frontiers | Frontiers in Genetics

TYPE Original Research
PUBLISHED 09 May 2024
DOI 10.3389/fgene.2024.1371607

OPEN ACCESS

EDITED BY
Rosalba Giugno,
University of Verona, Italy

REVIEWED BY
Advait Balaji,
Occidental Petroleum Corporation,
United States
Zhi-Ping Liu,
Shandong University, China

*CORRESPONDENCE
Dennis Shasha,
✉ shasha@cims.nyu.edu

RECEIVED 16 January 2024
ACCEPTED 17 April 2024
PUBLISHED 09 May 2024

CITATION
Shen B, Coruzzi GM and Shasha D (2024), Bipartite networks represent causality better than simple networks: evidence, algorithms, and applications.
Front. Genet. 15:1371607.
doi: 10.3389/fgene.2024.1371607

Bipartite networks represent causality better than simple networks: evidence, algorithms, and applications

Bingran Shen[1], Gloria M. Coruzzi[2] and Dennis Shasha[1]*

[1]Courant Institute of Mathematical Sciences, Department of Computer Science, New York University, New York, United States, [2]Center for Genomics and Systems Biology, Department of Biology, New York University, New York, United States

A network, whose nodes are genes and whose directed edges represent positive or negative influences of a regulatory gene and its targets, is often used as a representation of causality. To infer a network, researchers often develop a machine learning model and then evaluate the model based on its match with experimentally verified “gold standard” edges. The desired result of such a model is a network that may extend the gold standard edges. Since networks are a form of visual representation, one can compare their utility with architectural or machine blueprints. Blueprints are clearly useful because they provide precise guidance to builders in construction. If the primary role of gene regulatory networks is to characterize causality, then such networks should be good tools of prediction because prediction is the actionable benefit of knowing causality. But are they? In this paper, we compare prediction quality based on “gold standard” regulatory edges from previous experimental work with non-linear models inferred from time series data across four different species. We show that the same non-linear machine learning models have better predictive performance, with improvements from 5.3% to 25.3% in terms of the reduction in the root mean square error (RMSE) compared with the same models based on the gold standard edges. Having established that networks fail to characterize causality properly, we suggest that causality research should focus on four goals: (i) predictive accuracy; (ii) a parsimonious enumeration of predictive regulatory genes for each target gene *g*; (iii) the identification of disjoint sets of predictive regulatory genes for each target *g* of roughly equal accuracy; and (iv) the construction of a bipartite network (whose node types are genes and models) representation of causality. We provide algorithms for all goals.

KEYWORDS

RNA sequencing, gene regulatory network, causal inference, random forest, bipartite network

1 Background and motivation

A frequent goal of expression-based causality research is to construct a directed graph of genes having some inductive and some repressive edges. One of the popular approaches to solve the gene regulatory inference problem is to build some kind of regression models to fit the gene expression data and make regulatory relation inference based on the model parameters. Some of these regression-based methods use linear regression like TIGRESS (Haury et al., 2012), SCODE (Matsumoto et al., 2017), and Inferelator (Skok Gibbs et al., 2022), while others like GENIE3 (Huynh-Thu et al., 2010), BiXGBoost (Zheng et al., 2019), OutPredict (Cirrone et al., 2020), and SCENIC (Van de Sande et al., 2020) choose non-linear regression models for the same goal. The main metric of these methods is conformance to some gold standard (GS) network. However, let us consider the actionable result of such research: to influence the behavior of an organism to make it more useful (e.g., more drought-tolerant crop or one with higher nutrient yield).

This paper asks the question "Are networks a good representation for actionable insights?" Because the edges are simple edges between pairs of regulatory and target genes, the network representation does not suggest any kind of synergy between the putative causal regulatory genes, e.g., transcription factors. So, the natural model choice for a given target gene g given the network is a linear model on the genes pointing to g.

Any causality model should be able to make reasonably accurate predictions. In Newtonian mechanics, e.g., a model involving mass and gravity will be able to predict the speed curve of a ball on an inclined plane. Predictive accuracy is not **sufficient** to establish causality. Some mysterious force might cause the ball to move in that way, but a causal model should be predictive. We consider predictive accuracy to be a **necessary** condition of a causal model.

We now consider several approaches to prediction.

1. Starting with all transcription factors (TFs) as possible causal features, both a non-linear random forest (RF)-style model M_{nonlin} and a linear model M_{lin} are tried.
2. Based on the random forest model mentioned above, a minimal set of TFs that could produce similar regression results in a model $M_{minimal}$ are iteratively searched for.
3. On the edges of a gold standard network for some target gene g, a non-linear model M_{nonlin} or a linear model M_{lin} is used.
4. A random forest model that uses the same number of TFs for each target g is formed as known from the gold standard network. Those TFs are chosen according to their feature importance starting from M_{nonlin} in approach 1 above.

As discussed later, (i) the non-linear models work better than the linear models and (ii) starting with all transcription factors and then shrinking that set based on model accuracy is better than using the gold standard network.

Our models are all based on time series in which we predict the mRNA expression level of the target gene based on the state of regulatory genes at the previous time point. This agrees with the biological intuition that the state of causal regulatory genes takes at least minutes to affect their targets. One implication of this approach to model building is that if a transcription factor T and a target gene g are correlated (e.g., they rise in the same time points and fall in the same time points), T will **not** be identified as causal. On the other hand, if T rising (respectively, falling) at one time point were associated with g rising (respectively, falling) at the next time point, then causality might be hypothesized.

1.1 Contributions

Our novel contribution is a framework, algorithms, and software for encoding possible causality in transcriptional settings into a bipartite directed graph. Our framework consists of the following workflow:

1. A machine learning method M that predicts the behavior of a target gene g starting with all possible causal regulatory elements (transcription factors for genomic networks) as candidates is chosen. M may be statistical, a neural network, a forest, or a linear model. We do not advocate any particular model, although non-linear models generally have lower errors than linear ones.
2. The set of possible causal regulatory elements for g is reduced to a smaller set S_1 that provides statistically equal (based on the p-value) accuracy still based on M.
3. Inspired by an observation of the statistician Efron (2020), a set D of mutually disjoint sets $S_1, S_2 \dots S_n$ is found that all provide statistically the same accuracy as S_1 in predicting the expression of some target gene g, possibly by training a new model for each S_i. D may contain S_1 alone or may contain many sets.
4. A visual representation of these mutually disjoint subsets of regulatory elements is provided for each given target gene g. The visualization consists of a bipartite graph in which each transcription factor of the disjoint subsets (S_1, $S_2 \dots S_n$) of transcription factors feeds a model node whose output is the target gene g.

2 Materials and methods

2.1 Expression prediction setup

All the experiments carried out in this study focus on time series RNA sequencing (RNA-seq) data because gene regulation through transcription factors is a temporal causal process. Following this logic, we build regression models that predict the expression of each target gene based on the TF expression levels from a previous time point in the time series. Formally, suppose we are given time series RNA-seq data consisting of sequencing data from time points $t_0, t_1, t_2, \dots, t_i, \dots, t_n$. We use RNA-seq data at time t_i to predict the expression of a target gene at t_{i+1} $(i \geq 0, i < n)$.

In order to split the whole time series into training/testing sets for validating the prediction quality of different methods, we chose to always reserve the tail end of the time series for testing while using the preceding part in training. More specifically, if $n < 5$, then only t_n will be used as the test sample with t_{n-1} being the input. If $5 \leq n < 10$, t_n and t_{n-1} are reserved for testing, with t_{n-1} and t_{n-2} as model input. If $n \geq 10$, then the final three in the series constitute the testing set.

2.2 RNA sequencing data used

Bulk time series RNA-seq data from four different species with varying experimental setups were used, totaling more than 100 data points for each species. The experimental sources for each species are given here. For the training/testing setups below, we train on a prefix of the time points and test on the remaining time points.

TABLE 1 Information on the gold standard (GS) networks used in this study. The number of target genes and transcription factors are for genes that are both present in the regulatory network and the RNA-seq data for each species, respectively.

Species	GS network **source**	Number of **target genes**	Number of **TFs**	Number of **regulations**
Yeast	YEASTRACT (Teixeira et al., 2023)	4,794	213	162,100
B. subtilis	SubtiWiki (Pedreira et al., 2022)	1,878	146	3,973
Arabidopsis	ConnecTF (Brooks et al., 2021)	18,855	57	141,445
Mice	RegNetwork (Liu et al., 2015)	8,211	780	40,331
Humans	RegNetwork (Liu et al., 2015)	17,533	1,351	132,259

1. *Saccharomyces cerevisiae* (yeast): data from GSE145936 (Feder et al., 2021), GSE153609 (Mitra et al., 2021; Tran et al., 2021), GSE168699 (Li et al., 2021), and GSE226769 (Harris and Ünal, 2023) were aggregated into a gene expression dataset with 144 training samples and 58 testing samples.
2. *Bacillus subtilis* (strain 168) (*B. subtilis*): data from GSE108659 (Krawczyk et al., 2015), GSE128875 (Pisithkul et al., 2019), and GSE224332 were aggregated into a gene expression dataset with 84 training samples and 18 testing samples.
3. *Arabidopsis thaliana* (*Arabidopsis*): data from GSE97500 (Varala et al., 2018; Heerah et al., 2021) were used with 72 training samples and 24 testing samples.
4. *Mus musculus* (mouse): data from GSE115553 (Graham et al., 2018), GSE151173 (Greenwell et al., 2020), and GSE171975 (Aviram et al., 2021) were aggregated into a gene expression dataset with 208 training samples and 121 testing samples.
5. *Homo sapiens* (human): data from GSE221103 and GSE221173 (Cazarin et al., 2023) were aggregated into a gene expression dataset consisting of 109 training samples and 40 testing samples.

Thus, our study derives from 4 well-studied living organisms ranging from bacteria to humans. We sought datasets having time series RNA-seq data with relatively tight timing intervals (no greater than 4 h in all cases) and suitably long series (≥4 time points) to form training/testing splits. We collected as much public bulk RNA-seq data about the four species as possible given the above constraints. Because the data came from widely different species (from bacteria to human), we expect that our qualitative conclusions will be generalizable. For each species, we obtained a GS regulatory network from the sources given in Table 1.

2.3 Metrics

Because RNA-seq counts are strongly dependent on the amount of cellular material that is read, relative expression is a better metric to determine induction or repression than absolute expression. For that reason, we measure expression based on the z-score of the normalized RNA-seq counts in the form of transcripts per kilobase million (TPM):

$$z = \frac{TPM - \mu}{\sigma}. \tag{1}$$

To compare the performance of each method, we measure how accurate the regression results were by checking the error of the prediction on the test set for each of the target genes. More specifically, the root mean square error (RMSE) of the model prediction in the test set for the expression of each target gene was compared across different regression models. Because every regression model was trained/fitted to make predictions on the same set of time series expression samples for each target gene in question, the performance metrics can be compared based on a paired test. For this purpose, we use a non-parametric paired test (Katari et al., 2021).

2.4 Methods compared

For the purpose of predicting the expression of each target gene on a future unseen time point, we fitted four different types of regression models, as described above:

1. An RF model that takes the expression of all TFs as input.
2. A ridge regression (linear regression with L2 regularization) model that takes the expression of all TFs as input.
3. A random forest model that takes only the expression of TFs known from the GS network for each particular target gene as input.
4. A ridge regression model that takes only the expression of TFs known from the GS network for each particular target gene as input.

Next, we test how good the transcription factors from the GS network are compared to the same number of transcription factors derived from a non-linear model. For each target gene g, let k_g be the number of transcription factors in the GS network that point to g. In addition to the tests above, we compare a random forest on those GS TFs against a random forest for g based on the top k_g TFs found using method 1 above. The idea is to test the usefulness of GS edges for prediction. One may argue that GS edges are inferred using different methods—usually by modifying single regulatory genes and observing their effect—and, therefore, should not necessarily be useful for prediction but could still be useful if modifying a single gene is all that is possible for practical reasons. We do not contest their utility for such purposes. We do, however, aim to evaluate their predictive power in a synergistic setting (i.e., when potentially several regulatory genes can be simultaneously modified).

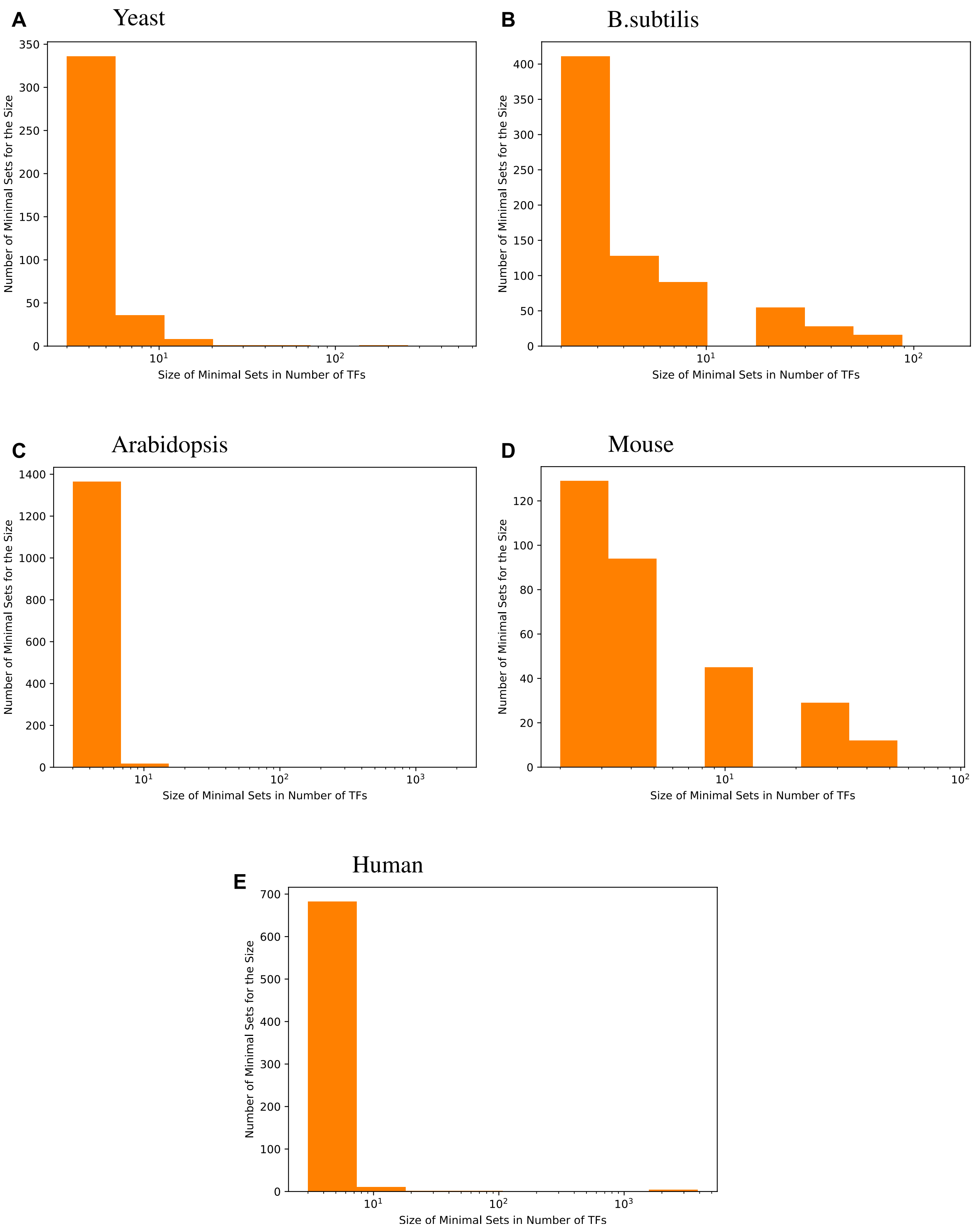

FIGURE 1
Size distribution of a minimal transcription factor (TF) set for each target gene. For every species, the majority of genes are best predicted by fewer than 8 transcription factors, i.e., 96.6% for yeast, 73.5% for *B. subtilis*, 98.7% for *Arabidopsis*, 71.9% for mice, and 97.7% for humans.

TABLE 2 Gold standard network edge coverage per target gene for different RNA-seq species compared with the size of minimal transcription factor set sizes per target gene, derived using random forest regression. Distributions are presented as the median followed by the interquartile range (IQR), which is the range between the 25th and 75th percentile of the data. The one exception is yeast, where the gold standard edges are much more numerous. As shown in the tables above, the minimal sets generally have better predictive power than the gold standard sets and roughly the same number of edges per target.

Median [IQR]	Yeast	*B. subtilis*	*Arabidopsis*	Mice	Humans
Size of the TF set per target in the gold standard network	51 [41, 63]	1 [1, 2]	13 [6, 19]	4 [2, 7]	8 [4, 14]
Size of the minimal TF set per target using RF regression	3 [3, 3]	2 [2, 9]	3 [3, 3]	5 [2, 10]	3 [3, 3]
Number of target genes	385	733	1,373	310	698

Finally, using the method given in Section 3, we construct a minimal random forest model for each target gene *g* on the training set and view its result on the test set. We choose random forest because decision tree-based regression models have proven to be among the best methods in gene regulatory tasks (Huynh-Thu et al., 2010; Moerman et al., 2019; Zheng et al., 2019). We did not expand our model selection because the main focus of our work is not to find the best fitted machine learning model for the task but rather to demonstrate a novel approach to the representation of potential causality in gene regulation.

3 Algorithms to construct bipartite causality graphs

Having chosen prediction as the metric for causality, we now turn to the other three goals of our proposed framework:

1. Finding minimal sets of predictive TFs.
2. Finding disjoint minimal sets that have *p*-value-indistinguishable predictive accuracy.
3. Creating a bipartite visual representation of causality.

3.1 Minimal sets of predictive transcription factors

Efron (2020) noted that disjoint sets of causal factors often have similar predictive accuracy. Inspired by this observation, we propose the following strategy. For a given target gene, a random forest predictor that takes the expression levels of all known TFs is fit to predict the expression of the target. Then, the number of TFs are iteratively cut in half based on their feature importance in the fitted model until a further reduction results in a statistically significantly (p-value < 0.05) worse-performing random forest. We refer to the final remaining set of TFs as the “minimal TF set per target.” The pseudo-code for this feature selection process is shown in Algorithm 1.

The histograms given in Figure 1 show the distribution of the size of minimal TF sets yielded for each target gene for the four species we investigated. These size distributions show that most of the minimal sets consist of a rather small number of TFs. When compared with the distribution of GS network coverage for each target gene in Table 2, we see that an accurate regression model constructed this way usually has fewer input transcription factors compared to the GS networks.

```
function MINIMALSET($G$, $TFs$, $E$)
    $F \leftarrow TFs$
    $M_{all} \leftarrow$ initialized regression model
    Fit $M_{all}$ with $F$ to predict $G$
    if $E > 0$ then
        $E_{baseline} \leftarrow E$
    else▷ $E == 0$ implies that no error value has been calculated yet
        $E_{baseline} \leftarrow$ training Error of $M_{all}$
    $F_{half} \leftarrow$ top half most influential TFs used in $M_{all}$
    $flag \leftarrow True$
    while $flag == True$ do
        $M_{current} \leftarrow$ initialized regression model
        Fit $M_{current}$ with $F_{half}$ to predict target gene $G$
        $E_{current} \leftarrow$ training Error of $M_{current}$
        if not $E_{current} > E_{baseline}$ with statistical significance then
            $F \leftarrow F_{half}$
            $F_{half} \leftarrow$ top half most influential TFs used in $M_{current}$
        else
            $flag \leftarrow False$
    return $F$
```

Algorithm 1. Minimal TF set per target: For each target gene *G*, initial set of TFs, and RMSE *E*, the set of necessary transcription factors are repeatedly reduced by half until the error increases significantly with respect to *E*. A minimal set of TFs for a given target gene *G* will be a call to this function MinimalSet(*G*, *all TFs*, 0).

3.2 Finding disjoint sets of predictive transaction factors

After finding a minimal set of predictive TFs, our algorithm performs a new round of TF searches to discover disjoint sets of roughly equally predictive TFs. Algorithm 2 describes the process for finding all such disjoint sets of a given target gene. Similar to the minimal set search algorithm, we based our iterative search on the random forest regression that takes all available TFs *U* as input. Rather than stopping after a minimal set *S*1 is found, we test if using all remaining TFs ($U - S1$) could also produce a regression prediction as good as the baseline. If that is the case, we carry on a similar feature reduction process that ends with a new “minimal set” *S*2 from $U - S1$. This process then repeats with $U - (S1 \cup S2)$ and continues until the baseline performance cannot be beaten or there are no TFs left. For each target gene *g*, we define the collection of

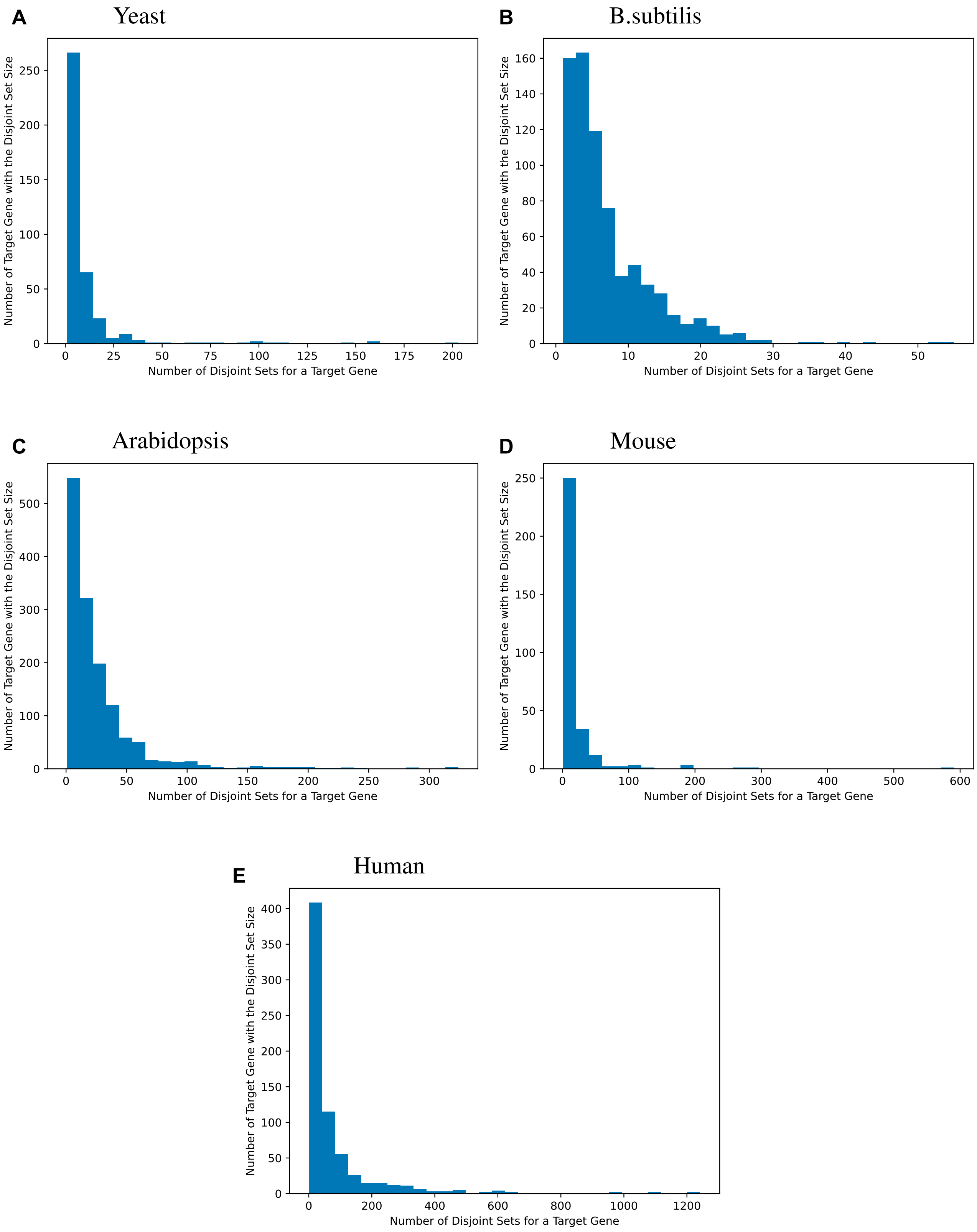

FIGURE 2
Distribution of the disjoint set count for each target gene. Many target genes can be best modeled by a handful of an explanatory set of TFs. For yeast, 44.2% of the target genes are best modeled by fewer than 5 explanatory disjoint sets of TFs, For *B. subtilis*, 44.1%, for *Arabidopsis*, 15.2%, for mice, 25.5%, and for humans, 10.3%. Sometimes, many disjoint sets of TFs are redundant. Bipartite graphs capture this causality information.

minimal sets discovered this way as the *minimal disjoint sets of predictive transcription factors for g* or *MinDisjoints(g)* for short.

We then surveyed the distribution of how many MinDisjoints are found for each target gene across the four species. The histograms given in Figure 2 show that most of the target genes have a small number of disjoint sets of TFs associated with them, while some target genes have a large number of MinDisjoints. Our analysis did not yield biological mechanisms for predictability/causality, so we have no mechanistic explanation for how multiple disjoint sets of TFs might control the same target gene. However, the result was not wholly unexpected, given the well-known redundancy in biological systems.

```
$D \leftarrow$ empty list
$F \leftarrow$ All TFs
$M_{all} \leftarrow$ initialized regression model
Fit $M_{all}$ with $F$ to predict target gene $G$
$E_{all} \leftarrow$ training Error of $M_{all}$
$F_m \leftarrow$ MINIMALSET($G$, $F$, $E_{all}$)
Add $F_m$ to $D$
$F_r \leftarrow F \setminus F_m$
$flag \leftarrow True$
while $F_r \neq \emptyset$ and $flag == True$ do
  $M_r \leftarrow$ initialized regression model
  Fit $M_r$ with $F_r$ to predict target gene $G$
  $E_r \leftarrow$ training Error of $M_r$
  if $E_r > E_{all}$, with statistical significance then
    $flag \leftarrow False$
    break
  $F \leftarrow$ MINIMALSET($G$, $F_r$, $E_{all}$)
  Add $F$ to $D$
  $F_r \leftarrow F_r \setminus F$
For a given target gene $G$, $D$ will be the set of disjoint sets of TFs $G$.
```

Algorithm 2. Disjoint sets of TFs: Finding minimal sets of disjoint TFs (MinDisjoints), where each minimal set has the same error as using all TFs.

3.3 Bipartite network representation

Networks have a pleasing visual representation, especially when focusing on one or a few target genes. However, what we showed is that the network itself is a poor basis for prediction. Now that we have constructed multiple disjoint sets of predictive TFs for each target gene *g*, we propose a bipartite representation for them. The bipartite representation for each target gene *g* consists of a model node *m* corresponding to each disjoint set d_m from D(*g*). The TFs from d_m in turn point to *m*.

Suppose that TFs A, B, and C through model M(A, B, C) provide good predictions regarding target gene *g*. Suppose further that TFs D, E, F, and H provide roughly equally good predictions on *g*. The classic gene regulatory approach would be a graph with arrows from A, B, C, D, E, F, and H all pointing to *g*. The bipartite approach would suggest instead to show a bipartite graph that would have A, B, and C point to a model node, which, in turn, points to g, and have D, E, F, and H point to a different model node, which also points to g.

To demonstrate this new representation, we picked one example for each species we studied, as shown in Figure 3. Here, we specifically highlighted one interesting scenario: a set of TFs were found to form one of the disjoint sets for more than one target gene. Such a relationship between two genes would not have been found in a simple network representation. The bipartite representation reveals group effects that would not otherwise be evident.

4 Results

4.1 Comparison of approaches

Figure 4 shows the accuracy of the six different modeling approaches listed in Section 2.4. Basically, feeding expression information from all the TFs into a random forest ("RF with all TFs") yielded the best outcome. Relying solely on known GS edges ("RF with GS TFs") usually performed poorly, even compared to using the same number of TFs for each target gene from the random forest model ("RF with top TFs").

We note that linear models on all TFs are competitive and sometimes better than random forests on minimal TF sets for *B. subtilis* and mouse. Still, overall, given the same input information, random forests perform better than linear models, which is the main point of that comparison.

Tables 3–7 list the detailed pairwise non-parametric results comparing the performance of all possible pairs of models. The tables show that using all TFs in the regression yields the highest prediction accuracy. Finding a minimal set of the most important TFs yields almost the same accuracy as using all TFs.

A question to ask is what biological meaning disjoint sets of transcription factors could have for a given target gene *g*. Our computational analysis does not provide a biological meaning other than predictive ability. Experimentalists might take various disjoint sets of TFs and manipulate them to achieve some desired effect on a target gene. The choice of such sets may depend on the side effects such manipulation might have on other genes. This is a direction for future work.

4.2 Batch effects

While the z-score takes care of quantity bias in different tests, batch effects may cause predictions on batch A based on data from batch A to be superior to predictions on batch A from data on many batches. This is a limitation of any predictive model in biology.

To test this, we created our models based on multiple batches and tested them on the tail end of all those batches. We compared that approach with batch-by-batch predictions. Figure 5 shows that the same model trained on all batches of data achieves the same level or better predictive performance than when using batch X data on batch X tail, for each batch X.

4.3 Ensemble of disjoint sets of transcription factors

Another potential use of the identification of disjoint sets of predictive TFs stems from the fact that each disjoint set represents a regression model for the prediction of the target gene. For all the

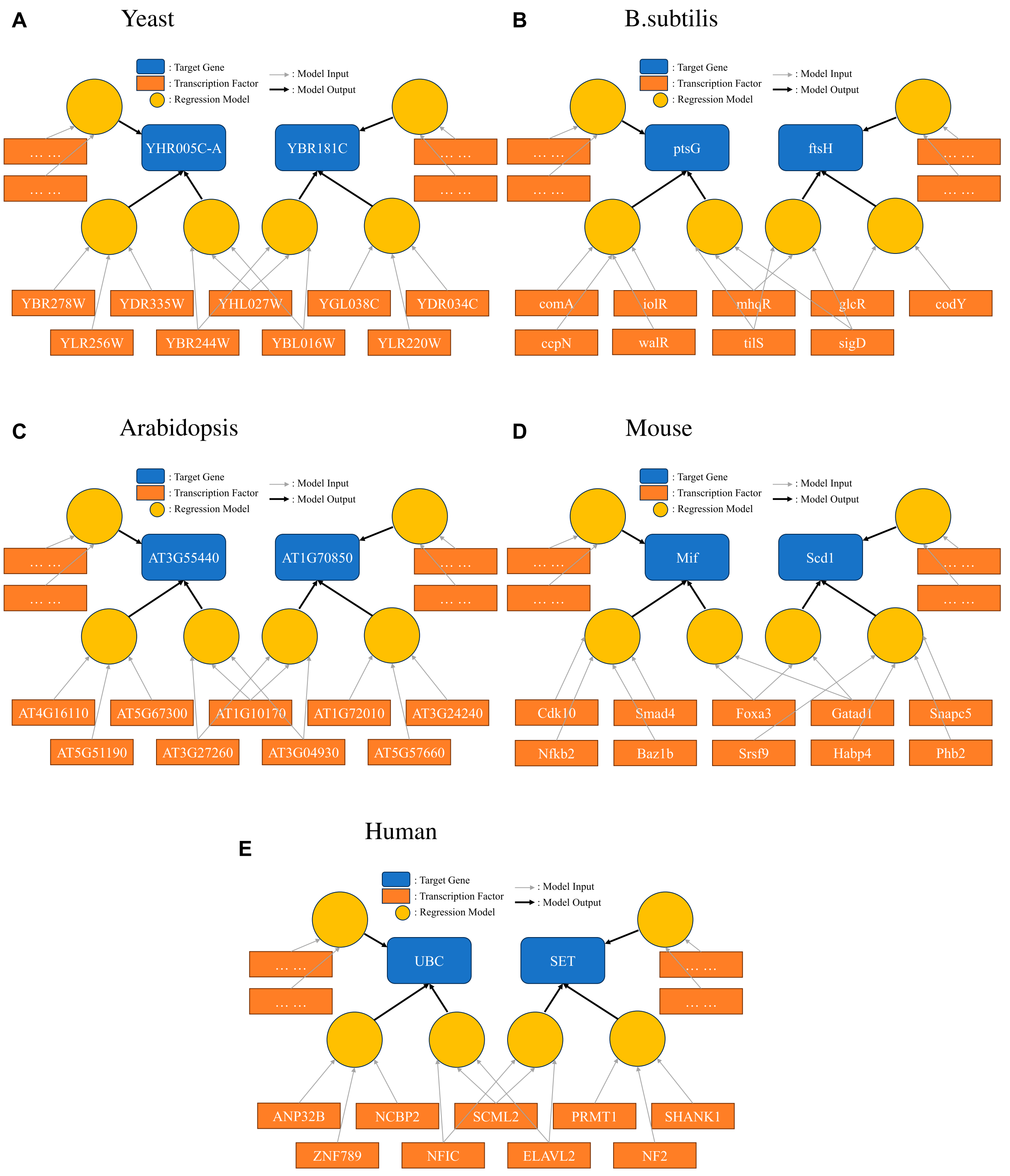

FIGURE 3
Bipartite representation of causality. Light circular orange nodes represent non-linear models that take transcription factors (dark orange rectangles) as inputs and produce predictions on a single target gene (blue). Here, we show a particular case where disjoint sets of TFs can form high-quality prediction models for one target gene, and the same TF can be in models for several target genes.

disjoint sets, we found a target gene *g*, and the regression model of each disjoint set can provide a prediction about the expression of the target gene, given the expression input of the TFs at the previous time point. As has been shown in many previous studies in both general machine learning and gene network inference (Dietterich, 2000; Marbach et al., 2012; Sagi and Rokach, 2018; Ganaie et al., 2022), an ensemble consisting of the arithmetic mean of these model predictions may lead to an overall better performing prediction. Inspired by those results, we compared the predictive performance of this ensemble of disjoint sets of TFs to that of all other RF-based

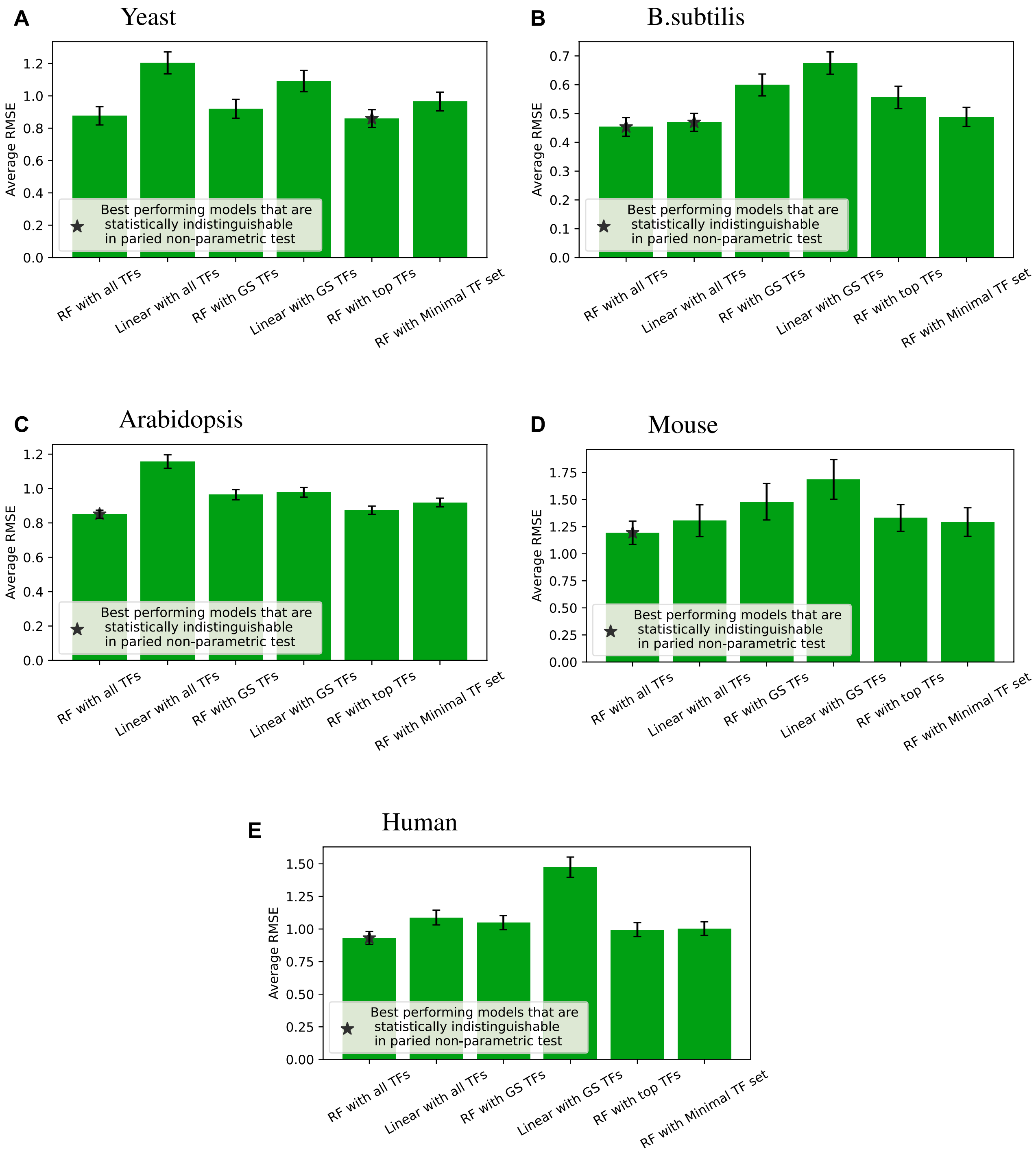

FIGURE 4
Root mean square error (RMSE) performance (lower is better) of different regression models compared across four species; error bars represent the standard error of each group. When we compare all the models for each of the tested target genes, a paired non-parametric test can be applied between each pair of models to see if the performance is statistically different. The best performing models that are statistically indistinguishable this way are marked with *. "RF with all TFs" is always one of the best performing models. The caption of Table 2 provides a glossary of terms.

regressions we discussed before, and the results are given in Table 8. In most cases, this ensemble yielded regression accuracies second only to the model that takes all TFs as input.

A complete list of all the minimal sets and disjoint sets of TFs for each target gene we surveyed in this study is given in Supplementary Table S1–S5.

4.4 An application: optimizing gene expression

Suppose our goal is to cause a gene g to be expressed at a certain level. We observed that the GS network, even when available, provides quite poor predictions. A better approach is to start with a good

TABLE 3 *S. cerevisiae* (yeast): Paired non-parametric results for the performance comparison on the test set between the model in blue and orange measured using root mean square error (RMSE) on target gene expression in the test datasets. Entry (*i, j*) shows the 95% confidence interval as the difference of the *i*th blue modeling method minus the *j*th orange modeling method. A negative number means the method in blue has a lower error and, hence, is better. When the difference in the blue method *i* with the red method *j* has a *p*-value below 0.05 based on a non-parametric paired test, the (*i, j*)th confidence interval will be red or blue. Otherwise, the (*i, j*)th entry will be black. Glossary: (i) TF = transcription factor; (ii) GS = gold standard; (iii) RF = random forest; (iv) linear = ridge regression; (v) RF with top TFs = for each target gene g, the same number of TFs from the random forest model are used as there were gold standard edges for g; and (vi) minimal TF = minimal set of most important TFs that provides *p*-value-indistinguishable results (on the training set) using all TFs.

	RF with all TFs	Linear with all TFs	RF with GS TFs	Linear with GS TFs	RF with top TFs	RF with the minimal TF set
Mean RMSE	0.877	1.204	0.920	1.091	0.859	0.965
RF with all TFs	-	(−0.397, −0.257)	(−0.055, −0.031)	(−0.263, −0.166)	(0.008, 0.027)	(−0.119, −0.058)
Linear with all TFs	(0.257, 0.397)	-	(0.214, 0.354)	(0.047, 0.178)	(0.275, 0.414)	(0.169, 0.309)
RF with GS TFs	(0.031, 0.055)	(−0.354, −0.214)	-	(−0.218, −0.124)	(0.046, 0.076)	(−0.075, −0.015)
Linear with GS TFs	(0.166, 0.263)	(−0.178, −0.047)	(0.124, 0.218)	-	(0.185, 0.279)	(0.081, 0.172)
RF with top TFs	(−0.027, −0.008)	(−0.414, −0.275)	(−0.076, −0.046)	(−0.279, −0.185)	-	(−0.133, −0.079)
RF with the minimal TF set	(0.058, 0.119)	(−0.309, −0.169)	(0.015, 0.075)	(−0.172, −0.081)	(0.079, 0.133)	-

TABLE 4 *B. subtilis*: Paired non-parametric results for the performance comparison between the model in blue and orange measured using RMSE on target gene expression in the test datasets. Entry (*i, j*) shows the 95% confidence interval as the difference of the *i*th blue modeling method minus the *j*th orange modeling method. A negative number means the method in blue has a lower error and, hence, is better. When the difference in the blue method *i* with the red method *j* has a *p*-value below 0.05 based on a non-parametric paired test, the (*i, j*)th confidence interval will be red or blue. Otherwise, the (*i, j*)th entry will be black. The caption of Table 2 provides a glossary of terms.

	RF with all TFs	Linear with all TFs	RF with GS TFs	Linear with GS TFs	RF with top TFs	RF with the minimal TF set
Mean RMSE	0.454	0.469	0.599	0.675	0.556	0.489
RF with all TFs	-	(−0.035, 0.003)	(−0.168, −0.122)	(−0.247, −0.196)	(−0.124, −0.080)	(−0.053, −0.017)
Linear with all TFs	(−0.003, 0.035)	-	(−0.161, −0.098)	(−0.243, −0.168)	(−0.113, −0.060)	(−0.037, −0.001)
RF with GS TFs	(0.122, 0.168)	(0.098, 0.161)	-	(−0.099, −0.054)	(0.015, 0.071)	(0.080, 0.141)
Linear with GS TFs	(0.196, 0.247)	(0.168, 0.243)	(0.054, 0.099)	-	(0.086, 0.153)	(0.153, 0.221)
RF with top TFs	(0.080, 0.124)	(0.060, 0.113)	(−0.071, −0.015)	(−0.153, −0.086)	-	(0.044, 0.091)
RF with the minimal TF set	(0.017, 0.053)	(0.001, 0.037)	(−0.141, −0.080)	(−0.221, −0.153)	(−0.091, −0.044)	-

TABLE 5 *Arabidopsis*: Paired non-parametric results for the performance comparison between the model in blue and orange measured using the RMSE on target gene expression in the test datasets. Entry (*i*, *j*) shows the 95% confidence interval as the difference of the *i*th blue modeling method minus the *j*th orange modeling method. A negative number means the method in blue has a lower error and, hence, is better. When the difference in the blue method *i* with the red method *j* has a *p*-value below 0.05 based on a non-parametric paired test, the (*i*, *j*)th confidence interval will be red or blue. Otherwise, the (*i*, *j*)th entry will be black. The caption of Table 2 provides a glossary of terms.

	RF with all TFs	Linear with all TFs	RF with GS TFs	Linear with GS TFs	RF with top TFs	RF with the minimal TF set
Mean RMSE	0.851	1.157	0.964	0.978	0.874	0.919
RF with all TFs	-	(−0.345, −0.268)	(−0.135, −0.092)	(−0.148, −0.108)	(−0.032, −0.014)	(−0.080, −0.057)
Linear with all TFs	(0.268, 0.345)	-	(0.159, 0.227)	(0.142, 0.215)	(0.245, 0.322)	(0.203, 0.274)
RF with GS TFs	(0.092, 0.135)	(−0.227, −0.159)	-	(−0.028, −0.001)	(0.069, 0.112)	(0.023, 0.067)
Linear with GS TFs	(0.108, 0.148)	(−0.215, −0.142)	(0.001, 0.028)	-	(0.086, 0.124)	(0.039, 0.081)
RF with top TFs	(0.014, 0.032)	(−0.322, −0.245)	(−0.112, −0.069)	(−0.124, −0.086)	-	(−0.057, −0.033)
RF with the minimal TF set	(0.057, 0.080)	(−0.274, −0.203)	(−0.067, −0.023)	(−0.081, −0.039)	(0.033, 0.057)	-

TABLE 6 Mice: Paired non-parametric results for the performance comparison between the model in blue and orange measured using the RMSE on target gene expression in the test datasets. Entry (*i*, *j*) shows the 95% confidence interval as the difference of the *i*th blue modeling method minus the *j*th orange modeling method. A negative number means the method in blue has a lower error and, hence, is better. When the difference in the blue method *i* with red method *j* has a *p*-value below 0.05 based on a non-parametric paired test, the (*i*, *j*)th confidence interval will be red or blue. Otherwise, the (*i*, *j*)th entry will be black. The caption of Table 2 provides a glossary of terms.

	RF with all TFs	Linear with all TFs	RF with GS TFs	Linear with GS TFs	RF with top TFs	RF with the minimal TF set
Mean RMSE	1.194	1.305	1.480	1.686	1.331	1.293
RF with all TFs	-	(−0.217, −0.006)	(−0.437, −0.136)	(−0.669, −0.316)	(−0.200, −0.074)	(−0.169, −0.029)
Linear with all TFs	(0.006, 0.217)	-	(−0.261, −0.088)	(−0.503, −0.259)	(−0.110, 0.059)	(−0.055, 0.080)
RF with GS TFs	(0.136, 0.437)	(0.088, 0.261)	-	(−0.290, −0.122)	(0.032, 0.266)	(0.072, 0.303)
Linear with GS TFs	(0.316, 0.669)	(0.259, 0.503)	(0.122, 0.290)	-	(0.203, 0.508)	(0.248, 0.539)
RF with top TFs	(0.074, 0.200)	(−0.059, 0.110)	(−0.266, −0.032)	(−0.508, −0.203)	-	(−0.025, 0.101)
RF with the minimal TF set	(0.029, 0.169)	(−0.080, 0.055)	(−0.303, −0.072)	(−0.539, −0.248)	(−0.101, 0.025)	-

TABLE 7 Humans: Paired non-parametric results for the performance comparison between the model in blue and orange measured using the RMSE on target gene expressions in the test datasets. Entry (*i*, *j*) shows the 95% confidence interval as the difference of the *i*th blue modeling method minus the *j*th orange modeling method. A negative number means the method in blue has a lower error so is better. When the difference in the blue method *i* with red method *j* has a *p*-value below 0.05 based on a non-parametric paired test, the (*i*, *j*)th confidence interval will be red or blue. Otherwise, the (*i*, *j*)th entry will be black. The caption of Table 2 provides a glossary of terms.

	RF with all TFs	Linear with all TFs	RF with GS TFs	Linear with GS TFs	RF with top TFs	RF with the minimal TF set
Mean RMSE	0.931	1.088	1.049	1.474	0.995	1.004
RF with all TFs	-	(−0.192, −0.122)	(−0.143, −0.093)	(−0.616, −0.469)	(−0.081, −0.047)	(−0.088, −0.056)
Linear with all TFs	(0.122, 0.192)	-	(0.005, 0.073)	(−0.457, −0.314)	(0.060, 0.126)	(0.052, 0.117)
RF with GS TFs	(0.093, 0.143)	(−0.073, −0.005)	-	(−0.489, −0.361)	(0.032, 0.076)	(0.019, 0.072)
Linear with GS TFs	(0.469, 0.616)	(0.314, 0.457)	(0.361, 0.489)	-	(0.413, 0.545)	(0.400, 0.541)
RF with top TFs	(0.047, 0.081)	(−0.126, −0.060)	(−0.076, −0.032)	(−0.545, −0.413)	-	(−0.029, 0.012)
RF with the minimal TF set	(0.056, 0.088)	(−0.117, −0.052)	(−0.072, −0.019)	(−0.541, −0.400)	(−0.012, 0.029)	-

TABLE 8 Paired non-parametric results for the performance comparison between the model in blue and orange measured using the RMSE on target gene expressions in the test datasets. Each column is a comparison for one of the four species that compared the ensemble prediction from disjoint sets of transcription factors (TFs) to other random forest (RF)-based methods. Entry (*i*, *j*) shows the 95% confidence interval as the difference of the *i*th blue modeling method minus the *j*th orange modeling method. A negative number means the method in blue has a lower error and, hence, is better. When the difference in the blue method *i* with the red method *j* has a *p*-value below 0.05 based on a non-parametric paired test, the (*i*, *j*)th confidence interval will be red or blue. Otherwise, the (*i*, *j*)th entry will be black. The caption of Table 2 provides a glossary of terms.

	Ensemble of disjoint sets in yeast	Ensemble of disjoint sets in *B. subtilis*	Ensemble of disjoint sets in *Arabidopsis*	Ensemble of disjoint sets in mice	Ensemble of disjoint sets in humans
RF with all TFs	(−0.058, −0.017)	(−0.009, 0.008)	(−0.009, 0.001)	(−0.038, 0.013)	(−0.021, −0.003)
RF with GS TFs	(0.227, 0.366)	(−0.003, 0.034)	(0.270, 0.349)	(0.034, 0.223)	(0.115, 0.181)
RF with top TFs	(−0.010, 0.027)	(0.122, 0.170)	(0.090, 0.133)	(0.169, 0.444)	(0.085, 0.134)
RF with the minimal TF set	(0.138, 0.231)	(0.196, 0.249)	(0.105, 0.146)	(0.350, 0.681)	(0.464, 0.610)

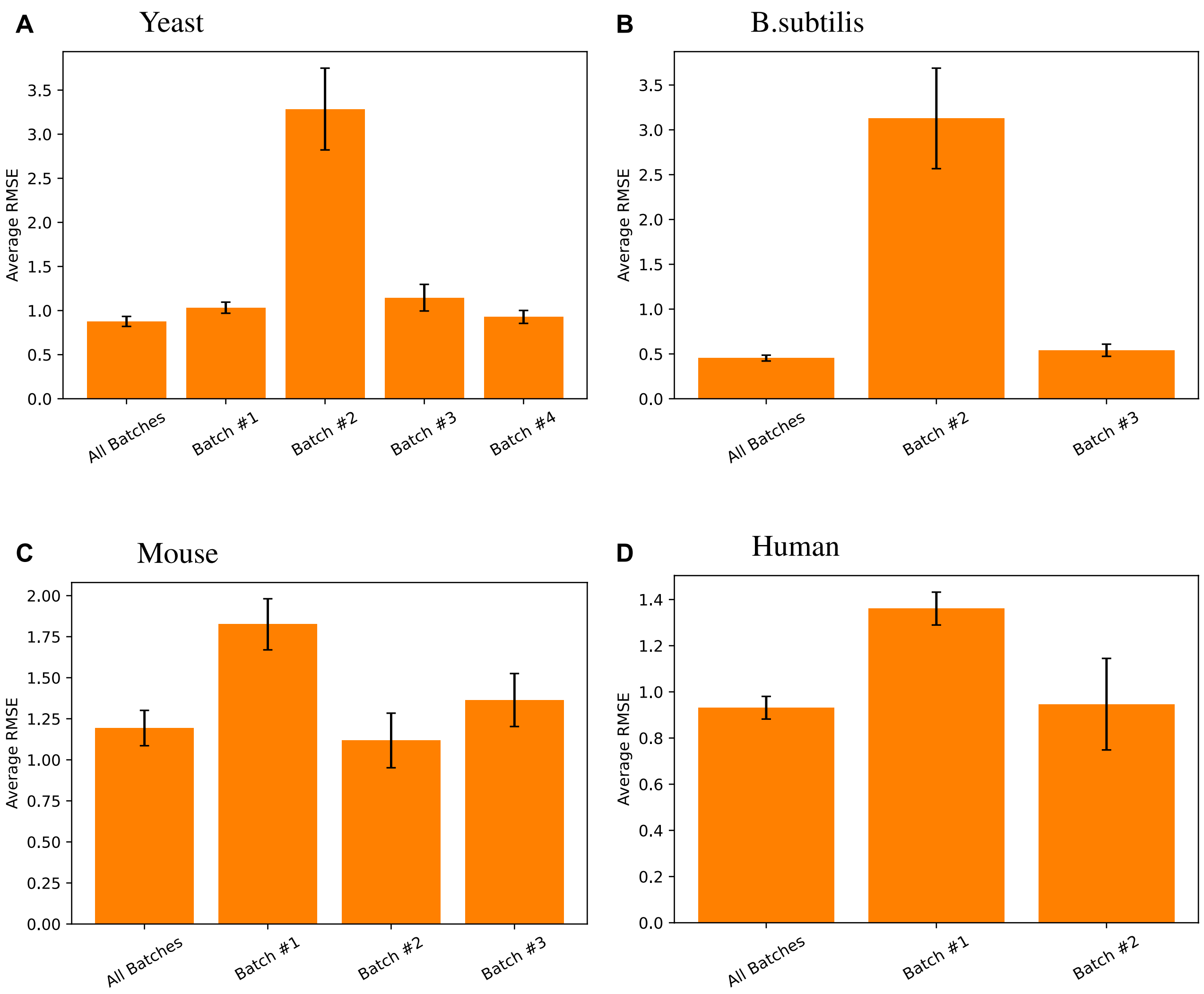

FIGURE 5
Random forest regression performance differences measured across different data batches for different species. Here, all batches results are default and presented relists shown in this work, where the random forest model using all TFs was trained on training data from all batches and tested on all the tail testing parts from different batches. The same model was then trained and tested on individual batches for its respective high-variance target genes. Note that for *Arabidopsis*, one singular batch was used, so such comparison was unnecessary, and for *B. subtilis*, the first batch does not have high-variance target genes in its testing set, so the comparison was also omitted.

predictive model for g on a small number of TFs T and then to determine values of the TFs in T that might lead to the desired expression level of g. This goal is supported by the three goals of our framework: to find a good model, reduce the number of TFs while preserving accuracy, and find possible alternative sets of TFs that also yield high prediction accuracy. Gene regulatory networks do not provide natural guidance for any goal like this.

Thus, the bipartite network approach provides an actionable approach to causality. At the same time, it provides (i) a visualization that shows alternative ways to manipulate a target gene and (ii) a simple ensemble approach to prediction.

5 Empirical findings

Our empirical findings are as follows:

- We confirm previous observations (Pratapa et al., 2020; Zhao et al., 2021) that non-linear models generally yield better results (as measured by RMSE) than linear models.
- Using all TFs yields better predictive results than using the TFs from GS edges. For each target gene *g*, there often exist several disjoint minimal sets (mostly of size eight or less) that yield predictive accuracy nearly as high as all TFs.
- Using all batches of each species together for training yields results on the time series test tails of each batch that are as good as or better than using each batch on its own test tail.
- For a given target gene *g*, forming a model consisting of the most influential k_g TFs in a non-linear model (e.g., random forests) as measured on the training set, where k_g is the number of TFs in the GS network that point to *g*, yields better prediction accuracy on the test set than using the same kind of model on the GS TFs. This superiority holds for all the species we tested from yeast with a

mean value of 53 TFs for each target gene to *B. subtilis* with a mean value of 1.9 TFs for each target gene.

6 Conclusion

Based on our empirical findings, we suggest a framework for studying causality in gene regulation having three main features.

First, the figure of merit for causality should be predictive accuracy rather than conformance with "gold standard" edges. One reason is epistemic: any causal model should be predictive. Another reason is pragmatic: prediction is useful if we want to manipulate some property such as the expression of a target gene.

Second, the network representation of such causality should be a bipartite graph consisting of gene (including transcription factor) nodes and model nodes. Such graphs encode the synergy of multiple TFs in the model nodes.

Third, the bipartite representation may include many model nodes that point to the same target gene, where each model node has a disjoint set of TFs as input. A single TF plays a role in disjoint sets of several target genes.

In addition to suggesting a modified approach to causality research for transcriptional regulation, we assert that our framework is applicable beyond transcriptional causality. Our main future work is to apply this form of analysis to other multifactor causality domains. We welcome other researchers to try this approach and offer our software to help.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found at: https://github.com/IcyFermion/feature_synergy.

Author contributions

BS: conceptualization, data curation, formal analysis, investigation, methodology, software, validation, visualization, writing–original draft, and writing–review and editing. GC: funding acquisition, project administration, resources, supervision, and writing–review and editing. DS: conceptualization, funding acquisition, investigation, methodology, project administration, resources, supervision, writing–original draft, and writing–review and editing.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This work was funded in part by NIH-NIGMS R01 GM121753 to GC. and DS by the Zegar Family Foundation A16-0051, and NYU Wireless.

Acknowledgments

The authors thank Manpreet Katari for his helpful suggestions.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fgene.2024.1371607/full#supplementary-material

References

Aviram, R., Dandavate, V., Manella, G., Golik, M., and Asher, G. (2021). Ultradian rhythms of akt phosphorylation and gene expression emerge in the absence of the circadian clock components per1 and per2. *PLoS Biol.* 19, e3001492. doi:10.1371/journal.pbio.3001492

Brooks, M. D., Juang, C.-L., Katari, M. S., Alvarez, J. M., Pasquino, A., Shih, H.-J., et al. (2021). Connectf: a platform to integrate transcription factor–gene interactions and validate regulatory networks. *Plant physiol.* 185, 49–66. doi:10.1093/plphys/kiaa012

Cazarin, J., DeRollo, R. E., Shahidan, S. N. A. B. A., Burchett, J. B., Mwangi, D., Krishnaiah, S., et al. (2023). Myc disrupts transcriptional and metabolic circadian oscillations in cancer and promotes enhanced biosynthesis. *PLoS Genet.* 19, e1010904. doi:10.1371/journal.pgen.1010904

Cirrone, J., Brooks, M. D., Bonneau, R., Coruzzi, G. M., and Shasha, D. E. (2020). Outpredict: multiple datasets can improve prediction of expression and inference of causality. *Sci. Rep.* 10, 6804. doi:10.1038/s41598-020-63347-3

Dietterich, T. G. (2000). "Ensemble methods in machine learning," in *International workshop on multiple classifier systems* (Cham: Springer), 1–15.

Efron, B. (2020). Prediction, estimation, and attribution. *Int. Stat. Rev.* 88, S28–S59. doi:10.1111/insr.12409

Feder, Z. A., Ali, A., Singh, A., Krakowiak, J., Zheng, X., Bindokas, V. P., et al. (2021). Subcellular localization of the j-protein sis1 regulates the heat shock response. *J. Cell Biol.* 220, e202005165. doi:10.1083/jcb.202005165

Ganaie, M. A., Hu, M., Malik, A., Tanveer, M., and Suganthan, P. (2022). Ensemble deep learning: a review. *Eng. Appl. Artif. Intell.* 115, 105151. doi:10.1016/j.engappai.2022.105151

Graham, D. B., Jasso, G. J., Mok, A., Goel, G., Ng, A. C., Kolde, R., et al. (2018). Nitric oxide engages an anti-inflammatory feedback loop mediated by peroxiredoxin 5 in phagocytes. *Cell Rep.* 24, 838–850. doi:10.1016/j.celrep.2018.06.081

Greenwell, B. J., Beytebiere, J. R., Lamb, T. M., Bell-Pedersen, D., Merlin, C., and Menet, J. S. (2020). Isoform-specific regulation of rhythmic gene expression by alternative polyadenylation. Available at: https://www.biorxiv.org/content/.

Harris, A., and Ünal, E. (2023). The transcriptional regulator ume6 is a major driver of early gene expression during gametogenesis. *Genetics* 225, iyad123. doi:10.1093/genetics/iyad123

Haury, A.-C., Mordelet, F., Vera-Licona, P., and Vert, J.-P. (2012). Tigress: trustful inference of gene regulation using stability selection. *BMC Syst. Biol.* 6, 145–217. doi:10.1186/1752-0509-6-145

Heerah, S., Molinari, R., Guerrier, S., and Marshall-Colon, A. (2021). Granger-causal testing for irregularly sampled time series with application to nitrogen signalling in arabidopsis. *Bioinformatics* 37, 2450–2460. doi:10.1093/bioinformatics/btab126

Huynh-Thu, V. A., Irrthum, A., Wehenkel, L., and Geurts, P. (2010). Inferring regulatory networks from expression data using tree-based methods. *PloS one* 5, e12776. doi:10.1371/journal.pone.0012776

Katari, M. S., Tyagi, S., and Shasha, D. E. (2021) *Statistics is easy: case studies on real scientific datasets*. Cham: Springer.

Krawczyk, A. O., de Jong, A., Eijlander, R. T., Berendsen, E. M., Holsappel, S., Wells-Bennik, M. H., et al. (2015). Next-generation whole-genome sequencing of eight strains of bacillus cereus, isolated from food. *Genome Announc.* 3, 014800–e11128. doi:10.1128/genomeA.01480-15

Li, Y., Hartemink, A. J., and MacAlpine, D. M. (2021). Cell-cycle–dependent chromatin dynamics at replication origins. *Genes* 12, 1998. doi:10.3390/genes12121998

Liu, Z.-P., Wu, C., Miao, H., and Wu, H. (2015). Regnetwork: an integrated database of transcriptional and post-transcriptional regulatory networks in human and mouse. *Database* 2015, bav095. doi:10.1093/database/bav095

Marbach, D., Costello, J. C., Küffner, R., Vega, N. M., Prill, R. J., Camacho, D. M., et al. (2012). Wisdom of crowds for robust gene network inference. *Nat. methods* 9, 796–804. doi:10.1038/nmeth.2016

Matsumoto, H., Kiryu, H., Furusawa, C., Ko, M. S., Ko, S. B., Gouda, N., et al. (2017). Scode: an efficient regulatory network inference algorithm from single-cell rna-seq during differentiation. *Bioinformatics* 33, 2314–2321. doi:10.1093/bioinformatics/btx194

Mitra, S., Zhong, J., Tran, T. Q., MacAlpine, D. M., and Hartemink, A. J. (2021). Robocop: jointly computing chromatin occupancy profiles for numerous factors from chromatin accessibility data. *Nucleic Acids Res.* 49, 7925–7938. doi:10.1093/nar/gkab553

Moerman, T., Aibar Santos, S., Bravo González-Blas, C., Simm, J., Moreau, Y., Aerts, J., et al. (2019). Grnboost2 and arboreto: efficient and scalable inference of gene regulatory networks. *Bioinformatics* 35, 2159–2161. doi:10.1093/bioinformatics/bty916

Pedreira, T., Elfmann, C., and Stülke, J. (2022). The current state of subti wiki, the database for the model organism bacillus subtilis. *Nucleic Acids Res.* 50, D875–D882. doi:10.1093/nar/gkab943

Pisithkul, T., Schroeder, J. W., Trujillo, E. A., Yeesin, P., Stevenson, D. M., Chaiamarit, T., et al. (2019). Metabolic remodeling during biofilm development of bacillus subtilis. *MBio* 10, e00623-19–e01128. doi:10.1128/mBio.00623-19

Pratapa, A., Jalihal, A. P., Law, J. N., Bharadwaj, A., and Murali, T. (2020). Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data. *Nat. methods* 17, 147–154. doi:10.1038/s41592-019-0690-6

Sagi, O., and Rokach, L. (2018). Ensemble learning: a survey. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* 8, e1249. doi:10.1002/widm.1249

Skok Gibbs, C., Jackson, C. A., Saldi, G.-A., Tjärnberg, A., Shah, A., Watters, A., et al. (2022). High-performance single-cell gene regulatory network inference at scale: the inferelator 3.0. *Bioinformatics* 38, 2519–2528. doi:10.1093/bioinformatics/btac117

Teixeira, M. C., Viana, R., Palma, M., Oliveira, J., Galocha, M., Mota, M. N., et al. (2023). Yeastract+: a portal for the exploitation of global transcription regulation and metabolic model data in yeast biotechnology and pathogenesis. *Nucleic Acids Res.* 51, D785–D791. doi:10.1093/nar/gkac1041

Tran, T. Q., MacAlpine, H. K., Tripuraneni, V., Mitra, S., MacAlpine, D. M., and Hartemink, A. J. (2021). Linking the dynamics of chromatin occupancy and transcription with predictive models. *Genome Res.* 31, 1035–1046. doi:10.1101/gr.267237.120

Van de Sande, B., Flerin, C., Davie, K., De Waegeneer, M., Hulselmans, G., Aibar, S., et al. (2020). A scalable scenic workflow for single-cell gene regulatory network analysis. *Nat. Protoc.* 15, 2247–2276. doi:10.1038/s41596-020-0336-2

Varala, K., Marshall-Colón, A., Cirrone, J., Brooks, M. D., Pasquino, A. V., Léran, S., et al. (2018). Temporal transcriptional logic of dynamic regulatory networks underlying nitrogen signaling and use in plants. *Proc. Natl. Acad. Sci.* 115, 6494–6499. doi:10.1073/pnas.1721487115

Zhao, M., He, W., Tang, J., Zou, Q., and Guo, F. (2021). A comprehensive overview and critical evaluation of gene regulatory network inference technologies. *Briefings Bioinforma.* 22, bbab009. doi:10.1093/bib/bbab009

Zheng, R., Li, M., Chen, X., Wu, F.-X., Pan, Y., and Wang, J. (2019). Bixgboost: a scalable, flexible boosting-based method for reconstructing gene regulatory networks. *Bioinformatics* 35, 1893–1900. doi:10.1093/bioinformatics/bty908