 40 nt were extracted from the promoter start ('h') and end ('i') of Meng et al. (2013)('c') and start ('k') of Zhao et al. (2020)('d'). F1-score (A) and the sum of the top three feature importance values (B) in response to feature amount. The coefficient of variation of the F1-scores in response to sequence diversity (C) and amount of training samples (D). Full table with numerical values in appendix.

(TATAAT, -10, -9, -8) confirmed results of the original article (Köbbing *et al.*, 2020, Figure 4 therein). However, we failed to observe the strong effect of the first part of the -10 box (TATAAT, -11) because the position was insufficiently sampled below the entropy cutoff and was neglected in the analysis. We used the classifier to apply a genetic algorithm and find sequences within a defined expression range and we confirmed their expression strength experimentally. Thus, even data sets with low sample size and sequence diversity allow for predicting expression ranges for biotechnological use.

Sample size and sequence diversity were the main factors to affect prediction confidence (coefficient of variation) while the number of feature was less important. A high sequence diversity resulted in lower prediction confidence, and increasing the sample size can help to reduce diversity and increase prediction confidence. Two libraries in the collection were limited by sample size ('a'+'c') whereas five libraries were limited by sequence diversity ('b','d','e','f'). We found that 100 samples still resulted in a high uncertainty for sequences with an average of 20% nucleotide difference ('a', 'c'), whereas 800 samples were sufficient ('k'). With higher nucleotide differences of 35%, 200 samples provided reasonable prediction qualities ('g'). Libraries including multiple sigma factors ('e', 'f')(Gilman *et al.*, 2019) resulted in lower prediction qualities, which parallels studies on heterogenous data in *E. coli* (Cambray *et al.*, 2018) and yeast (Liya *et al.*, 2021). More libraries are necessary to narrow the required sample size over the whole sequence diversity spectrum.

Our prediction quality (F1-score) performs poorly over the different libraries. Some studies have identified much stronger regression correlation coefficients (Rhodius and Mutalik, 2010; Meng *et al.*, 2013, 2017), though these were based on optimized train-test set partitions and do not report cross-validation statistics. Gilman *et al.* calculate a stronger correlation for their model, but when the authors tested new data, the prediction qualities were similar to those we observed. Also, the convolutional neural net trained with 500,000 sequences by Cuperus *et al.* (2017), achieves correlation coefficients of 0.62. The regulation of gene expression is controlled by multiple factors. Additional feature engineering can be used to increase predictability: GC-moving window, secondary structure for UTR (Cuperus *et al.*, 2017). However, because gene expression relies on various mechanisms, there will be no single machine learning tool to fit all and machine-learning ensembles are likely needed, composed of different approaches tailored to the various mechanisms.

The *Exp2Ipynb* workflow inludes training of RF, GBT and SVM whereas neural network based predictors are not explicitly implemented.

In the original articles of the libraries tested here, SVM and GBT either performed comparably or outperformed neural network based approaches (Meng *et al.*, 2017; Zhao *et al.*, 2020). There is no single generel optimal algorithm instead the most suitable algorithm has to be tested based on the specific data type and quality (Wolpert and Macready, 1997). With the low samples sizes of most of the libraries presented here, neural networks are unlikely to provide better performance. It is possible to include additional algorithms via the python interface easily.

So far, the analysis of promoter libraries was conducted with scripts tailored to the data, obfuscating reproducibility and interpretability. The *Exp2Ipynb* workflow is a contribution to harmonize analytical workflows and to enable an easy start for new groups with promoter libraries. Other general machine-learning toolboxes exist, *e.g.*, tpot, GAMA, or H2O (Truong *et al.*, 2019) in addition to tools more oriented towards biological data analysis, *e.g.*, JADBIO (Tsamardinos *et al.*, 2020). The advantage of the *Exp2Ipynb* is to be general enough to be used across different data sets in expression studies, but remaining domain specific to facilitate simple data integration.

5 Conclusion

We present a workflow called *Exp2Ipynb* for machine-learning supported analysis of gene-expression libraries. We applied *Exp2Ipynb* in a proof-of-concept on a *P. putida* KT2440 promoter library for use in metabolic engineering. In this context, the workflow allowed the identification of critical sequence features for expression and predicted new sequences with defined activity, tested retrospectively. Moreover, six published prokaryotic gene expression libraries are analyzed and identified the correlation between sample size and sequence diversity for successful analysis. The workflow provides support for deeper analysis of promoter libraries and allows users to adapt it to personal needs. Thus data quality assessments are improved and analysis is accelerated.

Acknowledgements

The authors thank Salome E. Nies and Dario Neves for promoter library data during initial tests. The authors are grateful for discussions with Birgitta E. Ebert, Tobias B. Alter and Bastian Kister.

Funding

UWL received funding by the Excellence Initiative of the German federal and state governments ((DE-82)EXS-PF-PFSDS015). This work is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy—Exzellenzcluster 2186, 'The Fuel Science Center ID: 390919832.' AMS is supported by the TU Delft AI Labs Programme.

References

Alper, H. *et al.* (2005). Tuning genetic control through promoter engineering. *Proc. Natl. Acad. Sci.*, **102**(36), 12678–12683.

Balakrishnan, R. *et al.* (2021). Principles of gene regulation quantitatively connect DNA to RNA and proteins in bacteria. *bioRxiv*, page 2021.05.24.445329.

Balzer, S. *et al.* (2013). A comparative analysis of the properties of regulated promoter systems commonly used for recombinant gene expression in *Escherichia coli*. *Microb. Cell Fact.*, **12**(1), 26.

Cambray, G. *et al.* (2018). Evaluation of 244,000 synthetic sequences reveals design principles to optimize translation in *Escherichia coli*. *Nat. Biotechnol.*, **36**(10), 1005–1015.

Cuperus, J. T. *et al.* (2017). Deep learning of the regulatory grammar of yeast 5' untranslated regions from 500,000 random sequences. *Genome Res.*, **27**(12), 2015–2024.

Czitrom, V. (1999). One-Factor-at-a-Time versus Designed Experiments. *Am. Stat.*, **53**(2), 126.

Fortin, F.-A. *et al.* (2012). DEAP: Evolutionary Algorithms Made Easy François-Michel De Rainville. *J. Mach. Learn. Res.*, **13**, 2171–2175.

Gilman, J. *et al.* (2019). Rapid, Heuristic Discovery and Design of Promoter Collections in Non-Model Microbes for Industrial Applications. *ACS Synth. Biol.*, **8**(5), 1175–1186.

Hammer, K. *et al.* (2006). Synthetic promoter libraries - Tuning of gene expression. *Trends Biotechnol.*, **24**(2), 53–55.

Hanahan, D. (1983). Studies on transformation of *Escherichia coli* with plasmids. *J. Mol. Biol.*, **166**(4), 557–580.

Harris, C. R. *et al.* (2020). Array programming with NumPy. *Nature*, **585**(7825), 357–362.

Hartmans, S. *et al.* (1989). Metabolism of Styrene Oxide and 2-Phenylethanol in the Styrene-Degrading *Xanthobacter* Strain 124X. *Appl. Environ. Microbiol.*, **55**(11), 2850—-2855.

Jensen, P. R. *et al.* (1993). The use of lac-type promoters in control analysis. *Eur. J. Biochem.*, **211**(1-2), 181–191.

Kalisky, T. *et al.* (2007). Cost–benefit theory and optimal design of gene regulation functions. *Phys. Biol*, **4**, 229–245.

Köbbing, S. *et al.* (2020). Characterization of Context-Dependent Effects on Synthetic Promoters. *Front. Bioeng. Biotechnol.*, **8**, 551.

Kochanowski, K. *et al.* (2017). Few regulatory metabolites coordinate expression of central metabolic genes in *Escherichia coli*. *Mol. Syst. Biol.*, **13**(1), 903.

Liu, D. *et al.* (2018). Construction, Model-Based Analysis, and Characterization of a Promoter Library for Fine-Tuned Gene Expression in *Bacillus subtilis*. *ACS Synth. Biol.*, **7**(7), 1785–1797.

Liya, D. H. *et al.* (2021). QPromoters: Sequence based prediction of promoter strength in *Saccharomyces cerevisiae*. *bioRxiv*, page 2021.04.27.441621.

Meng, H. *et al.* (2013). Quantitative Design of Regulatory Elements Based on High-Precision Strength Prediction Using Artificial Neural Network. *PLoS One*, **8**(4), e60288.

Meng, H. *et al.* (2017). Construction of precise support vector machine based models for predicting promoter strength. *Quant. Biol.*, **5**(1), 90–98.

Mutalik, V. K. *et al.* (2013). Precise and reliable gene expression via standard transcription and translation initiation elements. *Nat. Methods*, **10**(4), 354–360.

Paget, M. (2015). Bacterial sigma factors and anti-sigma factors: structure, function and distribution. *Biomolecules*, **5**(3), 1245–1265.

Rhodius, V. A. and Mutalik, V. K. (2010). Predicting strength and function for promoters of the *Escherichia coli* alternative sigma factor E. *Proc. Natl. Acad. Sci.*, **107**(7), 2854–2859.

Rhodius, V. A. *et al.* (2012). Predicting the strength of UP-elements and full-length *E. coli* σE promoters. *Nucleic Acids Res.*, **40**(7), 2907–2924.

Schweidtmann, A. M. and Mitsos, A. (2019). Deterministic Global Optimization with Artificial Neural Networks Embedded. *J. Optim. Theory Appl.*, **180**(3), 925–948.

Schweidtmann, A. M. *et al.* (2020). Global Optimization of Gaussian processes. *arXiv*, page arXiv:2005.10902.

Tareen, A. and Kinney, J. B. (2020). Logomaker: beautiful sequence logos in Python. *Bioinformatics*, **36**(7), 2272–2274.

Thebelt, A. *et al.* (2020). Global Optimization with Ensemble Machine Learning Models. *Comput. Aided Chem. Eng.*, **48**, 1981–1986.

Truong, A. *et al.* (2019). Towards Automated Machine Learning: Evaluation and Comparison of AutoML Approaches and Tools. In *2019 IEEE 31st Int. Conf. Tools with Artif. Intell.*, pages 1471–1479. IEEE.

Tsamardinos, I. *et al.* (2020). Just Add Data: Automated Predictive Modeling and BioSignature Discovery. *bioRxiv*.

Wolpert, D. H. and Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Trans Evol Comput*, **1**(1), 67–82.

Zaslaver, A. *et al.* (2009). Invariant Distribution of Promoter Activities in *Escherichia coli*. *PLoS Comput Biol*, **5**(10), e1000545.

Zhao, M. *et al.* (2020). Machine learning-based promoter strength prediction derived from a fine-tuned synthetic promoter library in *Escherichia coli*. *bioRxiv*, page 2020.06.25.170365.

Zobel, S. *et al.* (2015). Tn7-Based Device for Calibrated Heterologous Gene Expression in *Pseudomonas putida*. *ACS Synth. Biol.*, **4**(12), 1341–1351.