A Variational Inference Approach to Single-Cell Gene Regulatory Network Inference using Probabilistic Matrix Factorization

Omar Mahmood[*]
Center for Data Science
New York University

Claudia Skok Gibbs[*]
Center for Data Science
New York University

Richard Bonneau
Genomics & Systems Biology,
New York University
and
Prescient Design, Genentech

Kyunghyun Cho
Center for Data Science,
New York University
and
Prescient Design, Genentech
kyunghyun.cho@nyu.edu

Abstract

Inferring gene regulatory networks (GRNs) from single-cell gene expression datasets is a challenging task. Existing methods are often designed heuristically for specific datasets and lack the flexibility to incorporate additional information or compare against other algorithms. Further, current GRN inference methods do not provide uncertainty estimates with respect to the interactions that they predict, making inferred networks challenging to interpret. To overcome these challenges, we introduce Probabilistic Matrix Factorization for Gene Regulatory Network inference (PMF-GRN). PMF-GRN uses single-cell gene expression data to learn latent factors representing transcription factor activity as well as regulatory relationships between transcription factors and their target genes. This approach incorporates available experimental evidence into prior distributions over latent factors and scales well to single-cell gene expression datasets. By utilizing variational inference, we facilitate hyperparameter search for principled model selection and direct comparison to other generative models. To assess the accuracy of our method, we evaluate PMF-GRN using the model organisms *Saccharomyces cerevisiae* and *Bacillus subtilis*, benchmarking against database-derived gold standard interactions. We discover that, on average, PMF-GRN infers GRNs more accurately than current state-of-the-art single-cell GRN inference methods. Moreover, our PMF-GRN approach offers well-calibrated uncertainty estimates, as it performs gene regulatory network (GRN) inference in a probabilistic setting. These estimates are valuable for validation purposes, particularly when validated interactions are limited or a gold standard is incomplete.

***Keywords*—** Probabilistic Matrix Factorization, Variational Inference, Gene Regulatory Network Inference, Single Cell, Gene Expression.

[*]Equal contribution

1 Background

An essential problem in systems biology is to extract information from genome wide sequencing data to unravel the mechanisms controlling cellular processes within heterogeneous populations (1). Gene regulatory networks (GRNs) that annotate regulatory relationships between transcription factors (TFs) and their target genes (2) have proven to be useful models for stratifying functional differences between cells (3; 4; 5; 6) that can arise during normal development (7), responses to environmental signals (8) and dysregulation in the context of disease (9; 10; 11).

GRNs cannot be directly measured with current sequencing technology. Instead, methods must be developed to piece together snapshots of transcriptional processes in order to reconstruct a cell's regulatory landscape (12). Initial approaches to GRN inference relied on Microarray technology (13; 14; 15), a hybridization-based method to measure the expression of thousands of genes simultaneously (16). This technology was biased as it was limited to only those genes that were annotated at the time, which in turn presented challenges for inferring the complete regulatory landscape (1). Subsequently, the high-throughput sequencing method RNA-seq provided a genome wide readout of transcriptional output, allowing for the detection of novel transcripts (17) and thus improving GRN inference potential. More recently, single-cell RNA-seq technology has enabled the characterization of gene expression profiles within heterogeneous populations (18), vastly increasing the potential for GRN inference algorithms (19; 20). In contrast to bulk RNA experiments (Microarray and RNA-seq) that average measurements of gene expression across heterogenous cell populations, GRNs inferred from single-cell data have the advantage of unmasking biological signal in distinct cells (21).

Several matrix factorization approaches have been proposed to overcome the limitations of reconstructing GRNs from Microarray data (22). These include use of statistical techniques such as Singular Value Decomposition and Principal Component Analysis (23), Bayesian Decomposition (24), and Non-negative Matrix Factorization (25; 26; 27). More recently, matrix factorization approaches have been applied to integrative analysis of DNA methylation and miRNA expression data (28), as well as single-cell RNA-seq and single-cell ATAC-seq data (29). However, to the best of our knowledge, these matrix factorization approaches have not yet been used to infer GRNs from single-cell gene expression data. Meanwhile, several regression-based methods have been proposed to learn GRNs from single-cell RNA-seq and single-cell ATAC-seq to capture regulatory relationships at single-cell resolution (30). So far, these integrative approaches to GRN inference have been successfully implemented using regularized regression (31), self-organizing maps (32), tree-based regression (33), and Bayesian Ridge regression (34).

Although regression-based methods for inferring GRNs from single-cell data are available, they still suffer from significant limitations (35). Firstly, these methods heavily rely on the input data used to learn the GRN, causing issues when new data becomes available or new assumptions are required in the model. This can result in inaccurate predictions if the new data or assumptions are not well integrated into the existing model, leading to the need for a complete re-design of the algorithm, which can be costly and time-consuming. Additionally, these methods typically focus on inferring a single GRN that explains the available data, without performing hyperparameter search to determine the optimal model. This can lead to heuristic model selection, with no justification for the approach taken or evidence that the best possible model has been selected. Conversely, hyperparameter search ensures the accuracy of the GRN inference algorithm by finding the optimal model that fits the data well while avoiding overfitting. Regression-based GRN inference algorithms that do not perform hyperparameter search may miss important data features or overemphasize irrelevant ones, leading to inaccurate or incomplete models. Moreover, these methods do not provide an indication of their uncertainty about the predictions that they make. Finally, several regression-based GRN inference algorithms struggle to scale optimally to the size of typical single-cell datasets, limiting inference to small subsets of data or requiring enormous amounts of computational time."

In this study, we introduce PMF-GRN, a novel approach that uses probabilistic matrix factorization (36) to infer gene regulatory networks from single-cell gene expression and chromatin accessibility information. This approach extends previous methods that applied matrix factorization for GRN inference with Microarray data, to address the current limitations in regression-based single-cell GRN inference. We implement our approach in a probabilistic setting with variational inference, which provides a flexible framework to incorporate new assumptions or biological data as required, without changing the way the GRN is inferred. We also use a principled hyperparameter selection process with the Evidence Lower Bound (ELBO) objective function, which optimizes the parameters of our probabilistic model for automatic model selection. In this way, we replace heuristic model selection by comparing a variety of generative models and hyperparameter configurations before selecting the optimal parameters with which to infer a final GRN. Our probabilistic approach provides uncertainty estimates for each predicted regulatory interaction, serving as a proxy for the model confidence in each predicted interaction. Uncertainty estimates can be useful in the situation where there are limited validated interactions or a gold standard is incomplete. By using stochastic gradient descent (SGD), we perform GRN inference on a GPU, allowing us to easily scale to a large number of observations in a typical single-cell gene expression dataset. Unlike many existing methods, PMF-GRN is not limited by pre-defined organism restrictions, making it widely applicable for GRN inference.

To demonstrate the novelty and advantages of PMF-GRN, we apply our method to two single-cell gene expression datasets for the model organism *Sacchromyces cerevisiae*. We evaluate our model's performance in a normal inference setting, as well as with cross-validation and noisy data. To assess the accuracy of predicted regulatory interactions, we evaluate all regulatory predictions using Area Under the Precision Recall Curve (AUPRC) against database derived gold standards. Our findings show that the uncertainty estimates are well-calibrated for inferred TF-target gene interactions, as the accuracy of predictions increases when the associated uncertainty decreases. Here, in comparison to three state-of-the-art regression-based methods for inferring single-cell GRNs, namely the Inferelator (31), Scenic (33), and Cell Oracle (34), our method demonstrates an overall improved performance in recovering the true underlying GRN. We also include GRNs inferred using two microarray datasets for *Bacillus subtilis* by converting expression values to integers to simulate a single-cell-like experiment, demonstrating our method's performance on a second dataset.

2 Results

2.1 The PMF-GRN Model

The goal of our probabilistic matrix factorization approach is to decompose observed gene expression into latent factors, representing TF activity (TFA) and regulatory interactions between TFs and their target genes. These latent factors, which represent the underlying GRN, cannot be measured experimentally, unlike gene expression. We model an observed gene expression matrix $W \in \mathbb{R}^{N \times M}$ using a TFA matrix $U \in \mathbb{R}_{>0}^{N \times K}$, a TF-target gene interaction matrix $V \in \mathbb{R}^{M \times K}$, observation noise $\sigma_{obs} \in (0, \infty)$ and sequencing depth $d \in (0, 1)^N$, where N is the number of cells, M is the number of genes and K is the number of TFs. We rewrite V as the product of a matrix $A \in (0, 1)^{M \times K}$, representing the degree of existence of an interaction, and a matrix $B \in \mathbb{R}^{M \times K}$ representing the interaction strength and its direction:

$$V = A \odot B,$$

where $\odot$ denotes element-wise multiplication. An overview of the graphical model is shown in Figure 1A.

These latent variables are mutually independent *a priori*, i.e., $p(U, A, B, \sigma_{obs}, d) = p(U)p(A)p(B)p(\sigma_{obs})p(d)$. For the matrix A, prior hyperparameters represent an initial guess of the interaction between each TF and target gene which need to be provided by a user. These can be derived from genomic databases or obtained by analyzing other data types, such as the measurement of chromosomal accessibility, TF motif databases, and direct measurement of TF-binding along the chromosome, as shown in Figure 1B (see Methods section for details).

The observations W result from a matrix product $UV^\top$. We assume noisy observations by defining a distribution over the observations with the level of noise σ_{obs}, i.e., $p(W|U, V = A \odot B, \sigma_{obs}, d)$.

Given this generative model, we perform posterior inference over all the unobserved latent variables; U, A, B, d and σ_{obs}, and use the posterior over A to investigate TF-gene interactions. Exact posterior inference with an arbitrary choice of prior and observation probability distributions is, however, intractable. We address this issue by using variational inference (37; 38), where we approximate the true posterior distributions with tractable, approximate (variational) posterior distributions.

We minimize the KL-divergence $D_{KL}(q\|p)$ between the two distributions with respect to the parameters of the variational distribution q, where p is the true posterior distribution. This allows us to find an approximate posterior distribution q that closely resembles p. This is equivalent to maximizing the evidence lower bound (ELBO) i.e. a lower bound to the marginal log likelihood of the observations W:

$$\begin{aligned}\log p(W) \geq \mathbb{E}_{U,A,B,\sigma_{obs},d \sim q(U,A,B,\sigma_{obs},d)}[&\log p(W|U, V = A \odot B, \sigma_{obs}, d) \\ &+ \log p(U, A, B, \sigma_{obs}, d) \\ &- \log q(U, A, B, \sigma_{obs}, d)]\end{aligned}$$

The mean and variance of the approximate posterior over each entry of A from maximizing the ELBO are then used as the degree of existence of an interaction between a TF and a target gene and its uncertainty, respectively.

It is important to note that matrix factorization based GRN inference is only identifiable up to a latent factor (column) permutation. In the absence of prior information, the probability that the user assigns TF names to the columns of U and V in the same order as the order in which the inference algorithm implicitly assigns TFs to these columns is $\frac{1}{K!}$, which is essentially 0 for any reasonable value of K. Incorporating prior-knowledge of TF-target gene interactions into the prior distribution over A is therefore essential to give the inference algorithm information about which column corresponds to which TF.

With this identifiability issue in mind, we design an inference procedure that can be used on any dataset, described in Figure 1C. The first step is to randomly hold out prior information for some percentage of the genes in $p(A)$

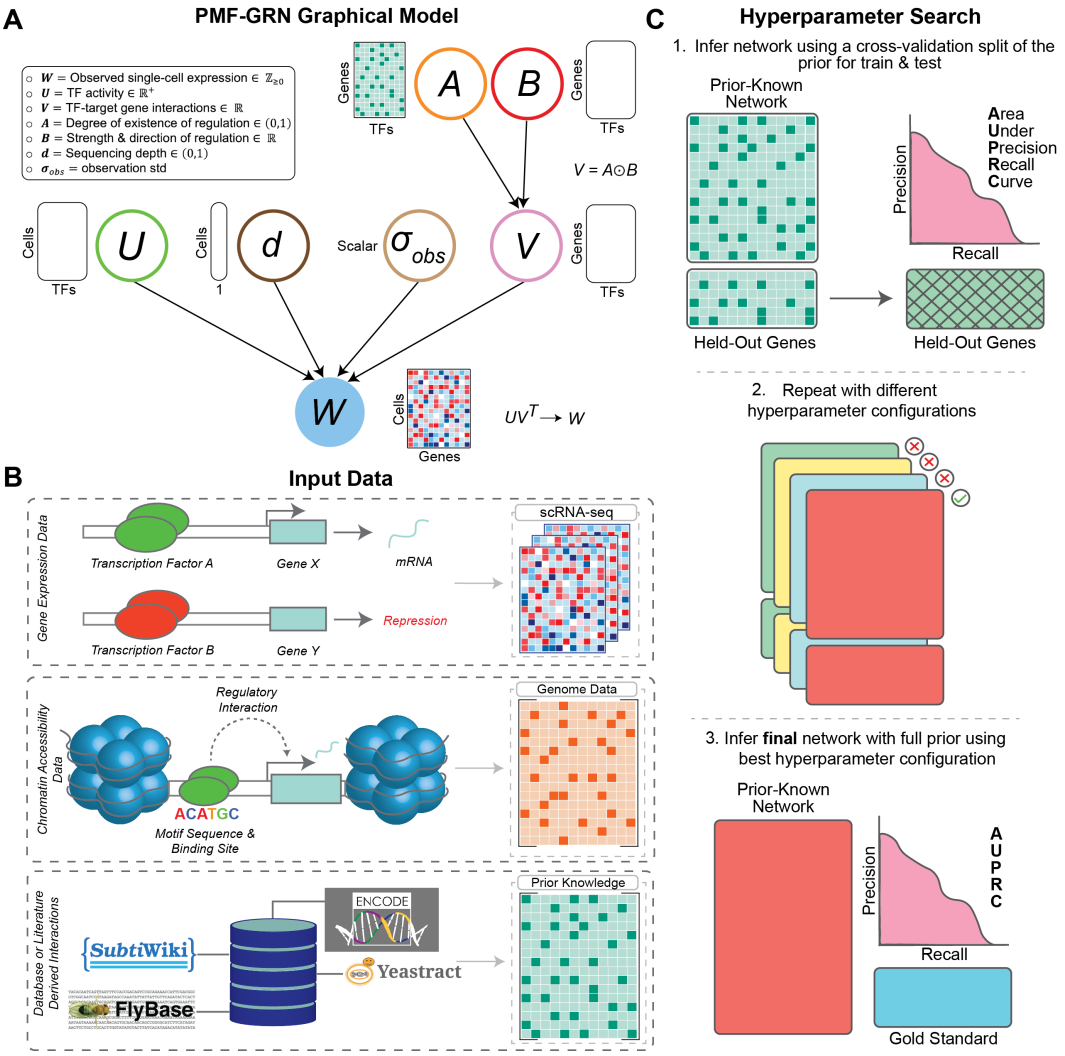


Figure 1: (**A**) PMF-GRN graphical model overview. Input single-cell gene expression W is decomposed into latent factors U and V, representing TF activity and TF-gene interactions respectively. V is further decomposed into A and B, representing the degree of existence of interaction, and the strength and direction of an interaction, respectively. Information obtained from chromatin accessibility data or genomics databases is incorporated into the prior distribution for A. Additional latent variables are included to model observation noise σ_{obs} and sequencing depth d, in order to better model our observed single-cell gene expression input data. (**B**) Input experimental data for PMF-GRN includes single-cell RNA-seq gene expression data. ATAC-seq is used to determine chromatin accessibility through peak calling. Motif enrichment within these accessible regions can be used to create a prior-known network to better inform the prior distribution. When experimental information is unavailable, databases can be used instead to construct a known-prior network. (**C**) Hyperparameter selection process is performed for model selection. The provided prior-known network is split into a train and validation dataset. 80% of the prior-known information is used to infer a GRN, while the remaining 20% is used for validation by computing AUPRC. This process is repeated multiple times, using different hyperparameter configurations in order to determine the optimal hyperparameters for the GRN inference task at hand. Finally, using the optimal hyperparameters, as determined by the highest achieved AUPRC, a final network is inferred using the full prior and evaluated using an independent gold standard.

(we choose 20%) by leaving the rows corresponding to these genes in A but setting the prior logistic normal means for all entries in these rows to be the same low number.

The second step is to carry out a hyperparameter search using this modified prior-knowledge matrix. The early stopping and model selection criteria are both the 'validation' AUPRC of the posterior point estimates of A corresponding to the held out genes against the entries for these genes in the full prior hyperparameter matrix. This step is motivated by the idea that inference using the selected hyperparameter configuration should yield a GRN whose columns correspond to the TF names that the user has assigned to these columns.

The third step is to choose the hyperparameter configuration corresponding to the highest validation AUPRC and perform inference using this configuration with the full prior. An importance weighted estimate of the marginal log likelihood is used as the early stopping criterion for this step. The resulting approximate posterior provides the final posterior estimate of A.

2.2 Advantages of PMF-GRN

Existing methods almost always couple the description of the data generating process with the inference procedure used to obtain the final estimated GRN (31; 34; 33). Designing a new model thus requires designing a new inference procedure specifically for that model, which makes it difficult to compare results across different models due to the discrepancies in their associated inference algorithms. Furthermore, this *ad hoc* nature of model building and inference algorithm design often leads to the lack of a coherent objective function that can be used for proper hyperparameter search as well as model selection and comparison, as evident in (31). Heuristic model selection in available GRN inference methods presents the challenge of determining and selecting the optimal model in a given setting.

The proposed PMF-GRN framework decouples the generative model from the inference procedure. Instead of requiring a new inference procedure for each generative model, it enables a single inference procedure through (stochastic) gradient descent with the ELBO objective function above, across a diverse set of generative models. Inference can easily be performed in the same way for each model. Through this framework, it is possible to define the prior and likelihood distributions as desired with the following mild restrictions: we must be able to evaluate the joint distribution of the observations and the latent variables, the variational distribution and the gradient of the log of the variational distribution.

The use of stochastic gradient descent in variational inference comes with a significant computational advantage. As each step of inference can be done with a small subset of observations, we can run GRN inference on a very large dataset without any constraint on the number of observations. This procedure is further sped up by using modern hardware, such as GPUs.

Under this probabilistic framework, we carry out model selection, such as choosing distributions and their corresponding hyperparameters, in a principled and unified way. Hyperparameters can be tuned with regard to a predefined objective, such as the marginal likelihood of the data or the posterior predictive probability of held out parts of the observations. We can further compare and choose the best generative model using the same procedure.

This framework allows us to encode any prior knowledge via the prior distributions of latent variables. For instance, we incorporate prior domain knowledge about TF-gene interactions as hyperparameters that govern the prior distribution over the matrix A. If prior knowledge about TFA is available, this can be similarly incorporated into the model via the hyperparameters of the prior distribution over U.

Because our approach is probabilistic by construction, inference also estimates uncertainty without any separate external mechanism. These uncertainty estimates can be used to assess the reliability of the predictions, i.e., more trust can be placed in interactions that are associated with less uncertainty. We verify this correlation between the degree of uncertainty and the accuracy of interactions in the experiments.

Overall, the proposed approach of probabilistic matrix factorization for GRN inference is scalable, generalizable and aware of uncertainty, which makes its use much more advantageous compared to most existing methods.

2.3 PMF-GRN Recovers True Interactions in Simple Eukaryotes

To demonstrate PMF-GRNs ability to infer informative and robust GRNs, we use two single-cell RNA-seq datasets from the model organism *Sacchromyces cerevisiae*. *S.cerevisiae* is a relatively simple and well studied eukaryote with an available and reliable gold standard, which allows us to test and evaluate our models performance.

We perform three experiments using two independently collected single-cell RNA-seq *S. cerevisiae* datasets (8; 39) to test PMF-GRN and compare our performance against three state-of-the-art GRN inference methods, the Inferelator (AMuSR, BBSR, StARS) (31), Scenic (33), and CellOracle (34). In the first experiment, we infer a GRN for each of the two single-cell datasets and average the posterior means of A to simulate a "multi-task"

GRN inference approach for building the final combined network. Using AUPRC, we show that PMF-GRN outperforms AMuSR, StARS, and Scenic, while performing competitively with BBSR and CellOracle (Figure 2A). To provide a baseline for each method in the scenario where data cannot be cleanly separated into tasks, we combine the two expression datasets into one observation before inferring a GRN. This baseline demonstrates a large performance decrease for BBSR, indicating that the method may only be useful when gene expression is organized into tasks. This could present challenges when attempting to infer GRNs in more complicated organisms where cell-types or conditions are less easily defined. Here, we show the effectiveness of PMF-GRN in recovering the true underlying GRN for both scenarios, as performance remains relatively stable whether a network was inferred using the individual or combined data. We also provide an example as to how we can use PMF-GRN on a single observation or multiple observation matrices to infer a consensus GRN by simple averaging.

In the second experiment, we implement a 5 fold cross-validation approach to establish a baseline for each model. Cross-validation is an essential technique for evaluating the performance of machine learning models like PMF-GRN as it allows us to test our method's ability to generalize to new data. Cross-validation further allows us to simulate the process of training and testing PMF-GRN on multiple subsets of the available data, providing a more robust and reliable estimate of model accuracy. In the context of GRN inference, cross-validation is particularly important because it helps us assess the performance of PMF-GRN in predicting TF-target gene interactions based on limited data, which is often the case in experimental settings.

We first combine the two *S. cerevisiae* single-cell RNA-seq datasets into one observation matrix for simplicity. To perform cross-validation, the gold standard is divided into an $80\% - 20\%$ split, where a network is inferred using 80% of the gold standard as "prior-known information", and evaluated using the remaining 20%. We repeat this cross-validation process five times using different random splits of the gold standard to obtain meaningful results. We observe that PMF-GRN outperforms Scenic and CellOracle, while achieving competitive performance to BBSR and StARS (Figure 2B). We note that for this experiment, we are unable to implement the AMuSR algorithm as it is a multi-task inference approach that requires more than one task (dataset).

Finally, in the third experiment, we demonstrate the robustness of each GRN inference method against noisy prior information. To do so, we infer GRNs where increasing amounts of noise have been added to the input prior-known information. Here, we show that as noise increases, PMF-GRN's AUPRC decreases similarly to CellOracle, while on average, performing better than BBSR, StARS and CellOracle, demonstrating that it is one of the most robust approaches to inferring accurate GRNs from noisy priors (Figure 2C).

From the results of our experiments on the *S. cerevisiae* data, we have the following observations. The first main observation is that on average the proposed PMF-GRN performs better than the Inferelator in recovering the true GRN, regardless of whether we pick the mean or median Inferelator algorithm in terms of AUPRC. Specifically, we see that PMF-GRN performs markedly better than two Inferelator algorithms (AMuSR and StARS), and similarly to the remaining algorithm (BBSR). However, when the expression data is not separated into tasks, PMF-GRN outperforms BBSR. In comparison to CellOracle, we observe that PMF-GRN infers competitive GRNs during normal inference. However, PMF-GRN greatly outperforms CellOracle when performing cross-validation. Finally, we observe that PMF-GRN consistently outperforms Scenic in all experiments considered.

The second main observation is that our approach eliminates the high variance associated with choosing between different inference algorithms. Implementing the Inferelator on the *S. cerevisiae* datasets in a normal setting yields AUPRCs approximately in the range 0.2 to 0.4, without any a priori information on which of these algorithms to use. The resulting inferred GRN could be arbitrarily accurate or inaccurate depending on which algorithm is chosen. In contrast, our method is reliable as it provides one set of results, chosen using a principled objective function, performing competitively with the best performing Inferelator algorithm (BBSR) and CellOracle.

Finally, in order to highlight the identifiability issue and ensure that the prior-known information provided is useful, we demonstrate the performance of PMF-GRN where prior information is not used (e.g. all prior logistic normal means of A are set to the same low number). We use the same process for all other GRN inference algorithms by providing an empty prior. Additionally, we demonstrate PMF-GRN's performance when we randomly shuffle the prior-known TF-target gene interaction hyperparameters before using them to build the prior distribution for A. We repeat this process for all other GRN inference algorithms by providing them with prior-known information in which the gene names have been shuffled randomly. As anticipated, the resulting AUPRC scores are close to 0, implying that our approach, as well as the Inferelator and CellOracle are capable of taking into account such prior information well and that the prior information we provided is useful and reliable (see Methods section for details). The results for GRNs inferred without prior-known information are demonstrated by the black dots, while the gray dots demonstrate GRNs inferred with shuffled prior-known information, shown in Figure 2A.

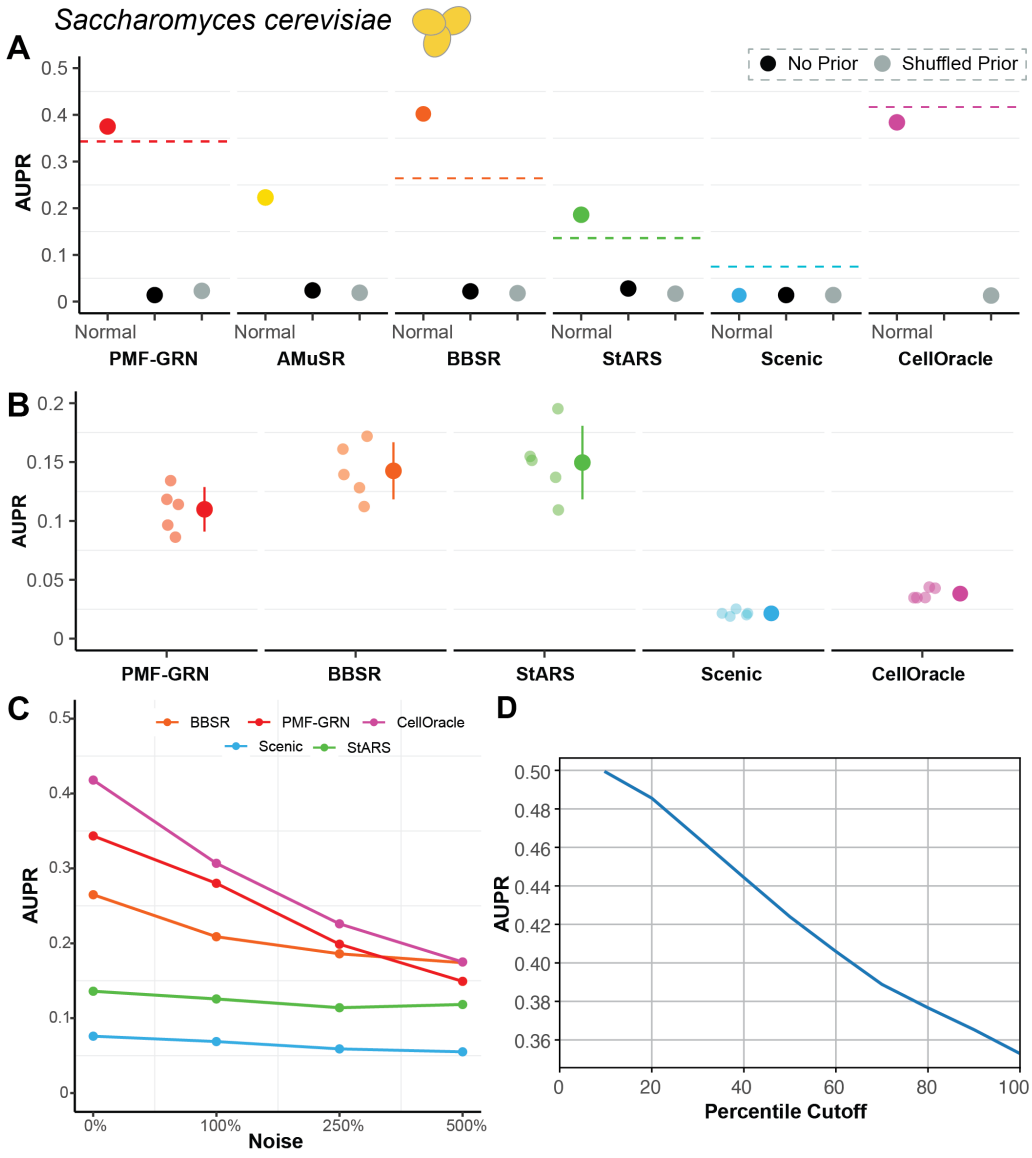


Figure 2: GRN inference in *Saacharomyces cerevisiae*. (**A**) Consensus network AUPR using gold standard network. Performance of PMF-GRN (red) is compared to three Inferelator algorithms, AMuSR (yellow), BBSR (orange), StARS (green), as well as to Scenic (blue), and CellOracle (purple). A baseline for each method (dashed line), demonstrates performance if the expression data is combined into one task. Two negative controls, no prior information (black) and shuffled prior information (gray), are inferred to ensure reliable results. (**B**) 5 fold cross-validation establishes a baseline for each model. Low-opacity dots represent each of the five cross-validation experiments. The mean AUPR $\pm$ standard deviation for each GRN inference method is depicted by colored dot and line. (**C**) GRNs inferred with increasing amounts of noise added to the prior. (**D**) Calibration results on the *S.cerevisiae* (GSE144820 (8) only) dataset. Posterior means are cumulatively placed in bins based on their posterior variances. The x-coordinate x of each point in the plot represents all posterior means that correspond to the bottom $x\%$ of posterior variances. The y-coordinate is the 'overlap' AUPRC (see Methods section for details) calculated on these posterior means against the gold standard.

2.4 PMF-GRN Recovers True Interactions in Prokaryotes as Evaluated by Cross-Validation

To demonstrate GRN inference on a second dataset, we carry out experiments using two microarray datasets for the prokaryote *Bacillus Subtilis* (B1 - GSE27219 (40) and B2 - GSE67023 (41)). Although PMF-GRN is not primarily designed to learn GRNs from microarray data, we show that it is still possible to learn informative GRNs with this data. For our *B. subtilis* experiments, we have access to prior-knowledge derived from the subtiwiki database (42; 43; 44). Here, we implement a 5 fold cross-validation approach by using five random splits of the subtiwiki database-derived information, where 80% is used as prior knowledge and 20% is used as the gold standard for evaluation.

The two *B. subtilis* datasets were previously normalized after data collection as part of standard microarray processing. However, each dataset was normalized using different approaches (described in 4). For B1, the expression data underwent no further normalization and was simply converted to integers to simulate single-cell-like data. For B2, the expression data was re-scaled and then converted to integers, in order to contain only positive integers resembling single-cell-like data. The results from our experiments are shown in Figure 3, and the numbers used to create this figure are given in Supplementary Table 4 and 5. Using five repeats of cross-validation, we show the performance of GRNs inferred for the two *B. subtilis* datasets (B1 and B2). We remark that the difference in performance between B1 and B2 is likely a result of the chosen microarray processing normalization. To further support this claim, we demonstrate GRN performance after re-scaling the data with min-max scaling (Supplemental Figure S1).

'No Prior' and 'Shuffled' results are also shown in Figure 3 by black and gray dots respectively. Here, we are able to demonstrate that for B1 and B2, each GRN yields a better performance as compared to negative controls.

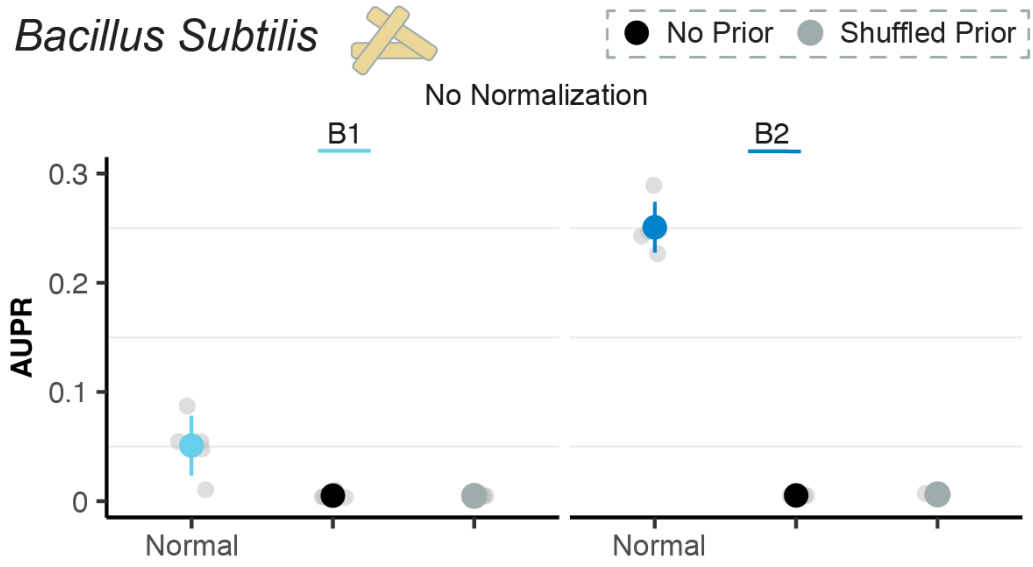


Figure 3: Results for GRNs learned in *B. subtilis* datasets B1 and B2 without data normalization. Light gray dots represent the results for each of the 5 cross-validation experiments. Colored dots represent the mean of the cross-validation experiments $\pm$ standard deviation. Negative controls are demonstrated by black dots for "No Prior" and grey dots for "Shuffled Prior".

2.5 PMF-GRN Provides Well-Calibrated Uncertainty Estimates

Through our inference procedure, we obtain a posterior variance for each element of A, in addition to the posterior mean. We interpret each variance as a proxy for the uncertainty associated with the corresponding posterior point estimate of the relationship between a TF and a gene. Due to our use of variational inference as the inference procedure, our uncertainty estimates are likely to be underestimates. However, these uncertainty estimates still provide useful information as to the confidence the model places in its point estimate of each interaction. We expect posterior estimates associated with lower variances (uncertainties) to be more reliable than those with higher variances.

In order to determine whether this holds for our posterior estimates, we cumulatively bin the posterior means of A according to their variances, from low to high. We then calculate the AUPRC for each bin as shown for the GSE125162 (8) *S.cerevisiae* dataset in Figure 2 D. It is evident from the figures that the AUPRC decreases as the posterior variance increases. Stated differently, inferred interactions associated with lower uncertainty are more likely to be accurate than those associated with higher uncertainty. This is in line with our expectations. The more certain the model is about the degree of existence of a regulatory interaction, the more accurate it is likely to be, showing that our model is well-calibrated.

3 Conclusion

In this paper we present a framework for probabilistic matrix factorization, optimized using automatic variational inference, for inferring GRNs from single cell gene expression data. In contrast with previous methods, our framework decouples the model that defines the data generation process from the inference procedure. Concretely, this means that we can modify the latent variables that constitute the model, along with their distributions, without altering the inference procedure. This flexibility will allow for different sequencing data and modeling assumptions to be readily incorporated into the model. Building new models no longer requires defining a new inference procedure, which has previously been the case.

Additionally, PMF-GRN provides a principled way to carry out model selection and hyperparameter configuration by using the same objective function and inference procedure across all models. This feature differs from previous GRN methods, where it is often unclear which algorithm or hyperparameters to use for a given dataset. In the PMF-GRN framework, we carry out hyperparameter searches across generative models and choose the configuration that corresponds to the optimal value of the objective function. This greatly reduces the need for heuristic model selection.

In order to demonstrate successful GRN inference, we infer a consensus GRN for *S.cerevisiae* using our principled model selection method, and compare our results to GRNs inferred by the Inferelator, Scenic and CellOracle with respect to a reliable gold standard. Whereas the Inferelator yields a set of highly varying results across the variants, our approach results in a single inferred GRN. This GRN yields an AUPRC that is higher than Scenic, as well as than the mean and median AUPRC achieved by the Inferelator's respective algorithms (AMuSR and StARS), and is comparable to the AUPRC achieved by the best performing Inferelator algorithm (BBSR), in addition to CellOracle. However, when the expression data is not separated into tasks, we demonstrate that BBSR can no longer recover a competitive network. Our model hence yields a reliable high-performing set of results without any need for heuristic model selection.

We further evaluate PMF-GRN by performing cross-validation and find that PMF-GRN, BBSR, and StARS yield high performance, indicating that these methods are able to generalize well to new data and do not overfit to training data. In contrast, Scenic and CellOracle do not perform well during cross-validation, indicating that these methods may not be generalizable.

Finally, because prior-known networks are inherently noisy due to limited validated regulatory interactions, we design an experiment to test each algorithm's robustness against increasing amounts of noise in the input prior-known information. This experiment identifies PMF-GRN and CellOracle as the methods which are overall most robust against noisy priors, indicating that inferred GRNs will remain reliable regardless of noisy interactions introduced into the prior during inference.

Using two microarray *B. subtilis* datasets, we further demonstrate that PMF-GRN is capable of learning informative GRNs. To include database information into both the prior knowledge and evaluation, we use an approach motivated by cross-validation. Here, we find that scaling allow us to place these datasets on the same scale, allowing them to be more comparable during inference. Although PMF-GRN is not primarily designed for microarray data, we show that is still possible to learn informative networks by simply converting the expression to integers to represent single-cell-like counts.

In order to determine the effect of incorporating our prior domain knowledge into the model, we compare results obtained using shuffled and unshuffled hyperparameters for the matrix A. We observe that for both *S. cerevisiae* and *B. subtilis*, not using prior information or shuffling the prior information results in very low AUPRCs, whereas using the prior information as intended results in significantly better AUPRCs. This result holds for PMF-GRN as well as for CellOracle and all Inferelator algorithms. However, for Scenic, we show that it is challenging to obtain a GRN that performs better than these negative controls. This shows that prior information is essential for addressing the latent factor identifiability issue and obtaining interpretable results from matrix factorization, as well as regression based approaches.

In contrast to previous methods, our model provides well-defined uncertainty estimation in addition to point estimates of GRNs. We evaluate these uncertainty estimates as provided by our model, by computing the AUPRC for inferred TF-target gene interactions corresponding to different levels of posterior uncertainty. We find that the AUPRC increases as the posterior variance decreases, demonstrating that when our model is more certain about its estimates, it produces better rankings of TF-target gene interactions compared to when it is uncertain. This

indicates that our model is well-calibrated. For downstream experimental validation, biologists could therefore place more trust in model estimates that have a lower posterior variance. We also note that the computational cost of our model scales linearly with the number of cells in the dataset. This enables application of our method to single-cell RNA-seq datasets of any size.

We envision many possible directions for future work to design a better algorithm for inferring GRNs under our framework. This framework could be extended to explicitly model multiple expression matrices and their batch effects. We could probabilistically model prior information for A obtained from ATAC-seq and TF motif databases, and include this as part of the probabilistic model over which we carry out inference. Evaluating the posterior estimates of the direction of transcriptional regulation, provided by the matrix B, could provide a useful benchmark for the computational estimation of TF activation and repression. Research could also be carried out on improved self-supervised objectives for hyperparameter selection.

Future work could also focus on how to use results from our framework to guide experimental wet-lab work. For example, the uncertainty quantification provided by our model could open up new research directions in active learning for GRN inference. Highly ranked, uncertain interactions could be experimentally tested and the results fed back into the prior hyperparameter matrix for A. Inference with this updated matrix would ideally yield a better posterior GRN estimate. Posterior estimates of TFA provided by our model could be useful to wet lab scientists, as this quantity incorporates information on post-transcriptional modifications.

Most importantly, the study of GRN inference is far from complete. So far, this has required new computational models and assumptions in order to keep up with relevant sequencing technologies. It is thus essential to develop a model that can be easily adapted to new biological datasets as they become available, without having to completely re-build each model. We have therefore proposed PMF-GRN as a modular, principled, probabilistic approach that can be easily adapted to both new and different biological data without having to design a new GRN inference method.

4 Methods

4.1 Model Details

We index cells, genes and TFs using $n \in \{1, \cdots, N\}$, $m \in \{1, \cdots, M\}$ and $k \in \{1, \cdots, K\}$, respectively. We treat each cell's expression profile W_n as a random variable, with local latent variables U_n and d_n, and global latent variables (that are shared among all cells) σ_{obs} and $V = A \odot B$. We use the following likelihood for each of our observations:

$$p(W_n|U, V, \sigma_{obs}, d) = \mathcal{N}(d_n * U_n V^\top, \sigma_{obs}^2).$$

We assume that U, V, σ_{obs} and d are independent i.e. $p(U, V, \sigma_{obs}, d) = p(U)p(V)p(\sigma_{obs})p(d)$. In addition to our iid assumption over the rows of U and d, We also assume that the entries of U_n are mutually independent, and that all entries of A and B are mutually independent. We choose a lognormal distribution for our prior over U and a logistic Normal distribution for our prior over d:

$$\begin{aligned} p(\log(U_{nk})) &= \mathcal{N}(\mu_u, \sigma_u^2), \\ p(\text{logit}(d_n)) &= \mathcal{N}(0, 9) \end{aligned}$$

where $\mu_u \in \mathbb{R}$ and $\sigma_u \in \mathbb{R}^+$.

We use a logistic Normal distribution for our prior over A, a Normal distribution for our prior over B and a logistic Normal distribution for our prior over σ_{obs}:

$$\begin{aligned} p(\text{logit}(A_{mk})) &= \mathcal{N}(\text{logit}(\text{clip}(\bar{A}_{mk}, a_{\max}, a_{\min})), \sigma_a^2), \\ p(B_{mk}) &= \mathcal{N}(0, \sigma_b^2). \\ p(\log(\sigma_{obs})) &= \mathcal{N}(0, 1), \end{aligned}$$

where $\bar{A}_{mk} \in \{0, 1\}$, $a_{\max} \in (0, 1)$, $a_{\min} \in (0, 1)$, $\sigma_a \in \mathbb{R}_{>0}$, $\text{clip}(\bar{A}_{mk}, a_{\max}, a_{\min}) = \max(\min(\bar{A}_{mk}, a_{\max}), a_{\min})$ and $\sigma_b \in \mathbb{R}_{>0}$. $\bar{A}_{mk}$ is given by a pipeline that is used by other methods such as the Inferelator. The pipeline leverages ATAC-seq and TF binding motif data to provide binary initial guesses of gene-TF interactions. $a_{\max}$ and $a_{\min}$ are hyperparameters that determine how we clip these binary values before transforming them to the logit space.

For our approximate posterior distribution, we enforce independence as follows:

$$q(U, A, B, \sigma_{obs}, d) = q(U)q(A)q(B)q(\sigma_{obs})q(d).$$

We impose the same independence assumptions on each approximate posterior as we do for its corresponding prior. Specifically, we use the following distributions:

$$
\begin{aligned}
q(\log(U_{nk})) &= \mathcal{N}(\tilde{U}_{nk}, \tilde{\sigma}^2_{U_{nk}}) \\
q(\text{logit}(d_n)) &= \mathcal{N}(\tilde{d}_n, \tilde{\sigma}^2_{d_n}) \\
q(\text{logit}(A_{mk})) &= \mathcal{N}(\tilde{A}_{mk}, \tilde{\sigma}^2_{A_{mk}}) \\
q(B_{mk}) &= \mathcal{N}(\tilde{B}_{mk}, \tilde{\sigma}^2_{B_{mk}}) \\
q(\log(\sigma_{obs})) &= \mathcal{N}(\tilde{o}, \tilde{\sigma}^2_o),
\end{aligned}
$$

where the parameters on the right hand sides of the equations are called variational parameters; $\tilde{U}_{nk}$, $\tilde{d}_n$, $\tilde{A}_{mk}$, $\tilde{B}_{mk}$, $\tilde{o} \in \mathbb{R}$ and $\tilde{\sigma}_{U_{nk}}$, $\tilde{\sigma}_{d_n}$, $\tilde{\sigma}_{A_{mk}}$, $\tilde{\sigma}_{B_{mk}}$, $\tilde{\sigma}_o \in \mathbb{R}^+$. To avoid numerical issues during optimization, we place constraints on several of these variational parameters.

4.2 Inference

We perform inference on our model by optimizing the variational parameters to maximize the ELBo. In doing so, we minimise the KL-divergence between the true posterior and the variational posterior. In practice, to help with addressing the latent factor identifiability issue, we use a modified version of the ELBo where the prior and posterior terms are weighted by a constant $\beta \geq 1$ (45):

$$
\begin{aligned}
\mathbb{E}_{U,A,B,\sigma_{obs},d\sim q(U,A,B,\sigma_{obs},d)}[\,&\log p(W|U, V = A \odot B, \sigma_{obs}, d) \\
&+ \beta(\log p(U, A, B, \sigma_{obs}, d) - \log q(U, A, B, \sigma_{obs}, d))]
\end{aligned}
$$

Inference is carried out using the Adam optimizer with learning rate 0.1 and beta values of 0.9 and 0.99. We clip gradient norms at a value of 0.0001. We set $a_{\min} = 0.005$, $a_{\max} = 0.995$, $\sigma_b^2 = 1$ and $\mu_u = 0$. We vary σ_a and σ_u as hyperparameters that control the strengths of the priors over A and U, respectively. We also vary β as a hyperparameter.

We choose a hyperparameter configuration using validation AUPRC as the objective function as well as the early stopping metric. We hold out hyperparameters for $p(A)$ for a fraction of the genes. We do this by setting $\bar{A}_{mk} = 0$ for m corresponding to these genes for all k. During inference we regularly obtain posterior point estimates for these entries and measure the AUPRC against the original values of these entries as given in the full prior. This quantity is known as the validation AUPRC.

Once we have picked the hyperparameter configuration corresponding to the best validation AUPRC, we perform inference with this model using the full prior without holding out any information. We use an importance weighted estimate of the marginal log likelihood as our early stopping criterion:

$$
\log p(W) = \log\left(\mathbb{E}_{U,A,B,\sigma_{obs},d\sim q(U,A,B,\sigma_{obs},d)}\left[\frac{p(W|U, A, B, \sigma_{obs}, d)p(U, A, B, \sigma_{obs}, d)}{q(U, A, B, \sigma_{obs}, d)}\right]\right),
$$

where the expectation is computed using simple Monte Carlo and the log-$\sum$-exp trick is used to avoid numerical issues.

4.3 Computing Summary Statistics for the Posterior

After training the model, we use $\tilde{A}$ and $\tilde{\sigma}_A$, the variational parameters of $q(A)$, to obtain a mean and a variance for each entry of A. Since $q(A)$ is logistic normal, it admits no closed form solution for the mean and variance. We therefore use Simple Monte Carlo i.e. we sample each entry of A several times from its posterior distribution and then compute the sample mean and sample variance from these samples. We use each mean as a posterior point estimate of the probability of interaction between a TF and a gene, and its associated variance as a proxy for the uncertainty associated with this estimate.

4.4 Calculating AUPRC

The gold standards for the datasets used in this paper do not necessarily perfectly overlap with the genes and TFs that make up the rows and columns of A as defined by the prior hyperparameters i.e. there may be genes and TFs in the gold standard with a recorded interaction or lack of interaction, that do not appear in our model at all because they are not present in the prior. The reverse is also true: the prior may contain genes and TFs that

are not in the gold standard. For this reason, we compute the AUPRC using one of two methods: 'keep all gold standard' or 'overlap', which correspond to evaluating only interactions that are present in the gold standard or only interactions that are present in both the gold standard and the prior/posterior. We present results with 'keep all gold standard' AUPRC as the evaluation metric when comparing our model to the Inferelator in Figures 2 and 3. For our evaluation of uncertainty calibration (Figure 2 D), we use the overlap AUPRC so that bins containing a lower number of posterior means do not have artificially deflated AUPRCs (see the Evaluating Calibration of Posterior Uncertainty part of the Methods Section for further information).

4.5 Evaluating Calibration of Posterior Uncertainty

We create 10 bins, corresponding to the lowest 10%, 20%, 30% and so on of posterior variances. We place the posterior point estimates of TF-gene interactions associated with these variances into these bins and then calculate the 'overlap AUPRC' for each bin using the corresponding gold standard. The AUPRC for each bin is calculated using those interactions that are in the gold standard and also in the bin. We use such a cumulative binning scheme because using a non-cumulative scheme could result in some bins having very small numbers of posterior interactions that are present in the gold standard, which would lead to noisier estimates of the AUPRC.

4.6 Inference and Evaluation on Multiple Observations of W

The Inferelator method applies two scRNA-seq experiments separately on *S. cerevisiae*, with each resulting in a distinct model. These models are used to infer TF-gene interaction matrices, which are then sparsified. The final matrix is obtained by taking the intersection of the two matrices and retaining only the entries that are non-zero in both matrices. In our approach, we also train a separate model on each expression matrix, and obtain a posterior mean matrix for A for each of them. To obtain the final posterior mean matrix for A, we average the posterior mean matrices from each model. While this approach works well, future research could focus on explicitly modeling separate expression matrices within the model, as discussed in the Conclusion section.

4.7 Measuring the Impact of Prior Hyperparameters

We evaluate the utility of each of the prior hyperparameter matrices used in our experiments. In Figures 2A and 3A, we present with grey dots the AUPRCs achieved when performing inference using shuffled prior hyperparameters for A. This corresponds to randomly assigning to each row (gene) of A, the prior hyperparameters that correspond to a different row of A. Shuffling the hyperparameters should lead to worse performance, as the posterior estimates should then also be shuffled, whereas the row/column labels for the posterior will remain unshuffled. For the 'no prior' setting, shown with black dots in the figures, we set $\bar{A}_{mk} = 0 \; \forall \; m, k$. The difference in AUPRC achieved using the unshuffled vs shuffled or no hyperparameters measures the usefulness of the provided hyperparameters for the inference task on the dataset in question.

4.8 Cross-Validation

For each model organism, *S. cerevisiae* and *B. subtilis*, we perform a five-fold cross validation experiment. Cross-validation is performed by partitioning the gold standard into an 80% - 20% split, where 80% of the data represents prior-known information to be used as a prior for $p(A)$, and the remaining 20% is treated as the gold standard for evaluation. This process is repeated five times to generate five random splits of the data in order to robustly evaluate GRN inference. It is important to note that PMF-GRN performs hyperparameter search before inferring a final GRN within each cross-validation split. For each of the five partitioned cross-validation folds the 80%, or prior portion, is further split into 80% train and 20% test for hyperparameter search and evaluation. Once the optimal hyperparameters have been determined, the initial 80% split is treated as the training data, while the remaining 20%, which was not seen during hyperparameter selection, is used for evaluation.

4.9 Datasets and Preprocessing

We inferred each GRN using a single-cell RNA-seq expression matrix, a TF-target gene connectivity matrix, and a gold standard for bench-marking purposes. We modeled the single-cell expression matrices based on the raw UMI counts obtained from sequencing for the *S. cerevisiae* datasets, which were therefore not normalized for the purpose of this work. For the two *B. subtilis* datasets used in this work, we demonstrate the effect of different normalization and scaling techniques, and convert all data used to integers in order to create a single-cell-like dataset. We further obtained binary TF-gene matrices representing prior-known interactions, which served as prior hyperparameters over $\mathbf{A}$, and were derived from the YEASTRACT and subtiwiki databases. We acquired a gold standard for *S. cerevisiae* our datasets from independent work which is detailed below.

Saccharomyces cerevisiae

We used two raw UMI count expression matrices for the organism *S. cerevisiae* obtained from NCBI GEO (GSE125162 (8) and GSE144820 (39)). For this well studied organism, we employed the YEASTRACT (46; 47) literature derived network of TF-target gene interactions to be used as a prior over A in both *S. cerevisiae* networks. A gold standard for *S. cerevisiae* was additionally obtained from a previously defined network (48) and used for bench-marking our posterior network predictions.We note that the gold standard is roughly a reliable subset of the YEASTRACT prior. Additional interactions in the prior can still be considered to be true but have less supportive evidence than those in the gold standard.

Bacillus subtilis

We used two microarray datasets for *B. subtilis*, which we label as B1 (GSE27219) and B2 (GSE67023). Both B1 and B2 underwent different normalization as part of standard microarray processing, described in detail in (40) and (41). For the experiment "No Normalization", B1 was simply converted to integers, while B2 contained negative numbers and had to be scaled and then converted to integers so that the data represented positive integers similar to single-cell data.

For this well studied organism, we use the subtiwiki database (42; 43; 44) to obtain a network of prior-known TF-target gene interactions to be used as a prior over A as well as a gold standard for benchmarking posterior predictions.

5 Data Availability

The datasets used in this work are publicly available. They are referenced in the Methods section and are available through `https://github.com/nyu-dl/pmf-grn`.

6 Code Availability

Code, inferred GRNs, and inference and evaluation scripts can be found at `https://github.com/nyu-dl/pmf-grn`.

7 Author Contributions

CSG and KC contributed to Conceptualization of the project. OM and KC designed the probabilistic model. OM implemented PMF-GRN Software, Experiments and Validation. CSG implemented PMF-GRN Experiments, Validation, and Inferelator Software. OM, CSG, and KC contributed to Methodology, Software, Validation, Formal Analysis, Visualization, and Writing Original Draft Preparation. CSG contributed to Data Curation. KC and RB contributed to Supervision, Project Administration and Funding Acquisition.

8 Acknowledgements

We thank members of the Bonneau lab for insightful discussions and feedback on this manuscript. We also thank the the staff of the NYU IT High Performance Computing and Flatiron Institute Scientific Computing Core. This work was supported by Samsung Advanced Institute of Technology (under the project *Next Generation Deep Learning: From Pattern Recognition to AI*); NSF Award 1922658 NRT-HDR: FUTURE Foundations, Translation, and Responsibility for Data Science; the National Institutes of Health (RM1HG011014, R01NS116350, R01NS118183, R01AI130945); and the Simons Foundation.

References

[1] Hecker M, Lambeck S, Toepfer S, Van Someren E, Guthke R. Gene regulatory network inference: data integration in dynamic models—a review. Biosystems. 2009;96(1):86-103.

[2] Chai LE, Loh SK, Low ST, Mohamad MS, Deris S, Zakaria Z. A review on the computational approaches for gene regulatory network construction. Computers in biology and medicine. 2014;48:55-65.

[3] Karlebach G, Shamir R. Modelling and analysis of gene regulatory networks. Nature reviews Molecular cell biology. 2008;9(10):770-80.

[4] Äijö T, Lähdesmäki H. Learning gene regulatory networks from gene expression measurements using non-parametric molecular kinetics. Bioinformatics. 2009;25(22):2937-44.

[5] Nachman I, Regev A, Friedman N. Inferring quantitative models of regulatory networks from expression data. Bioinformatics. 2004;20(suppl_1):i248-56.

[6] Burdziak C, Azizi E, Prabhakaran S, Pe'er D. A nonparametric multi-view model for estimating cell type-specific gene regulatory networks. arXiv preprint arXiv:190208138. 2019.

[7] Allaway KC, Gabitto MI, Wapinski O, Saldi G, Wang CY, Bandler RC, et al. Genetic and epigenetic coordination of cortical interneuron development. Nature. 2021;597(7878):693-7.

[8] Jackson CA, Castro DM, Saldi GA, Bonneau R, Gresham D. Gene regulatory network reconstruction using single-cell RNA sequencing of barcoded genotypes in diverse environments. Elife. 2020;9:e51254.

[9] Ciofani M, Madar A, Galan C, Sellars M, Mace K, Pauli F, et al. A validated regulatory network for Th17 cell specification. Cell. 2012;151(2):289-303.

[10] Ji Z, He L, Regev A, Struhl K. Inflammatory regulatory network mediated by the joint action of NF-kB, STAT3, and AP-1 factors is involved in many human cancers. Proceedings of the National Academy of Sciences. 2019;116(19):9453-62.

[11] Yosef N, Shalek AK, Gaublomme JT, Jin H, Lee Y, Awasthi A, et al. Dynamic regulatory network controlling TH17 cell differentiation. Nature. 2013;496(7446):461-8.

[12] Mercatelli D, Scalambra L, Triboli L, Ray F, Giorgi FM. Gene regulatory network inference resources: A practical overview. Biochimica et Biophysica Acta (BBA)-Gene Regulatory Mechanisms. 2020;1863(6):194430.

[13] Huynh-Thu VA, Irrthum A, Wehenkel L, Geurts P. Inferring regulatory networks from expression data using tree-based methods. PloS one. 2010;5(9):e12776.

[14] Wang Y, Joshi T, Zhang XS, Xu D, Chen L. Inferring gene regulatory networks from multiple microarray datasets. Bioinformatics. 2006;22(19):2413-20.

[15] Chang C, Ding Z, Hung YS, Fung PCW. Fast network component analysis (FastNCA) for gene regulatory network reconstruction from microarray data. Bioinformatics. 2008;24(11):1349-58.

[16] Dufva M. Introduction to microarray technology. DNA Microarrays for Biomedical Research: Methods and Protocols. 2009:1-22.

[17] Wang Z, Gerstein M, Snyder M. RNA-Seq: a revolutionary tool for transcriptomics. Nature reviews genetics. 2009;10(1):57-63.

[18] Saliba AE, Westermann AJ, Gorski SA, Vogel J. Single-cell RNA-seq: advances and future challenges. Nucleic acids research. 2014;42(14):8845-60.

[19] Akers K, Murali T. Gene regulatory network inference in single-cell biology. Current Opinion in Systems Biology. 2021;26:87-97.

[20] Lähnemann D, Köster J, Szczurek E, McCarthy DJ, Hicks SC, Robinson MD, et al. Eleven grand challenges in single-cell data science. Genome biology. 2020;21(1):1-35.

[21] Chen G, Ning B, Shi T. Single-cell RNA-seq technologies and related computational data analysis. Frontiers in genetics. 2019:317.

[22] Ochs MF, Fertig EJ. Matrix factorization for transcriptional regulatory network inference. In: 2012 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB). IEEE; 2012. p. 387-96.

[23] Alter O, Brown PO, Botstein D. Singular value decomposition for genome-wide expression data processing and modeling. Proceedings of the National Academy of Sciences. 2000;97(18):10101-6.

[24] Moloshok TD, Klevecz R, Grant JD, Manion FJ, Speier IV W, Ochs MF. Application of Bayesian decomposition for analysing microarray data. Bioinformatics. 2002;18(4):566-75.

[25] Kim PM, Tidor B. Subsystem identification through dimensionality reduction of large-scale gene expression data. Genome research. 2003;13(7):1706-18.

[26] Brunet JP, Tamayo P, Golub TR, Mesirov JP. Metagenes and molecular pattern discovery using matrix factorization. Proceedings of the national academy of sciences. 2004;101(12):4164-9.

[27] Gao Y, Church G. Improving molecular cancer class discovery through sparse non-negative matrix factorization. Bioinformatics. 2005;21(21):3970-5.

[28] Yang Z, Michailidis G. A non-negative matrix factorization method for detecting modules in heterogeneous omics multi-modal data. Bioinformatics. 2016;32(1):1-8.

[29] Duren Z, Chen X, Zamanighomi M, Zeng W, Satpathy AT, Chang HY, et al. Integrative analysis of single-cell genomics data by coupled nonnegative matrix factorizations. Proceedings of the National Academy of Sciences. 2018;115(30):7723-8.

[30] Hu X, Hu Y, Wu F, Leung RWT, Qin J. Integration of single-cell multi-omics for gene regulatory network inference. Computational and Structural Biotechnology Journal. 2020;18:1925-38.

[31] Skok Gibbs C, Jackson CA, Saldi GA, Tjärnberg A, Shah A, Watters A, et al. High-performance single-cell gene regulatory network inference at scale: the Inferelator 3.0. Bioinformatics. 2022.

[32] Jansen C, Ramirez RN, El-Ali NC, Gomez-Cabrero D, Tegner J, Merkenschlager M, et al. Building gene regulatory networks from scATAC-seq and scRNA-seq using linked self organizing maps. PLoS computational biology. 2019;15(11):e1006555.

[33] Van de Sande B, Flerin C, Davie K, De Waegeneer M, Hulselmans G, Aibar S, et al. A scalable SCENIC workflow for single-cell gene regulatory network analysis. Nature Protocols. 2020;15(7):2247-76.

[34] Kamimoto K, Stringa B, Hoffmann CM, Jindal K, Solnica-Krezel L, Morris SA. Dissecting cell identity via network inference and in silico gene perturbation. Nature. 2023:1-10.

[35] Äijö T, Bonneau R. Biophysically motivated regulatory network inference: progress and prospects. Human heredity. 2016;81(2):62-77.

[36] Mnih A, Salakhutdinov RR. Probabilistic matrix factorization. Advances in neural information processing systems. 2007;20.

[37] Blei DM, Kucukelbir A, McAuliffe JD. Variational inference: A review for statisticians. Journal of the American statistical Association. 2017;112(518):859-77.

[38] Ranganath R, Gerrish S, Blei D. Black box variational inference. In: Artificial intelligence and statistics. PMLR; 2014. p. 814-22.

[39] Jariani A, Vermeersch L, Cerulus B, Perez-Samper G, Voordeckers K, Van Brussel T, et al. A new protocol for single-cell RNA-seq reveals stochastic gene expression during lag phase in budding yeast. elife. 2020;9:e55320.

[40] Nicolas P, Mäder U, Dervyn E, Rochat T, Leduc A, Pigeonneau N, et al. Condition-dependent transcriptome reveals high-level regulatory architecture in Bacillus subtilis. Science. 2012;335(6072):1103-6.

[41] Arrieta-Ortiz ML, Hafemeister C, Bate AR, Chu T, Greenfield A, Shuster B, et al. An experimentally supported model of the Bacillus subtilis global transcriptional regulatory network. Molecular systems biology. 2015;11(11):839.

[42] Michna RH, Zhu B, Mäder U, Stülke J. Subti Wiki 2.0—an integrated database for the model organism Bacillus subtilis. Nucleic acids research. 2016;44(D1):D654-62.

[43] Zhu B, Stülke J. Subti Wiki in 2018: from genes and proteins to functional network annotation of the model organism Bacillus subtilis. Nucleic acids research. 2018;46(D1):D743-8.

[44] Pedreira T, Elfmann C, Stülke J. The current state of Subti Wiki, the database for the model organism Bacillus subtilis. Nucleic Acids Research. 2022;50(D1):D875-82.

[45] Higgins I, Matthey L, Pal A, Burgess C, Glorot X, Botvinick M, et al. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In: International Conference on Learning Representations; 2017. Available from: `https://openreview.net/forum?id=Sy2fzU9gl`.

[46] Monteiro PT, Oliveira J, Pais P, Antunes M, Palma M, Cavalheiro M, et al. YEASTRACT+: a portal for cross-species comparative genomics of transcription regulation in yeasts. Nucleic acids research. 2020;48(D1):D642-9.

[47] Teixeira MC, Monteiro PT, Palma M, Costa C, Godinho CP, Pais P, et al. YEASTRACT: an upgraded database for the analysis of transcription regulatory networks in Saccharomyces cerevisiae. Nucleic acids research. 2018;46(D1):D348-53.

[48] Tchourine K, Vogel C, Bonneau R. Condition-specific modeling of biophysical parameters advances inference of regulatory networks. Cell reports. 2018;23(2):376-88.

[49] Faria JP, Overbeek R, Taylor RC, Conrad N, Vonstein V, Goelzer A, et al. Reconstruction of the regulatory network for Bacillus subtilis and reconciliation with gene expression data. Frontiers in Microbiology. 2016;7:275.

Supplementary Information: Probabilistic Matrix Factorization for Gene Regulatory Network Inference

A Supplementary Tables

Method	Prior Information		
	Regular	None	Shuffled
PMF-GRN	0.375	0.014	0.023
AmUSR	0.223	0.024	0.019
BBSR	0.402	0.022	0.018
StARS	0.186	0.028	0.017
Scenic	0.014	0.014	0.014
CellOracle	0.383	N/A	0.013

Supplementary Table 1: AUPRCs achieved by PMF-GRN, the Inferelator algorithms (AMuSR, BBSR, and StARS), Scenic and CellOracle on *S. cerevisiae* datasets.

Method	Cross Validation Split				
	Split 1	Split 2	Split 3	Split 4	Split 5
PMF-GRN	0.114	0.096	0.086	0.1342	0.118
BBSR	0.112	0.128	0.161	0.171	0.139
StARS	0.109	0.137	0.154	0.195	0.151
Scenic	0.020	0.021	0.018	0.025	0.021
CellOracle	0.034	0.042	0.034	0.043	0.034

Supplementary Table 2: AUPRCs achieved by PMF-GRN, the Inferelator algorithms (AMuSR, BBSR, and StARS), Scenic and CellOracle on *S. cerevisiae* datasets using the gold standard for 5-fold cross validation.

Method	Noise Added			
	No Noise	100% Noise	250% Noise	500% Noise
PMF-GRN	0.343	0.280	0.198	0.149
BBSR	0.264	0.208	0.186	0.174
StARS	0.136	0.125	0.114	0.118
Scenic	0.075	0.068	0.059	0.055
CellOracle	0.417	0.306	0.226	0.175

Supplementary Table 3: AUPRCs achieved by PMF-GRN, the Inferelator algorithms (AMuSR, BBSR, and StARS), Scenic and CellOracle on *S. cerevisiae* datasets using increasing amounts of noise added to the prior-knowledge data.

B Supplementary Methods

B.1 TF Target Gene Connectivity Matrix Generation

B.1.1 Saccharomyces cerevisiae

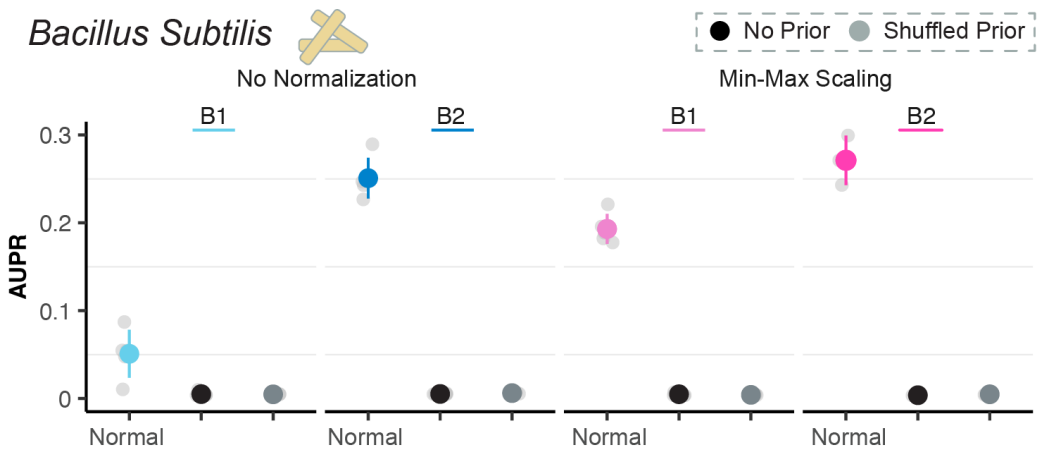
Datasets were obtained from (31) without further modification.

B.1.2 Bacillus subtilis

A prior-known TF-target gene interactions matrix was obtained from the Subtiwiki database (49) from "regulations" (downloaded 07/21/22). Using the columns "regulator locus" and "gene locus" a cross-tab integer matrix was created, where 1 represents the existence of an interaction and 0 represents no interaction. This matrix was randomly split 5 times in 80%-20% proportions along the gene axis to generate independent prior-known information and gold standard matrices.

To demonstrate the importance of scaling microarray data to place independently collected datasets on the same scale, we demonstrate how Min-Max Scaling improves inference in both *B. subtilis* datasets. For "Min-Max Scaling", both B1 and the positive scaled B2 dataset were subsequently normalized using the following logic.

Using the observation axis, values were linearly transformed so that the minimum value was mapped to 0 and the maximum value was mapped to 1. Each value was then multiplied by 100 and converted to integers to produce the resulting expression matrix of scaled single-cell-like integers.



Supplemental Figure S1: Results for GRNs learned in *B. subtilis* datasets B1 and B2 comparing "No Normalization" to "Min-Max Scaling". Light gray dots represent the results for each of the 5 cross-validation experiments. Colored dots represent the mean of the cross-validation experiments ± standard deviation. Negative controls are demonstrated by black dots for "No Prior" and grey dots for "Shuffled Prior".

Method	*B. subtilis* Cross-Validation Dataset B1		
	Regular	No Prior	Shuffled
No Normalization	0.0509 ± 0.0273	0.0048 ± 0.0003	0.0050 ± 0.0028
Min-Max Scaling	0.1931 ± 0.0171	0.0042 ± 0.0003	0.0042 ± 0.0018

Supplementary Table 4: AUPRCs achieved by PMF-GRN on *B. subtilis* B1 dataset. Results are reported as the mean AUPRC across five 'cross-validation' splits ± standard deviation

Method	*B. subtilis* Cross-Validation Dataset B2		
	Regular	No Prior	Shuffled
No Normalization	0.2508 ± 0.0232	0.0062 ± 0.0006	0.0052 ± 0.0004
Min-Max Scaling	0.2886 ± 0.0312	0.0048 ± 0.0008	0.0038 ± 0.0005

Supplementary Table 5: AUPRCs achieved by PMF-GRN on *B. subtilis* B2 dataset. Results are reported as the mean AUPRC across five 'cross-validation' splits ± standard deviation

B.2 Inferelator, Scenic, and CellOracle Networks

B.2.1 Saccharomyces cerevisiae

Networks were inferred using the "multitask" workflow setting of the Inferelator for the same single-cell *S.cerevisiae* datasets described in (31). For each algorithm, BBSR, StARS, and AMuSR, the following parameters were used: gold_standard_filter_method="keep_all_gold_standard", num_bootstraps=5. Aggregated multi-task networks were used for benchmarking, while single-task networks were disregarded for the purpose of this work. To make these networks directly comparable to PMF, we did not make use of normalization, count minimum, or meta-data options available within the Inferelator workflow.

Networks inferred with Scenic and CellOracle used the same input files, with no additional parameters specified.