Uncovering Functional Gene Regulatory Networks in Bulk and Single-Cell Data through Robust Transcription Factor Activity Estimation and Model-Guided Experimental Validation

Alireza Fotuhi Siahpirani[1,2], Sunnie Grace McCalla[1,3], Saptarshi Pyne[1], Caleb Dillingham[1], Rupa Sridharan[1,4], and Sushmita Roy[1,5*]

[1]Wisconsin Institute for Discovery, University of Wisconsin-Madison

[2]Department of Computer Sciences, University of Wisconsin-Madison

[3]Laboratory of Genetics, University of Wisconsin-Madison

[4]Department of Cell and Regenerative Biology, University of Wisconsin-Madison

[5]Department of Biostatistics and Medical Informatics, University of Wisconsin-Madison

[*]To whom correspondence should be addressed.

Abstract

Reconstructing genome-scale gene regulatory networks (GRNs) remains a difficult problem in systems biology, and many experimental and computational methods have been developed to address this problem. Recent computational methods have aimed to more accurately model GRNs by estimating the hidden Transcription Factor Activity (TFA), from prior knowledge of TF target regulatory connections, encoded as an input directed graph, to relax the assumption that mRNA level of the regulator correlates with the protein activity of the regulator. However, the noise in the prior knowledge can adversely affect the estimated TFA levels and the quality of the downstream inferred GRNs. Here, we present a new approach, MERLIN+P+TFA, that uses prior knowledge-guided sparsity regularization to robustly and accurately estimate TFA and downstream GRNs. We apply our method to simulated and real expression data in yeast and mammalian systems and show improved quality of inferred GRNs for both bulk and single-cell datasets. Regularized TFA offers benefits to a variety of other GRN inference algorithms, including those that have traditionally be used with expression alone, in both bulk and scRNA-seq settings. We used the inferred GRN to prioritize key regulators for the mouse Embryonic Stem Cell (mESC) state and validate 58 regulators experimentally. We identify both known and novel regulators of the mESC state and further validate the targets of 4 known and novel regulators. Our validation experiments suggest that computationally inferred networks can capture functional targets of TFs with higher precision than estimated in current benchmarks, however, it is important to generate context-specific gold standards.

Introduction

Transcriptional gene regulatory networks determine the context-specific expression levels of genes by specifying which regulatory proteins such as transcription factors and signaling proteins, regulate the expression of a gene. Expression-based network inference is a powerful way to reconstruct an initial genome-scale gene regulatory network. Given genome-wide mRNA profiles, these methods predict the regulators of a gene based on the ability of a regulator to explain the variation in the mRNA level of a target gene. Current methods for network inference can be grouped into those that rely on expression alone [1] and those that combine gene expression with other data sources, such as sequence-specific motif information, ChIP-seq datasets as prior knowledge to constrain the network structure [2–8]. A major limitation of these approaches is that mRNA levels of regulators are assumed to correlate with its activity levels, which may not be necessarily true due to post-translational regulation of proteins. To address this issue, a recent direction has been to estimate latent transcription factor activity (TFA) levels from the noisy input prior network, which is also used to constrain the network structure [9–12]. However, the prior knowledge of regulatory interactions is typically noisy, which can significantly influence the quality of the estimated TFA and the subsequent network inference.

Here, we present MERLIN+P+TFA, a regulatory network inference algorithm that extends our previous prior-based inference algorithm, MERLIN+Prior (MERLIN+P) [7], by incorporating a sparsity inducing approach to estimate "regularized TFA" that is robust to noisy input networks. MERLIN+P+TFA extends the Network Component Analysis (NCA) method to incorporates uncertainty in the input prior networks to enable data-driven selection of the most reliable edges in the TFA estimation. To demonstrate the broad applicability of MERLIN+P+TFA to mammalian species, we collected a large compendium of expression data spanning in four commonly used mammalian cell lines, and inferred TFA and regulatory networks. Based on extensive empirical analyses on both simulated and real expression data from yeast, human and mouse species, we show that such estimated TFA is more robust to noise, is more accurate and infers better networks compared to an approach that naively uses the noisy input network. Furthermore, the estimated TFA values are beneficial for other popular expression-based network inference algorithms included several that have previously not been tested with TFA for both buk and single cell expression datasets. In both human and mouse systems, regularized TFA improves the quality of inferred networks and prioritizes novel

regulators that could not be found using expression alone.

We experimentally validate 58 prioritized regulators for the mouse embryonic stem cell (mESC) state using knockdown experiments and phenotypic assays and recover known and novel mESC regulators. We functionally test the predicted target genes of 8 TFs using RT-pCR and find that many of the predicted targets have a significant change in expression indicative of a functional target. Taken together, MERLIN+P+TFA is a powerful tool for reconstructing gene regulatory networks and is broadly useful across diverse systems for both bulk and single cell gene expression datasets.

Results

MERLIN+P+TFA: inferring genome-scale GRNs with regularized transcription factor activity (TFA) levels

MERLIN+P+TFA combines expression-based gene regulatory network inference with TFA estimation to build a genome-scale gene regulatory network. The MERLIN+P+TFA algorithm takes as input the gene expression matrix, a noisy prior network and a list of candidate regulator, and performs network inference in two steps (**Figure 1**): 1) estimate TFA using a new approach of NCA, NCA-LASSO, based on regularized network component analysis (NCA) [13], and 2) infer gene regulatory connections using MERLIN+P, a previously developed method [7] (**Materials and methods**), which combines gene expression with prior information. MERLIN+P and MERLIN+P+TFA use a "dependency network" [14], a type of "probabilistic graphical models" (PGMs) [15], to represent the statistical dependencies between the regulators and the target genes. The prior network represents putative regulatory relationships between regulators and genes. Given this noisy prior network and the expression matrix, the regularized NCA step iteratively estimates the TFA and an updated connectivity matrix. The original NCA algorithm estimated TFA using a pseudo inverse operation, while the connectivity matrix is updated in turn by a per-gene neighborhood selection from a set of candidate regulators from the prior network. In our regularized version, NCA-LASSO, we leverage the confidence of edge between the regulator and target gene within an L1-based regression framework to constrain the neighborhood selection (**Materials and methods**). MERLIN+P takes the estimated TFA matrix and the gene expression matrix and infers a set of regulators from their gene expression levels and, if

available, estimated TF activities. Additionally, MERLIN+P+TFA identifies gene modules where the genes having similar regulatory programs are assigned to the same gene module. MERLIN+P+TFA is applied in a stability selection mode to estimate edge and module membership confidence.

Regularized NCA is beneficial for TFA estimation and inferring GRNs with simulated data

We first assessed the performance of MERLIN+P+TFA's regularized NCA estimation algorithm, NCA+LASSO and downstream network inference when considering input prior networks with different extents of noise. To this end, we performed a simulation experiment with ground truth TFA and network structures, which were used to simulate an expression dataset (**Methods**, **Supp table S1**). The prior networks were created by adding noise by randomly adding and removing edges from the true network and assigning edge weights using two different beta distribution whose parameters, a and b can be used to change the skew of the distribution. In the "uniform" setting $(a = 1, b = 1)$ (**Supp figure S1**), the edge weights of the true and false edges were set at random, and did not discriminate the true/false status of the edges. In the "non-uniform" setting $(a = 5, b = 1)$, the true edges tend to have higher edge confidence than the false edges. For each of the edge weight schemes, we prepared three versions of the prior network by introducing 10%, 30%, and 50% noise by randomly adding or removing edges. The simulated expression dataset along with the prior network were used to estimate the TFA. Subsequently, we used the estimated TFA, the prior network, and the simulated expression dataset to infer a network using the MERLIN-P algorithm. This allows us to compare the estimated TFA to true TFA, and inferred network to true network.

We first compared the estimated TFA using the original NCA algorithm from Liao et al [13], and true TFA across different noisy edge and edge weight settings. As expected, when we increase the noise level in the input prior network, the agreement between the estimated TFA and true TFA decreases irrespective of the edge weight scheme (**Figure 2A, B**). For example, the median correlation decreases from 0.79 to 0.57 to 0.44 when the noise level increases from 10% to 30% to 50% in the case of uniform edge weights (**Figure 2A**). We also observed that the networks inferred from the estimated TFA have lower agreement (as measured by AUPR) to the true network as we increase the noise, again irrespective of the edge weight scheme (**Figure 2B**). When using non-uniform edge weights in the prior, the AUPR values at the same noise level are higher than that with uniform edge weights. This is because MERLIN+P incorporates the edge

weight information in the inference process [7]; hence, a prior network with more accurate edge weights results in a more accurate inferred network, while a more noisy prior network produces an inferred network of a lower quality and lower AUPR.

Next, we asked whether incorporating prior edge weights through regularization can improve the quality of the estimated TFA. We used the prior network with 10%, 30%, and 50% noise and estimated TFA using NCA+LASSO at different regularization levels (**Section Simulation experiments**), and compared the estimated TFA to the true TFA based on absolute correlation (**Figure 2C, D**). For both uniform and non-uniform edge weight distributions, the correlation between the true and NCA-LASSO estimated TFA improve and are higher than the correlation of true TFA to TFA profiles estimated using unregularized NCA (NTFA, **Figure 2C, D**). When the edge weights follow the non-uniform distribution (**Figure 2D**), regularization significantly improves the correlation between the true TFA and estimated TFA. Finally, the AUPR values of the networks inferred from the NCA+LASSO-based TFA are higher than networks inferred using TFA from NCA (**Figure 2E**). As in the correlation of TFA values, the difference in AUPRs between the NCA and NCA-LASSO is more striking for the non-uniform edge weights. Taken together these results suggest that regularized NCA methods such as NCA-LASSO offer an important advantage with noisy prior networks both for estimating the activity of the TFs and also for downstream GRN inference.

Regularized TFA is beneficial for diverse network inference methods

We next investigated the performance of regularized TFA of NCA-LASSO on a real dataset from the yeast, *Saccharomyces cerevisiae*. We used a motif-based network as input prior and expression from a large scale natural variation dataset (**Materials and methods**). We estimated TFAs and downstream regulatory networks using both regularized and unregularized versions of the NCA algorithm. For network inference, in addition to MERLIN-P, we included several other network inference methods, namely GENIE3 [16], Inferelator [6, 17], TIGRESS [18], mLASSO-StARS [12], NetREX [19], MERLIN [20], and MERLIN+P [7]. Among these methods, MERLIN+P+TFA, mLASSO-StARS, Inferelator, and NetREX explicitly infer TFA and utilize the estimated TFAs for network inference. MERLIN+P+TFA, MERLIN+P, Inferelator, NetREX, and mLASSO-StARS also incorporate the prior network during network inference in addition to using the prior for TFA estimation.

We evaluated the inferred networks based on AUPR scores with respect to three gold standard networks (**Section Real datasets**), regulator ChIP binding (ChIP), regulator knock out or knock down (KD), and the intersection of the ChIP and KD gold standard networks using different ways of TFA estimation (**Figure 3**). When an algorithm had hyper-parameters (e.g., regularization of TFA), we used the setting with the highest AUPR for that algorithm. Furthermore, for the regularized TFA, we selected the regularization level that gave the highest AUPR of all the regularization levels tested to use for further comparisons (**Supp file 1**; **Supp file 2**). MERLIN+P+TFA outperformed all other algorithms across all three gold standard networks when comparing methods that use expression alone, expression with prior, or unregularized TFA (**Figure 3A**). Interestingly, methods that explicitly incorporate unregularized TFA do not always outperform methods that do not (**Figure 3A** gold bars). When comparing networks inferred without TFA, (**Figure 3A** blue bars), incorporation of prior (MERLIN+P) is better than not using prior for all three gold standard networks. We next asked whether regularized TFA is beneficial for methods other than MERLIN-P including methods like GENIE3 and TIGRESS which were developed for expression alone. We find that compared to not using TFA or unregularized TFA, incorporation of TFA is beneficial for network inference for all methods compared (**Figure 3B**, green vs gold bars).

As AUPR is a coarse metric for assessing the entire inferred network structure, we next examined per TF recovery. To do so, we counted the number of "predictable TFs" in each inferred network with respect to a given gold standard network with different TFA settings. This is defined as the number of TFs whose targets in the inferred network have a significant overlap with their targets in the gold standard network (**Section Real datasets**). Across all methods compared, the highest number of predictable TFs is achieved when they utilized prior or TFA or both along with the expression data (**Figure 3C**). When comparing regularized vs unregularized TFA, most methods either benefited or did not suffer in performance (**Figure 3D**). Exceptions to this were mLASSO-StARS on ChIP and MERLIN+P+TFA on KD and ChIP+KD, which exhibited a small reduction in performance. We also examined the enrichment metric for each TF and found that the overlap between its true and predicted targets tends to be more significant when TFA is utilized along with the expression data (**Figure 4**). Overall, we observed that the addition of TFA improves the number of predictable TFs, and regularized TFA performs better than original TFA in almost all the cases (11 out of 16 cases while having ties in 2 cases).

MERLIN+P+TFA for inferring genome-scale transcriptional regulatory networks in mammalian cell lines

Having demonstrated the benefits of MERLIN+P+TFA's regularized TFA in yeast, we next applied it to mammalian datasets coming from four different cell lines: the mouse embryonic stem cell (mESC) line, human ESC (hESC) line, human Lymphoblastoid cell line (LCL), and human Michigan Cancer Foundation-7 cell line (MCF7). For each of the cell lines we collected a large dataset of published RNA-seq expression samples in mouse embryonic stem cell (**Section Real datasets**). For mESC, we additionally collected a microarray dataset resulting in a total of 5 datasets. For each of the datasets, we used a motif-based prior network filtered by DNase I footprints, and estimated TFA at different levels of regularization (**Section MERLIN+P+TFA for GRN inference from bulk gene expression from yeast and mammals**). We applied MERLIN and MERLIN-P with expression alone (no TFA), with unregularized TFA based on Network Components analysis (NTFA) and regularized TFA (RegTFA). We compared the inferred networks based on AUPR, and predictable TFs across three gold standards for each cell line and dataset (**Figure 5**). For networks inferred using RegTFA, we selected the regularization setting that returned the highest AUPR for further comparisons (**Supp file 3**; **Supp file 4**). Based on global AUPRs (**Fig 5**), both MERLIN and MERLIN-P had higher AUPRs when using TFA versus not and especially with regTFA. Furthermore, MERLIN+P obtained higher AUPR scores than MERLIN; further demonstrating the utility of using priors for network inference. Across the three types of gold standards, addition of prior was much more useful on ChIP and ChIP-KO gold standards.

We next examined the per TF recovery of the structure based on the predictable TF metric. MERLIN achieved the highest number of predictable TFs for regularized TFA for 4 out of the 13 cases (each case corresponding to a dataset-gold standard combination) and was higher compared to expression alone for 6 out of 13 cases (**Figure 6**). On the other hand, MERLIN+P achieved its highest numbers of predictable TFs with regularized TFA for 5 out of 13 comparisons. With respect to comparing MERLIN to MERLIN+P per TFA treatment for the ChIP gold standard, MERLIN+P always predicted more predictable TFs than that of MERLIN across all datasets, except for expression alone and regularized TFA for hESC (**Figure 6C**). We observe a similar behavior in the other cell lines as well for the other gold standards and datasets as well (**Supp figure S2- S6**). The exceptions to this are a few KD based comparisons of the mESC array dataset

(**Figure 6B**)and hESC dataset where MERLIN with TFA was worse (**Figure 6C**). For the ChIP gold standard intersected with the KD gold standard, MERLIN+P performed better than MERLIN for mESC RNA-seq and the LCL cell line, but this pattern was not consistently replicated for the mESC array and hESC dataset applications (**Figure 6A-C**). Overall, for the mammalian cell lines, the predictable TFs metric was best when using regularized TFA demonstrating the advantage od using MERLIN+P+TFA for mammalian datasets.

Experimental validation experiments for the mESC transcriptional regulatory network

Given MERLIN+P+TFA's good performance in both mammalian and yeast datasets, we used its predictions to find new regulators and targets important for the mouse ESC state. We first ranked regulators by an importance metric that assessed the change in predictive power of the inferred network if the regulator was removed from the network (**Section Additional evaluation metrics for real expression datasets**). We considered both array and RNA-seq-based networks inferred using both MERLIN and MERLIN-P (**Figure 7A**). We evaluated this ranking based on how highly ranked were genes known to be important for the ESC state. To create the list of known ESC genes, we combined an experimentally derived gene sets from Wong et al. [21] and Mueller et al. [22] available from MsigDB [23, 24] together with genes annotated with ESC relevant Gene Ontology (GO) terms [25], "regulation of embryonic development", "embryonic organ development". MERLIN+P+TFA achieved highest AUPR scores with regularized TFA for both array and RNA-seq networks. This pattern is true for MERLIN as well indicating the importance of regularized TFA for prioritizing important cell-state specific regulators. Among the highly ranked genes were Esrrb and Nanog (**Figure 7B**) that are key componets of the pluripotency GRN.

We next used these rankings to identify novel regulators important for maintenance of the ESC state and also to validate regulators and their targets. Briefly, we took the union of the top 10% of regulators in all 10 MERLIN+P+TFA networks (both array and RNA-seq, for all TFA regularization settings i.e. all settings excluding "No TFA"). The union resulted in a list of 205 regulators. We examined the literature for these regulators and excluded 147 regulators already known to be important for the maintenance of or differentiation from the ESC state (**Supp table S2**). The remaining 58 regulators previously not associated with pluripotency were tested for their association with pluripotency using small interfering RNA (siRNA) knockdown experiments. Briefly, we used siRNAs of each of the regulators at a time and measured the ex-

pression of Nanog, a key marker of pluripotency. We did these experiments in the 2D4 and E14 ESC lines. We compared the distribution of Nanog in the regulator-targeted cell population and a control population targeted by a control siRNA using the Kolmogorov-Smirnov (KS) test (**Table 7, Supp file 5**). Nanog's expression was measured with the Nanog-GFP reporter in the 2D4 cell line and through intracellular staining in the E14 cell line. Out of these 58 novel regulators, 19 regulators showed significant decrease in pluripotency of the 2D4 cell line and 18 showed significant decrease in the E14 cell line (FDR corrected p-value¡1E-10). Among these regulators, 10 are common in both the cell lines. There are 11 and 23 TFs in the 2D4 and E14 cell lines, respectively, that show significant increase (p-value$<$1E-10) in pluripotency as a result of knock down (7 of them are common in both the cell lines).

We next tested the high-confidence targets corresponding to the high-confidence edges in the MERLIN+P+TFA inferred network of 4 novel TFs that are not present in the mESC gold standard networks (namely, Fosl1, Gtf2a2, Zbtb7b, Lztr1) and 4 known TFs (Esrrb, Nanog, Oct4, Zfp281), by measuring the expression of the targets (with RT-PCR) after the siRNA knockdowns of these TFs. We examined the fold changes in the expression levels of the targets compared to that of the non-target negative controls (Kdm3b, Cbx3, and AF9, dashed red line **Figure 7D**). 39 out of the 81 tested TF-target pairs (48%) show significant change in expression compared to control. Of these 39 pairs, 5 correspond to the targets of the 4 novel TFs. Of the remaining 34 pairs, 20 show ChIP support and 20 show KD support (14 are common in both). The remaining 34 pairs correspond to the targets of the known TFs. Among them, 24 TF-target pairs were already part of the ChIP and KD gold standard networks (6 pairs belonging to ChIP only, 5 pairs belonging to KD only, and 13 pairs belonging to both ChIP and KD; see **Supp file 6** in Supplementary). Subsequently, we investigated how many of these 39 TF-target pairs had been predicted using the estimated TFA profiles of the regulators. It was observed that 20 out of 39 pairs had been predicted using the estimated TFA profiles of the corresponding regulators while the remaining 19 pairs had been predicted using the gene expression levels of the regulators; no pairs had been predicted using both the TFA profiles and gene expression levels of the regulators (see **Supp file 6- 8** in Supplementary). It shows that MERLIN+P+TFA was able to complement its ability to utilize gene expression data with its estimation of TFA profiles in uncovering and recapitulating regulatory edges. Taken together, these experiments suggest that our predicted regulators and high-confidence inferred interactions can provide useful insight into the regulatory mechanism of

pluripotency.

Regularized TFA improves network structure recovery for scRNAseq datasets

The availability of single-cell RNA sequencing (scRNAseq) datasets opens up new opportunities for genome-scale GRN inference by leveraging the variaiton in expression across individual cells. The effect of TFA incorporation for learning GRNs on scRNAseq datasets has been evaluated on a limited scale [26–28], which has indicated that TFA often increases network performance. Given the advantage of using MERLIN+P+TFA's regularized TFA on bulk gene expression data, we next examined the incorporation of regularized TFA in three network inference algorithms and measured their performance across six scRNAseq datasets.

We selected MERLIN [20] [7], Inferelator [29], and Single-Cell rEgulatory Network Inference and Clustering (SCENIC) [30] for further testing as these were among the top performing methods in our previous benchmarking study of scRNA-seq network inference methods [28]. Additionally, MERLIN and Inferelator incorporated priors which we could utilize for comparisons. Further, these three algorithms have been previously benchmarked [28] which allowed us to extend the scope of previous findings here with the addition of regularized TFA. We utilized datasets from McCalla *et al.* [28], which encompassed three different species from yeast, mouse, and human samples and represented dataset sizes of varying numbers of cells (**Materials and methods**). The smallest dataset contained 163 cells [31] while the largest dataset contained 17,396 cells [29]. As before, we used the Perturb, ChIP, and the intersection of Perturb and ChIP gold standards to evaluate the performance for each algorithm configuration. When networks inferred for an algorithm with multiple regularized TFA settings, the setting that returned the highest AUPR was used for comparisons (**Supp file 9**; **Supp file 10**).

Comparing network inference results within an algorithm to evaluate regularized TFA performance, SCENIC+RegTFA outperformed SCENIC and SCENIC+NTFA across all three gold standard networks for F-score (**Figure 8**A, **Supp file 9**; **Supp file 10**). Overall, MERLIN+P+NTFA or MERLIN+P+TFA outperformed MERLIN alone 5 out of 9 times across metric and gold standards. MERLIN+P+TFA was tied for best rank with MERLIN+Prior for the ChIP gold standard, but was not best for Perturb or Perturb+ChIP. When comparing the MERLIN TFA configurations for AUPR, MERLIN+P+TFA ranked higher

than MERLIN alone for two out of three gold standards (ChIP, Perturb+ChIP). Within SCENIC, AUPR performance progressively improved with the incorporation of TFA from SCENIC to SCENIC+TFA, and then from SCENIC+TFA to SCENIC+RegTFA, for both the Perturb and ChIP gold standards. This pattern was slightly disrupted for the Perturb+ChIP gold standard where SCENIC alone outperformed SCENIC+NTFA, although SCENIC+RegTFA still achieved the highest rank among the three SCENIC regularization configurations tested. Based on F-score for the Inferelator TFA configurations, Inferelator+P+NTFA was top ranking for Perturb and ChIP, but second best for Perturb+ChIP (**Figure 8**A). AUPR showed Inferelator+P+TFA, which uses Inferelator's inbuilt pseudo-inverse TFA estimation, performed best for the Perturb gold standard, while Inferelator+P+NTFA was best for ChIP. For the Perturb+ChIP gold standard, Inferelator with Prior (Inferelator+P) was ranked highest.

Regarding predictable TFs, Inferelator without TFA performed best for Perturb and Perturb+ChIP (Inferelator and Inferelator+Prior, respectively). Inferelator+P+TFA was ranked highest for the ChIP gold standard. When considering the gold standard networks individually, Inferelator+P+RegTFA ranked last for Perturb, ChIP, and Perturb+ChIP. However when performance was combined across all of the gold standards, Inferelator+P+RegTFA outperformed all of the other TFA configurations for Inferelator (**Figure 8**B-C). For MERLIN, predictable TF rankings across TFA configurations for MERLIN showed three distinct performance trends across gold standards. MERLIN alone achieved the highest rank for the Perturb gold standard while MERLIN+P ranked lowest, however this pattern was reversed for both the ChIP and Pertubr+ChIP gold standards where MERLIN alone had the lowest rank and MERLIN+P achieved the best performance. Across all three gold standards, TFA application (MERLIN+P+NTFA and MERLIN+P+TFA) ranked between either MERLIN or MERLIN+P, but notably MERLIN+P+TFA consistently outperformed MERLIN+P+NTFA across all gold standards when recovering predictable TFs. Algorithm assessment for SCENIC for predictable TFs followed a consistent trend in increasing rank as TFA configuration was applied from SCENIC alone, to SCENIC+NTFA, to SCENIC+RegTFA for both the Perturb and ChIP gold standards. The Perturb+ChIP gold standard slightly deviated from this pattern with SCENIC alone outperforming SCENIC+NTFA, but SCENIC+RegTFA still ranked highest.

Overall trends summarized across gold standards show algorithm configurations that include TFA are among the top-ranked (**Figure 8**B-C). The top 3 ranking algorithms for F-score include TFA for network

inference (Inferelator+P+NTFA, SCENIC+RegTFA, Inferelator+P+RegTFA). For AUPR, not only did the top ranked algorithm included TFA (Inferelator+P+TFA), the bottom three ranking algorithms did not incorporate a prior or TFA (MERLIN, SCENIC, Inferelator) (**Figure 8**B-C). MERLIN+P+TFA tied for the best ranked algorithm for the predictable TFs metric along with MERLIN+P. In fact, when evaluated for predictable TF performance, six out of the seven best performing algorithms included TFA for network inference. Specifically regarding TFA origin, RegTFA performed better than NTFA across F-score, AUPR, and predictable TF metrics for the MERLIN and SCENIC algorithm (**Figure 8**C). Taken together, these results demonstrate that MERLIN+P+TFA's regularized TFA can benefit network inference for scRNA-seq datasets as well.

Discussion

Reconstruction of genome-scale gene regulatory networks is a long-standing problem in gene regulation and remains experimentally and computationally challenging. Recent computational methods have advanced the field beyond purely expression-based network inference by incorporating network structure priors [6–8, 20, 32, 33]. The prior can not only constrain the inferred networks with biological prior knowledge, but also estimate the hidden transcription factor (TF) activity using Network Components Analysis (NCA [13]). The latter helps relax the assumption that mRNA and protein levels of the TF must correlate [9–12]. However, the prior can be noisy, which can adversely affect the quality of the estimated TFA and downstream network inference. To address this problem, we developed MERLIN+P+TFA, that uses regularized regression to selectively incorporate high confidence prior edges enabling more robust TFA estimation and network inference. Our experiments in yeast showed that the addition of regularized TFA improved the performance of not only MERLIN+P+TFA but also other state-of-the-art network inference methods that may or may not have originally used TFA.

Studies of TFA and GRN inference have thus far been conducted in yeast and bacteria with only a handful in multi-cellular organisms [12, 19]. Therefore, we extended our study to four well-studied mammalian cell lines: mouse embryonic stem cell (mESC), human ESC (hESC), human lymphoblastoid cell line (LCL), and human Michigan Cancer Foundation-7 cell line (MCF7) by systematically collecting expression data, prior information and gold standards in these cell lines. We applied both MERLIN-P and MERLIN+P+TFA to these datasets and showed that regularized TFA improved the quality of these inferred networks. Our expression, prior network and gold standards are available to the community to build new inference methods as well as to benchmark them.

Given the increasing availability of single cell RNAseq (scRNAseq) and their potential to capture cell type specific regulation, we studied the effect of regularized TFA for GRN inference from scRNAseq datasets. Across different datasets, we found that regularized TFA generally resulted in better inferred GRNs compared to unregularized TFA. SCENIC), regularized TFA was better than unregularized TFA across the different metrics used for GRN inference (**Figure 8**C). Finally, MERLIN+P+TFA consistently ranked in the top half of the 12 scRNAseq algorithm comparisons across all evaluation metrics (**Figure 8**C).

An unmet challenge in the field of GRN inference is the lack of sufficient ground truth regulatory inter-

actions, especially for mammalian systems. The large number of possible interactions are not feasible to test experimentally. Model-based iterative learning can make the generation of relevant experimental datasets more efficient. To this end, we used MERLIN+P+TFA network to both prioritize important regulators, as well as select targets of select regulators to validate the regulatory relationships. Among the top 10% regulators across the MERLIN+P+TFA networks in mESC, the majority (147 out of 205 regulators) of them are already known to be associated with pluripotency, indicating the promise of using MERLIN+P+TFA as a powerful hypothesis generation tool. For each of the remaining regulators (58 of 205 regulators), we tested its association with pluripotency through siRNA knockdown experiments. Many of them were found to have significant positive or negative effect on the pluripotency state of cells (measured by the protein expression of "Nanog" in a cell), thereby establishing them as novel activators or inhibitors of pluripotency. Finally, we experimentally validated the high-confidence target interactions of 4 known and 4 novel regulators, using regulator knock down followed by fold change of expression of the targets and demonstrated the importance of using both mRNA and TFA levels of regulators for obtaining high confidence targets. Taken together, these results show that regularized TFA is beneficial for GRN inference leading to more accurate and comprehensive GRNs across a variety of model systems and for both bulk and single cell RNA-seq datasets and provides a powerful framework for systematic model-based prioritization, validation and discovery of regulatory networks in mammalian systems.

Materials and methods

MERLIN+P+TFA framework for TFA estimation and gene regulatory network inference

MERLIN+P+TFA combines GRN inference using MERLIN+P [7,20] with regularized Transcription Factor Activity (TFA) estimation to allow a regulator to be included based on its mRNA or TFA level. Below we describe the original Network Components Analysis (NCA), followed by the regularized version of NCA for more accurate TFA estimation, and finally, the GRN inference procedure of MERLIN+P+TFA.

Network Component Analysis for TFA estimation

Network component analysis (NCA, [13]) is a method for estimating the unobserved or hidden activity values of TFs given (a) an input graph depicted by an $n \times m$ weighted adjacency matrix, A, where each entry $A(i,j)$ specifies the strength of the connection between gene i and TF j, and (b) the expression levels of the target genes across a set of experiments, depicted by an $n \times s$ matrix E, where each entry $E(i,j)$ specifies the expression of gene i in experiment j. $A(i,j) = 0$ if i is not a target of j, and positive otherwise. NCA assumes that the input network structure constrains the TF action on the targets and TFA value can be inferred back by using an alternating least squares optimization approach. Given the expression matrix E, NCA decomposes the expression matrix as $E = \widehat{A}P$, where P is the $m \times s$ TFA matrix for m TFs across s samples. Each entry $P(j,l)$ specifies the estimated activity of TF j in experiment l. $\widehat{A}$ is the updated adjacency matrix, having the additional constraint that $\widehat{A}$ should match the sparsity pattern of the input network, A. This can be written as the following minimization problem:

$$\min \|E - \widehat{A}P\|_2^2, \quad \textit{s.t.}\ \widehat{A} \text{ matches the sparsity pattern of the input network, } A. \tag{1}$$

NCA solves this minimization problem by iteratively performing a two-step least square regression. We start by randomly initializing $A^{(0)}$ from A preserving the zero-pattern for $k = 0$ iteration, and then at each iteration $k > 0$, it first estimates $P^{(k)}$ by keeping $A^{(k-1)}$ fixed:

$$P^{(k)} = \arg\min_{P^*} \|E - A^{(k-1)}P^*\| \tag{2}$$

The first step can be solved in closed-form using a pseudoinverse formulation $P = (AA^\intercal)^{-1}A^\intercal E$.

Next, NCA estimates $A^{(k)}$ by keeping $P^{(k)}$ fixed:

$$A^{(k)} = \arg\min_{A^*} \|E - A^* P^{(k)}\| \tag{3}$$

The estimation of the second step, the connectivity matrix $A^{(k)}$, can be written as a per-gene regression problem. The ith row of matrix A corresponds to the regression coefficients of all the regulators for target gene i. These regression coefficients are estimated for non-zero entries in the original input network, by minimizing the following mean-squared loss:

$$a^{(k)}_{j,\cdot} = \arg\min_{a^*_{j,\cdot}} \|e_{j,\cdot} - a^*_{i,\cdot} P^{(k)}\| \tag{4}$$

We run NCA multiple times with different random initializations of $A^{(0)}$ to obtain multiple P matrices and take the average of these matrices as the final estimated TFA matrix.

Regularized TFA estimation

To estimate more robust TFAs from a noisy input network, we incorporate regularization within the NCA framework, which is informed by the weights of TF-target edges in the original input network. It is assumed that higher the weight the more confident is the prior support for the TF-target edge. The original NCA algorithm assumes the input graph is binary and as such ignores these edge weights. In order to incorporate our prior knowledge, we use L1-regularized regression scheme. Let $W = (w_{ij})_{n\times m}$ represent the penalties on the edges of the input network (the higher the confidence, the lower the penalty). The L1-regularized regression object is:

$$a^{(k)}_{j,\cdot} = \arg\min_{a^*_{j,\cdot}} \|e_{j,\cdot} - a^*_{j,\cdot} P^{(k)}\| + \lambda \|a^*_{j,\cdot}.w'_{j,\cdot}\| \tag{5}$$

where λ is the regularization hyper-parameter. This objective is solved using is implemented using the "Cyclic Coordinate Descent" algorithm [34].

MERLIN+P+TFA network inference

MERLIN+P+TFA infers a genome-scale gene regulatory network (GRN) by combining regularized TFA estimation with MERLIN+P. This is important because the TFA estimation step only prunes the input network and does not add new edges. MERLIN+P+TFA first applies the regularized NCA to estimate the TFA given a prior network and an expression dataset. Then MERLIN+P is employed to infer the regulatory network given the TFA, prior network, and expression dataset. More specifically, let X_{nca} denote the estimated TFA level for regulator X. We appended these estimated TFA profiles for all regulators for which this was estimated allowing a regulator to participate in the GRN based on its mRNA or estimated TFA level, if available. Thus, if regulator "X" was present in both the expression dataset and the estimated TFA profile, it was assigned two rows in the combined (expression+TFA) matrix, one with "X" (its observed expression levels) and another with "X_nca" (its estimated TF activities).

MERLIN+P+TFA uses a similar stability selection framework as MERLIN+P which can improve network inference results [35]. Briefly, we create n random subsamples of the (expression+TFA) matrix without replacement where each subsample consists of half the number of samples in the (expression+TFA) matrix ($n = 10$ for our simulation studies and 100 for the real data experiments). Subsequently, one regulatory network is inferred from each subsample (with the default setting of MERLIN+P: clustering threshold $\texttt{-h} = 0.6$, sparsity hyper-parameter $\texttt{-s} = -5$, modularity prior $\texttt{-r} = 4$, and prior weight $\texttt{-q} = 5$. Finally, the union of the inferred networks is taken to create a consensus network and each edge is assigned a confidence score which is defined by the proportion of the inferred networks where that edge is present. The edge confidence of each inferred network is estimated by the frequency of observing that edge in the n individual inferred networks. In addition to inferring the regulatory network, MERLIN+P also reports a co-expression/co-regulation cluster assignment for all target genes [20]. To create a consensus clustering or module assignment, we create a co-clustering similarity matrix (where the $(i, j)^{th}$ element indicates how many times genes i and j were in the same cluster among the n individual runs of MERLIN+P), and apply hierarchical clustering using our in-house programs (average linkage and similarity threshold of 0.3). In our downstream analysis, we only considered the modules with 5 or more members.

Simulation experiments

The goal of our simulation experiments was two-fold: (1) to examine how noise in the prior network affects the TFA estimation, and (2) to assess how much improvement in the quality of the estimated TFA can be achieved by incorporating accurate edge weights in the prior network. To generate the input prior network, we randomly selected 100 TFs from the yeast motif prior network (Section) and as target genes, we randomly chose $1,000$ genes from the targets of the selected TFs. Subsequently, NCA+LASSO was applied to the prior network and yeast NatVar expression data to estimate the TFA profile (P) and regression coefficients for the edges of the prior network (A). We generated the simulated expression data (E) using $E = AP$.

Adding noise to the input prior network: We simulate different levels of noise in the prior network by removing $X\%$ of true edges and adding $X\%$ of false edges at random, where $X\% \in \{10\%, 30\%, 50\%\}$, relative to the size of input network (i.e. for a network with $1,000$ edges, 10% noise means removing 100 edges that are present in the network, and adding 100 edges that are not present in the network). For each noise level, we produced 5 replicate networks (5 randomly initialized iterations of removing true edges and adding false edges at random).

Incorporating edge weights in the prior network: Our method is designed based on the premise that incorporating accurate edge weights in the prior network can help estimating more accurate TFA profiles. To test this, we used the following two weighting schemes:

- **Uniform:** All edge weights (for both true and false edges) were set uniformly. To do this, we used beta distribution with $(a = 1, b = 1)$ which is equivalent to uniform distribution over interval $[0, 1]$. In this scenario, edge weights are uniformly distributed between 0 and 1, thus do not reflect the true labels of the edges.

- **Non-uniform:** We used two beta distributions, namely, $(a = 1, b = 5)$ where false edges tend to have higher weights and $(a = 5, b = 1)$ where true edges tend to have higher weights. The second distribution reflects the true labels of the edges. If regularization in NCA is helpful, we should observe improvement when using non-uniform edge weights from the second distribution. True edges are

more likely to get higher edge confidence, and false edges are more likely to get lower edge confidence. To do this, we used beta distribution ($a = 1, b = 5$ for false edges and $a = 5, b = 1$ for true edges. In this scenario edge weights reflects the true label of the edges i.e. the true edges tend to have higher weights than that of the false edges.

If the regularization in NCA is helpful, we should observe improvement when using non-uniform edge weights where true edges have higher confidence. For each noise level, we produced 5 networks (5 rand inits of adding false edges and removing true edges).

Given a specific noise level, for each of the 5 replicates of the prior network, we created three variants using the three different edge-weight distributions. As a result, these variants (uniform and non-uniform) have the same structure but different edge weights. These noisy and edge-weighted prior networks were used for both estimating TFAs and inferring networks.

Estimating TFAs and inferring networks: We created 10 subsamples of the simulated expression data (half the samples in each, without replacement). For each subsample and prior network, we first estimated a TFA profile and then inferred a network using the MERLIN+P method (with its default setting). To estimate TFA, we used $\lambda \in \{0, 0.001, 0.005, 0.010, 0.015, 0.020, 0.05, 0.10, 0.15, 0.20\}$. Here, $\lambda = 0$ corresponds to the original unregularized NCA.

Comparing the estimated TFA to true TFA: For each of our experiments (different noise levels, different edge-weighting options of the prior network, different subsamples of the expression data) we concatenated the estimated TFA profiles of all 100 TFs into a long vector and calculated its absolute Pearson correlation with the true TFA profile. When comparing the performance of one noise scenario to another given a specific edge weighting variant, we used the distribution and median of these correlation values (50 correlation values from 5 replicates of the prior network and 10 expression subsamples, for each noise scenario).

Evaluating the inferred networks: For each noise level and edge weighting scheme of the prior network, MERLIN+P inferred 10 networks corresponding to 10 expression subsamples. We created a consensus network by taking the union of these 10 networks. Each edge was assigned a confidence value based on how many times it had appeared in the inferred networks. We ranked the edges in the consensus network by their

confidence values. This ranked list was inputted to the AUCCalculator Java package [36] along with the gold standard network to calculate the area under the precision-recall curve (AUPR). For each noise level, we estimated 5 AUPR values corresponding to the 5 replicates of the prior network.

MERLIN+P+TFA for GRN inference from bulk gene expression from yeast and mammals

Real datasets

For real data experiments, we used various datasets of yeast and mammalian cell lines (mouse and human). The datasets and experimental designs are described below.

Yeast expression dataset. To create the expression matrix of yeast, three natural variation expression datasets (NatVar) were combined from Brem et al. [37], Smith et al. [38], and Zhu et al. [39], as described in Siahpirani et al. [7].

Mouse embryonic stem cell (mESC) expression datasets. To create the expression matrix, we used publicly available datasets from the gene expression omnibus (GEO) [40] database. We searched GEO for expression datasets (both microarray and RNA-seq) that are measured in mESC or induced pluripotent stem cells (iPSC). Consequently, We collected $3,504$ microarray samples from 194 datasets, and $2,182$ RNA-seq samples from 216 datasets. For the microarrays, when the raw data was available, the expression was extracted using appropriate R [41] packages depending on the type of the array. On the other hand, when the raw data was not available for the microarrays, we used the processed expression values uploaded to GEO. The RNA-seq data were processed using RSEM [42].

Human expression datasets. For human cell lines, we searched MetaSRA [43] (which can search the meta data of the sequence read archive (SRA) [44] database for human samples) and GEO for relevant samples. We downloaded $1,198$ samples from 130 datasets for the MCF7 cell line, $1,501$ samples from 43 datasets for the LCL cell line, and $2,195$ samples from 131 datasets for the human ESC (hESC)/iPSC cell line. Since all of them are RNA-seq samples, they were processed with RSEM.

For each of the aforementioned datasets (yeast and mammalian), we collapsed the replicates to mean, and applied quantile normalization (with `quantilenorm()` in MATLAB [45]) to the merged datasets. To

remove batch effects, each gene's expression was zero-meaned per dataset. Moreover, in each dataset, we retained a gene only if its expressions are at least $(2 \times \text{standard deviation})$ higher than the mean in no less than 6% of the samples. **Table 1** summarizes the number of samples and datasets for each cell line.

Prior networks.

The motif prior network in yeast was obtained from Siahpirani et al. [7]. Briefly, we used position-weight matrices from Gordân et al. [46] and YeTFaSCo [47], and scanned yeast promoters (defined as $1,000$ bp upstream of the first ATG of a gene) using the TestMotif program [48]. The edges were weighted by percentile ranking (defined by motif p-value, with highest ranks assigned to lowest p-values). To create the prior network for mESC, we used motif instances from the CIS-BP database [49] and the DNase I data of mouse ESC from the ENCODE database [50]. Subsequently, PIQ [51] was applied to find motif instances and score them by their DNase footprint. We mapped motif instances to $\pm 10Kbp$ of the transcription start site (TSS), and selected the motif instances with purity score ≥ 0.85. Finally, the edges were weighted by their percentile ranks based on their motif p-values. For the human cell lines, we used corresponding DNase I data from ENCODE, and in addition to CIS-BP motifs, we also used ENCODE and JASPAR motif instances. For the human cell lines, we used DNase I data from ENCODE and the CIS-BP motif instances as well as the ENCODE [52] and JASPAR [53] motif instances. **Table 2** describes the details of these prior networks.

Gold standard networks.

For the yeast experiments, we used 5 different gold standard networks: one based on ChIP-chip experiments from MacIsaac et al. [54], two from the YEASTRACT database [55] (*type≥2* consists of edges found in two or more different types of experimental assays, and *count ≥3*, consists of edges that were detected in three or more experiments), and two KO based networks (one from Hu et al. [56] and the other from Kemmeren et al. [57]).

For mESC, we used ChIP-based and LOGOF (loss or gain of function) based gold standard networks from the ESCAPE database [58] (**Supp file 11**). Moreover, we retrieved the significant knockdown interactions identified in [59–61] and added them to the LOGOF network (**Supp file 12**). As another gold standard,

we formed a network with the edges that are common between the ChIP and LOGOF gold standard networks. Furthermore, a small gold standard network was created from the regulatory interactions reported in the literature [62–68].

For hESC, we transformed mESC gold standard networks ChIP, LOGOF, Literature, to hESC gold standard networks ChIP, KD, Literature, respectively, by converting the mouse gene names to human gene names. Additionally, we used the ChIP-seq peaks available in ENCODE for the ESC cell line. For each TF, the peaks were mapped to $\pm 10Kbp$ of TSS of human genes and the resultant TF-target interactions were added to the hESC ChIP gold standard network. For LCL, we used the ChIP and KD interactions reported in Cusanovich et al. [69], and, as an additional gold standard, we used the intersection of the ChIP and KD networks. For MCF7, we obtained the corresponding ChIP-seq peaks from ENCODE and processed them using the same pipeline as that of hESC.

For each gold standard, we only used the edges where both regulator and target have expression in the given dataset (and if a regulator only had TFA we did not use that edge). **Table 3** summarizes the details of these networks.

TFA estimation and network inference.

For each expression dataset and the corresponding prior network, we used NCA+LASSO to estimate the TFA from all the samples. For the yeast experiments, we used 10 random initializations of $A^{(0)}$ (i.e. we estimated TFA 10 times and took the average) along with $\lambda \in \{0.000, 0.005, 0.020, 0.100\}$ for regularization (where $\lambda = 0$ corresponds to no regularization). For the mammalian experiments, we used 100 random initializations of $A^{(0)}$ along with $\lambda \in \{0.000, 0.005, 0.010, 0.020, 0.100\}$ for regularization. For the network inference in yeast, we employed MERLIN+P [7], MERLIN [20] (MERLIN+P without prior), GENIE3 [16], Inferelator [6,17], TIGRESS [18], NetREX [19], and mLASSO-StARS [12]. For the mammalian network inference, only MERLIN and MERLIN+P were employed. For GENIE3, we used $K = \texttt{sqrt}$ and $\texttt{nb_trees} = 1000$. For Inferelator, modified elastic net (MEN) was used with 20 bootstraps where RNA degradation rate (τ) was set to 20.1 and prior weight was set to 0.01 (as done in Siahpirani et al. [7]). MERLIN and MERLIN+P were ran as described in Section . For all inference methods, we used both expression and TFA of regulators when available.

Additional evaluation metrics for real expression datasets

AUPR and predictable TFs To evaluate the inferred networks, we used two evaluation metrics: 1) area under the precision-recall curve (AUPR) and 2) the number of predictable TFs. We used the complete network for AUPR calculation whereas only the top $30K$ of inferred edges were used for counting the number of predictable TFs. Our previous experiments has shown that changing the number of edges (from $10K$ to $50K$) does not significantly change the results [7]. We calculated each AUPR score by comparing an inferred network to a given gold standard network as described in the previous sections.

Regulator ranking. To assess the importance of a regulator in an inferred network, we ~~ask~~ determined how much the predictive power of the network decreases in predicting the expression levels of the genes if we remove that ~~TF~~ regulator. For that purpose, we followed the strategy described in Siahpirani et al. [7]. Briefly, for each target gene, a linear regression model was trained (with 5-fold cross validation) to predict the expression levels of the target gene as a function of that of its regulators (**Figure 7**). Subsequently, for each of the regulators of that gene, we removed the regulator and calculated the change in the prediction error. The importance of a regulator was then calculated as the sum of the changes in the prediction errors for all of its targets. To create the list of regulators associated with the ESC state, we used the genes assigned to GO terms "regulation of embryonic development" and "embryonic organ development" in addition to the genes reported by Müller et al. [70] and Wong et al. [71] on MSigDB [72, 73].

Experimental validation of MERLIN+P+TFA mESC GRN

mESC cell culture E14 feeder independent mESC and 2D4 feeder dependent iPSC cell lines were maintained in mouse ESC media containing serum.

siRNA transfection Cell lines were plated at a density of 100,000 cells per well in 12-well plates. Reverse transfection was performed with 20 nM of siRNA (purchased from IDT) using DharmaFECT-1 according to the manufacturer's protocol. Each transcription factor was targeted with a 4 siRNApool (see Supp file 14) in quadruplicate. Two rounds of forward transfection were performed at 36 hours with 25 nM siRNA and 60 hours with 40 nM of siRNA to accommodate increased cell numbers from cell division. siRNAs targeting

Pou5f1 and Nanog were used as positive controls for pluripotency depletion and non-targeting (NT) siRNA sequence (UAGCGACUAAACACAUCAA) and mock transfected without any siRNA were used as negative controls. Cells were harvested 72 hours after initial transfection to assess knockdown efficiency.

RNA extraction, cDNA synthesis, and RT-qPCR Total RNA was isolated using the ISOLATE II RNA Mini Kit following the manufacturer's protocol. For each of the tested TFs, positive and negative controls three technical replicates were obtined. 1 ug of RNA was converted to cDNA using the qScript cDNA Synthesis Kit. Real time quantitative PCR qPCR was for Pou5f1 and Nanog was performed to assess pluripotency with Gapdh and RNA Polymerase II serving as housekeeping controls (see **Supp file 15** for primer details).

Flow cytometry To detect pluripotency protein expression, cells permeabilized using the BD Cytofix/Cytoperm Kit (BD Biosciences) according to manufacturer instructions. Briefly, cells were incubated in BD Cytofix/Cytoperm Reagent for 15-30 minutes at room temperature, washed with BD Perm/Wash buffer, and further treated with BD Permeabilization Plus buffer on ice for 10 minutes. After an additional wash, intracellular staining was performed using the BD Perm/Wash buffer containing anti-mouse Nanog-eFluor 660 antibodies for 1 hour at room temperature. Cells were washed, pelleted, and resuspended in BD Perm/Wash buffer for analysis. Samples were quantified on a Flow Cytometer and analyzed using FlowJo Software.

Target validation of novel TFs To further validate TFA prediction accuracy and measure the effect of the novel TFs on pluripotency, we selected a subset of the novel TFs tested with siRNA knockdown to assess the effect of the siRNA on the predicted target genes of the novel TFs. Specifically, we investigated four novel TFs not present in the mESC gold standard networks (Fosl1, Gtf2a2, Zbtb7b, Lztr1), along with four of the well-characterized TFs (Esrrb, Nanog, Oct4, Zfp281) serving as positive controls. Cell culture and siRNA transfection were conducted in the E14 cell line with each condition performed in duplicate to obtain a total of three biological replicates for both novel and known TFs. Following RNA extraction and cDNA synthesis steps as described for earlier siRNA sample processing, qPCR was performed in duplicate for the six tested TFs with primers designed to measure expression of targets for the TF undergoing siRNA knockdown (see **Supp file 15** for primer details).

MERLIN+P+TFA for GRN inference from scRNAseq datasets

We applied MERLIN+P+TFA, Inferelator with prior (Inferelator+P) and SCENIC network inference algorithms on scRNA-seq datasets from our previous benchmarking study of scRNA-seq GRN inference algorithms by McCalla et al. [28].We selected these three algorithms for further investigation as MERLIN and Inferelator can utilize prior network inputs, while SCENIC was highly ranked with reasonable computational runtimes in a previous scRNAseq network inference benchmarking study [28]. For regularized TFA, we used $\lambda \in \{0.005, 0.010, 0.020, 0.100\}$. We evaluated the performance of regularized TFA for scRNAseq datasets when applied with the MERLIN [20] [7], Inferelator [29], and SCENIC [30] network inference algorithms. We utilized the previously inferred scRNAseq networks from McCalla et al. [28] that compared networks inferred from expression alone, the addition of priors, TFA generated using NCA (unregularized TFA), and Inferelator's in-algorithm estimated TFA.

The datasets we considered were from three different organisms: human dendritic cells (Shalek [74]), human pluripotent stem cells (Han), yeast (Gasch [31] and Jackson [29]), and two mouse reprogramming datasets (Sridharan FBS [75], Sridharan A2S [75]). Preprocessed datasets and input priors were prepared as detailed in McCalla et. al [28]. The inferred GRNs were evaluated using F-score, AUPR, and predictable TFs similar to our bulk RNA-seq experiments using the gold standard networks Perturb, ChIP, and the intersection of Perturb and ChIP as described in McCalla et al. [28]. For the output of each algorithm that utilized regularized TFA, we selected the one best inferred network out of the four regularized TFA networks that returned the best AUPR result when testing regularized TFA settings of $\lambda \in \{0.005, 0.010, 0.020, 0.100\}$. This "best regularized TFA" network was used for network metric calculations for both AUPR and predictable TFs.

Acknowledgments

We thank the Center for High Throughput Computing (CHTC) at UW-Madison for computational resources. We thank Khoa Tran for providing the list of ESC associated genes and help in reviewing mouse ESC samples. We thank the members of the Roy and Sridharan groups for helpful discussions. This work was supported by NSF grants IOS 1840687 and IOS 24066533, NIH grants R01GM144708 and R01GM117339, and DOE grant DE-SC0021052 to S.R., R01HD105151 to R.S. and a UW-Madison Stem Cell and Regenerative Medicine Center predoctoral fellowship to C.M.D.

References

[1] Guido Sanguinetti and Vân Anh Huynh-Thu, editors. *Gene Regulatory Networks: Methods and Protocols*, volume 1883 of *Methods in Molecular Biology*. Springer New York, New York, NY, 2019.

[2] D. Husmeier and A. V. Werhli. Bayesian integration of biological prior knowledge into the reconstruction of gene regulatory networks with bayesian networks. *Comput. Syst. Bioinf., CSB2007 Conf. Proc., 6th*, 6:85–95, 2007.

[3] Adriano V. Werhli and Dirk Husmeier. Reconstructing gene regulatory networks with bayesian networks by combining expression data with multiple sources of prior knowledge. *Stat. Appl. Genet. Mol. Biol.*, 6(1), January 2007.

[4] Sach Mukherjee and Terence P. Speed. Network inference using informative priors. *Proc. Natl. Acad. Sci. U. S. A.*, 105(38):14313–14318, September 2008.

[5] Steven M. Hill, Yiling Lu, Jennifer Molina, Laura M. Heiser, Paul T. Spellman, Terence P. Speed, Joe W. Gray, Gordon B. Mills, and Sach Mukherjee. Bayesian inference of signaling network topology in a cancer cell line. *Bioinformatics*, 28(21):2804–2810, November 2012.

[6] Alex Greenfield, Christoph Hafemeister, and Richard Bonneau. Robust data-driven incorporation of prior knowledge into the inference of dynamic regulatory networks. *Bioinformatics*, 29(8):1060–1067, April 2013.

[7] Alireza F Siahpirani and Sushmita Roy. A prior-based integrative framework for functional transcriptional regulatory network inference. *Nucleic acids research*, 45:e21, February 2017.

[8] Christopher Koch, Jay Konieczka, Toni Delorey, Ana Lyons, Amanda Socha, Kathleen Davis, Sara A. Knaack, Dawn Thompson, Erin K. O'Shea, Aviv Regev, and Sushmita Roy. Inference and evolutionary analysis of genome-scale regulatory networks in large phylogenies. *Cell Systems*, 4(5):543 – 558.e8, 2017.

[9] Mario L Arrieta-Ortiz, Christoph Hafemeister, Ashley Rose Bate, Timothy Chu, Alex Greenfield, Bentley Shuster, Samantha N Barry, Matthew Gallitto, Brian Liu, Thadeous Kacmarczyk, Francis Santoriello, Jie Chen, Christopher D A Rodrigues, Tsutomu Sato, David Z Rudner, Adam Driks, Richard Bonneau, and Patrick Eichenberger. An experimentally supported model of the bacillus subtilis global transcriptional regulatory network. *Molecular systems biology*, 11:839, November 2015.

[10] Olivia Wilkins, Christoph Hafemeister, Anne Plessis, Meisha-Marika Holloway-Phillips, Gina M. Pham, Adrienne B. Nicotra, Glenn B. Gregorio, S.V. Krishna Jagadish, Endang M. Septiningsih, Richard Bonneau, and Michael Purugganan. Egrins (environmental gene regulatory influence networks) in rice that function in the response to water deficit, high temperature, and agricultural environments. *The Plant Cell*, 28(10):2365–2384, 2016.

[11] Dayanne M. Castro, Nicholas R. de Veaux, Emily R. Miraldi, and Richard Bonneau. Multi-study inference of regulatory networks for more accurate models of gene regulation. *PLOS Computational Biology*, 15(1):1–22, 01 2019.

[12] Emily R. Miraldi, Maria Pokrovskii, Aaron Waters, Dayanne M. Castro, Nick De Veaux, Jason Hall, June-Yong Lee, Maria Ciofani, Aviv Madar, Nick Carriero, and et al. Leveraging chromatin accessibility for transcriptional regulatory network inference in t helper 17 cells. *Genome Research*, page gr.238253.118, Jan 2019.

[13] James C. Liao, Riccardo Boscolo, Young-Lyeol Yang, Linh M. Tran, Chiara Sabatti, and Vwani P. Roychowdhury. Network component analysis: Reconstruction of regulatory signals in biological systems. *Proc. Natl. Acad. Sci. U. S. A.*, 100(26):15522–15527, December 2003.

[14] David Heckerman, David M. Chickering, Christopher Meek, Robert Rounthwaite, and Carl Kadie. Dependency networks for inference, collaborative filtering, and data visualization. *J. Mach. Learn. Res.*, 1:49–75, 2001.

[15] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[16] Vân A. Huynh-Thu, Alexandre Irrthum, Louis Wehenkel, and Pierre Geurts. Inferring regulatory networks from expression data using Tree-Based methods. *PLoS One*, 5(9):e12776+, September 2010.

[17] Richard Bonneau, David Reiss, Paul Shannon, Marc Facciotti, Leroy Hood, Nitin Baliga, and Vesteinn Thorsson. The inferelator: an algorithm for learning parsimonious regulatory networks from systems-biology data sets de novo. *Genome Biol.*, 7(5):R36+, 2006.

[18] Anne-Claire C. Haury, Fantine Mordelet, Paola Vera-Licona, and Jean-Philippe P. Vert. TIGRESS: Trustful inference of gene REgulation using stability selection. *BMC Syst. Biol.*, 6(1):145+, 2012.

[19] Yijie Wang, Dong-Yeon Cho, Hangnoh Lee, Justin Fear, Brian Oliver, and Teresa M. Przytycka. Reprogramming of regulatory network using expression uncovers sex-specific gene regulation in drosophila. *Nature Communications*, 9(1):4061, 2018.

[20] Sushmita Roy, Stephen Lagree, Zhonggang Hou, James A. Thomson, Ron Stewart, and Audrey P. Gasch. Integrated module and Gene-Specific regulatory inference implicates upstream signaling networks. *PLoS Comput. Biol.*, 9(10):e1003252+, October 2013.

[21] David J. Wong, Helen Liu, Todd W. Ridky, David Cassarino, Eran Segal, and Howard Y. Chang. Module Map of Stem Cell Genes Guides Creation of Epithelial Cancer Stem Cells. *Cell Stem Cell*, 2(4):333–344, April 2008.

[22] Franz-Josef Müller, Louise C. Laurent, Dennis Kostka, Igor Ulitsky, Roy Williams, Christina Lu, In-Hyun Park, Mahendra S. Rao, Ron Shamir, Philip H. Schwartz, Nils O. Schmidt, and Jeanne F. Loring. Regulatory networks define phenotypic classes of human stem cell lines. *Nature*, 455(7211):401–405, September 2008.

[23] Arthur Liberzon, Chet Birger, Helga Thorvaldsdóttir, Mahmoud Ghandi, Jill P. Mesirov, and Pablo Tamayo. The Molecular Signatures Database Hallmark Gene Set Collection. *Cell Systems*, 1(6):417–425, December 2015.

[24] Aravind Subramanian, Pablo Tamayo, Vamsi K. Mootha, Sayan Mukherjee, Benjamin L. Ebert, Michael A. Gillette, Amanda Paulovich, Scott L. Pomeroy, Todd R. Golub, Eric S. Lander, and Jill P. Mesirov. Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences*, 102(43):15545–15550, October 2005.

[25] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet*, 25(1):25–29, May 2000.

[26] Argenis Arriojas, Susan Patalano, Jill Macoska, and Kourosh Zarringhalam. A bayesian noisy logic model for inference of transcription factor activity from single cell and bulk transcriptomic data. *NAR Genomics and Bioinformatics*, 5(4):lqad106, 2023.

[27] Tomasz Wlodarczyk, Aaron Lun, Diana Wu, Shreya Menon, Shushan Toneyan, Kerstin Seidel, Liang Wang, Jenille Tan, Shang-Yang Chen, Timothy Keyes, et al. Epiregulon: Inference of single-cell transcription factor activity to dissect mechanisms of lineage plasticity and drug response. *bioRxiv*, pages 2023–11, 2023.

[28] Sunnie Grace McCalla, Alireza Fotuhi Siahpirani, Jiaxin Li, Saptarshi Pyne, Matthew Stone, Viswesh Periyasamy, Junha Shin, and Sushmita Roy. Identifying strengths and weaknesses of methods for computational network inference from single-cell RNA-seq data. *G3 Genes—Genomes—Genetics*, 13(3):jkad004, 01 2023.

[29] Christopher A Jackson, Dayanne M Castro, Giuseppe-Antonio Saldi, Richard Bonneau, and David Gresham. Gene regulatory network reconstruction using single-cell rna sequencing of barcoded genotypes in diverse environments. *elife*, 9:e51254, 2020.

[30] Sara Aibar, Carmen Bravo González-Blas, Thomas Moerman, Vân Anh Huynh-Thu, Hana Imrichova, Gert Hulselmans, Florian Rambow, Jean-Christophe Marine, Pierre Geurts, Jan Aerts, et al. Scenic: single-cell regulatory network inference and clustering. *Nature methods*, 14(11):1083–1086, 2017.

[31] Audrey P Gasch, Feiqiao Brian Yu, James Hose, Leah E Escalante, Mike Place, Rhonda Bacher, Jad Kanbar, Doina Ciobanu, Laura Sandor, Igor V Grigoriev, et al. Single-cell rna sequencing reveals intrinsic and extrinsic regulatory heterogeneity in yeast responding to stress. *PLoS biology*, 15(12):e2004050, 2017.

[32] Deborah Chasman, Alireza Fotuhi Siahpirani, and Sushmita Roy. Network-based approaches for analysis of complex biological systems. *Current Opinion in Biotechnology*, 39:157–166, June 2016.

[33] Deborah Chasman and Sushmita Roy. Inference of cell type specific regulatory networks on mammalian lineages. *Current Opinion in Systems Biology*, 2:130 – 139, 2017. Regulatory and metabolic networks • Cancer and systemic diseases.

[34] Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical Learning with Sparsity: The Lasso and Generalizations*. Chapman & Hall/CRC, 2015.

[35] Nicolai Meinshausen and Peter Buehlmann. Stability selection, May 2009.

[36] Jesse Davis and Mark Goadrich. The relationship between Precision-Recall and ROC curves. In *Proceedings of the 23rd International Conference on Machine Learning (ICML 2006)*, ICML '06, pages 233–240, New York, NY, USA, 2006. ACM.

[37] Rachel B. Brem and Leonid Kruglyak. The landscape of genetic complexity across 5,700 gene expression traits in yeast. *Proc. Natl. Acad. Sci. U. S. A.*, 102(5):1572–1577, February 2005.

[38] Erin N. Smith and Leonid Kruglyak. Gene–Environment interaction in yeast gene expression. *PLoS Biol.*, 6(4):e83+, April 2008.

[39] Jun Zhu, Pavel Sova, Qiuwei Xu, Kenneth M. Dombek, Ethan Y. Xu, Heather Vu, Zhidong Tu, Rachel B. Brem, Roger E. Bumgarner, and Eric E. Schadt. Stitching together multiple data dimensions

reveals interacting metabolomic and transcriptomic networks that modulate cell regulation. *PLoS Biol.*, 10(4):e1001301+, April 2012.

[40] Tanya Barrett, Stephen E. Wilhite, Pierre Ledoux, Carlos Evangelista, Irene F. Kim, Maxim Tomashevsky, Kimberly A. Marshall, Katherine H. Phillippy, Patti M. Sherman, Michelle Holko, Andrey Yefanov, Hyeseung Lee, Naigong Zhang, Cynthia L. Robertson, Nadezhda Serova, Sean Davis, and Alexandra Soboleva. NCBI GEO: archive for functional genomics data sets—update. *Nucleic Acids Research*, 41(D1):D991–D995, 11 2012.

[41] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2020.

[42] Bo Li and Colin N Dewey. Rsem: accurate transcript quantification from rna-seq data with or without a reference genome. *BMC bioinformatics*, 12:323, August 2011.

[43] Matthew N Bernstein, AnHai Doan, and Colin N Dewey. MetaSRA: normalized human sample-specific metadata for the Sequence Read Archive. *Bioinformatics*, 33(18):2914–2923, 05 2017.

[44] Rasko Leinonen, Hideaki Sugawara, Martin Shumway, and International Nucleotide Sequence Database Collaboration. The sequence read archive. *Nucleic acids research*, 39(suppl_1):D19–D21, 2010.

[45] MATLAB. *version 9.11.0.1837725 (R2021b) Update 2*. The MathWorks Inc., Natick, Massachusetts, 2021.

[46] Raluca Gordân, Kevin F. Murphy, Rachel P. McCord, Cong Zhu, Anastasia Vedenko, and Martha L. Bulyk. Curated collection of yeast transcription factor DNA binding specificity data reveals novel structural and gene regulatory insights. *Genome Biol.*, 12(12):R125+, December 2011.

[47] Carl G. de Boer and Timothy R. Hughes. YeTFaSCo: a database of evaluated yeast transcription factor sequence specificities. *Nucleic Acids Res.*, 40(D1):D169–D179, January 2012.

[48] Y. Barash, G. Elidan, T. Kaplan, and N. Friedman. CIS: compound importance sampling method for protein–DNA binding site p-value estimation. *Bioinformatics*, 21(5):596–600, March 2005.

[49] Matthew T. Weirauch, Ally Yang, Mihai Albu, Atina G. Cote, Alejandro Montenegro-Montero, Philipp Drewe, Hamed S. Najafabadi, Samuel A. Lambert, Ishminder Mann, Kate Cook, Hong Zheng, Alejandra Goity, Harm van Bakel, Jean-Claude Lozano, Mary Galli, Mathew G. Lewsey, Eryong Huang, Tuhin Mukherjee, Xiaoting Chen, John S. Reece-Hoyes, Sridhar Govindarajan, Gad Shaulsky, Albertha J M. Walhout, François-Yves Bouget, Gunnar Ratsch, Luis F. Larrondo, Joseph R. Ecker, and Timothy R. Hughes. Determination and inference of eukaryotic transcription factor sequence specificity. *Cell*, 158(6):1431–1443, Sep 2014.

[50] ENCODE Project Consortium. An integrated encyclopedia of dna elements in the human genome. *Nature*, 489(7414):57–74, Sep 2012.

[51] Richard I. Sherwood, Tatsunori Hashimoto, Charles W. O'Donnell, Sophia Lewis, Amira A. Barkal, John Peter van Hoff, Vivek Karun, Tommi Jaakkola, and David K. Gifford. Discovery of directional and nondirectional pioneer transcription factors by modeling dnase profile magnitude and shape. *Nat Biotechnol*, 32(2):171–178, Feb 2014.

[52] Pouya Kheradpour and Manolis Kellis. Systematic discovery and characterization of regulatory motifs in ENCODE TF binding experiments. *Nucleic Acids Research*, 42(5):2976–2987, 12 2013.

[53] Aziz Khan, Oriol Fornes, Arnaud Stigliani, Marius Gheorghe, Jaime A Castro-Mondragon, Robin van der Lee, Adrien Bessy, Jeanne Chèneby, Shubhada R Kulkarni, Ge Tan, Damir Baranasic, David J Arenillas, Albin Sandelin, Klaas Vandepoele, Boris Lenhard, Benoît Ballester, Wyeth W Wasserman, François Parcy, and Anthony Mathelier. JASPAR 2018: update of the open-access database of transcription factor binding profiles and its web framework. *Nucleic Acids Research*, 46(D1):D260–D266, 11 2017.

[54] Kenzie MacIsaac, Ting Wang, D. Benjamin Gordon, David Gifford, Gary Stormo, and Ernest Fraenkel. An improved map of conserved regulatory sites for Saccharomyces cerevisiae. *BMC Bioinf.*, 7(1):113+, March 2006.

[55] Miguel C. Teixeira, Pedro Monteiro, Pooja Jain, Sandra Tenreiro, Alexandra R. Fernandes, Nuno P. Mira, Marta Alenquer, Ana T. Freitas, Arlindo L. Oliveira, and Isabel Sá-Correia. The YEASTRACT

database: a tool for the analysis of transcription regulatory associations in saccharomyces cerevisiae. *Nucleic Acids Res.*, 34(suppl 1):D446–D451, January 2006.

[56] Zhanzhi Hu, Patrick J. Killion, and Vishwanath R. Iyer. Genetic reconstruction of a functional transcriptional regulatory network. *Nat. Genet.*, 39(5):683–687, May 2007.

[57] Patrick Kemmeren, Katrin Sameith, Loes A. van de Pasch, Joris J. Benschop, Tineke L. Lenstra, Thanasis Margaritis, Eoghan O'Duibhir, Eva Apweiler, Sake van Wageningen, Cheuk W. Ko, and et al. Large-scale genetic perturbations reveal regulatory networks and an abundance of gene-specific repressors. *Cell*, 157(3):740–752, April 2014.

[58] Huilei Xu, Caroline Baroukh, Ruth Dannenfelser, Edward Y Chen, Christopher M Tan, Yan Kou, Yujin E Kim, Ihor R Lemischka, and Avi Ma'ayan. Escape: database for integrating high-content published data collected from human and mouse embryonic stem cells. *Database (Oxford)*, 2013:bat045, 2013.

[59] Akira Nishiyama, Li Xin, Alexei A. Sharov, Marshall Thomas, Gregory Mowrer, Emily Meyers, Yulan Piao, Samir Mehta, Sarah Yee, Yuhki Nakatake, Carole Stagg, Lioudmila Sharova, Lina S. Correa-Cerro, Uwem Bassey, Hien Hoang, Eugene Kim, Richard Tapnio, Yong Qian, Dawood Dudekula, Michal Zalzman, Manxiang Li, Geppino Falco, Hsih-Te Yang, Sung-Lim Lee, Manuela Monti, Ilaria Stanghellini, Md. Nurul Islam, Ramaiah Nagaraja, Ilya Goldberg, Weidong Wang, Dan L. Longo, David Schlessinger, and Minoru S.H. Ko. Uncovering early response of gene regulatory networks in escs by systematic induction of transcription factors. *Cell Stem Cell*, 5(4):420 – 433, 2009.

[60] Lina S. Correa-Cerro, Yulan Piao, Alexei A. Sharov, Akira Nishiyama, Jean S. Cadet, Hong Yu, Lioudmila V. Sharova, Li Xin, Hien G. Hoang, Marshall Thomas, Yong Qian, Dawood B. Dudekula, Emily Meyers, Bernard Y. Binder, Gregory Mowrer, Uwem Bassey, Dan L. Longo, David Schlessinger, and Minoru S. H. Ko. Generation of mouse es cell lines engineered for the forced induction of transcription factors. *Scientific Reports*, 1:167 EP –, Nov 2011. Article.

[61] Akira Nishiyama, Alexei A. Sharov, Yulan Piao, Misa Amano, Tomokazu Amano, Hien G. Hoang, Bernard Y. Binder, Richard Tapnio, Uwem Bassey, Justin N. Malinou, Lina S. Correa-Cerro, Hong

Yu, Li Xin, Emily Meyers, Michal Zalzman, Yuhki Nakatake, Carole Stagg, Lioudmila Sharova, Yong Qian, Dawood Dudekula, Sarah Sheer, Jean S. Cadet, Tetsuya Hirata, Hsih-Te Yang, Ilya Goldberg, Michele K. Evans, Dan L. Longo, David Schlessinger, and Minoru S. H. Ko. Systematic repression of transcription factors reveals limited patterns of gene expression changes in es cells. *Scientific Reports*, 3:1390 EP –, Mar 2013. Article.

[62] Qing Zhou, Hiram Chipperfield, Douglas A. Melton, and Wing Hung Wong. A gene regulatory network in mouse embryonic stem cells. *Proceedings of the National Academy of Sciences*, 104(42):16438–16443, 2007.

[63] Jonghwan Kim, Jianlin Chu, Xiaohua Shen, Jianlong Wang, and Stuart H. Orkin. An extended transcriptional network for pluripotency of embryonic stem cells. *Cell*, 132(6):1049–1061, March 2008.

[64] Richard A. Young. Control of the embryonic stem cell state. *Cell*, 144(6):940 – 954, 2011.

[65] Yosef Buganim, Dina A. Faddah, Albert W. Cheng, Elena Itskovich, Styliani Markoulaki, Kibibi Ganz, Sandy L. Klemm, Alexander van Oudenaarden, and Rudolf Jaenisch. Single-cell expression analyses during cellular reprogramming reveal an early stochastic and a late hierarchic phase. *Cell*, 150(6):1209–1222, September 2012.

[66] S. J. Dunn, G. Martello, B. Yordanov, S. Emmott, and A. G. Smith. Defining an essential transcription factor program for naïve pluripotency. *Science*, 344(6188):1156–1160, June 2014.

[67] Huilei Xu, Yen-Sin Ang, Ana Sevilla, Ihor R. Lemischka, and Avi Ma'ayan. Construction and validation of a regulatory network for pluripotency and Self-Renewal of mouse embryonic stem cells. *PLoS Comput Biol*, 10(8):e1003777+, August 2014.

[68] Mohan Malleshaiah, Megha Padi, Pau Rué, John Quackenbush, Alfonso Martinez-Arias, and Jeremy Gunawardena. Nac1 coordinates a sub-network of pluripotency factors to regulate embryonic stem cell differentiation. *Cell Reports*, 14(5):1181–1194, January 2016.

[69] Darren A. Cusanovich, Bryan Pavlovic, Jonathan K. Pritchard, and Yoav Gilad. The functional consequences of variation in transcription factor binding. *PLoS Genet*, 10(3):e1004226+, March 2014.

[70] Franz-Josef Müller, Louise C. Laurent, Dennis Kostka, Igor Ulitsky, Roy Williams, Christina Lu, In-Hyun Park, Mahendra S. Rao, Ron Shamir, Philip H. Schwartz, Nils O. Schmidt, and Jeanne F. Loring. Regulatory networks define phenotypic classes of human stem cell lines. *Nature*, 455:401 EP –, Aug 2008.

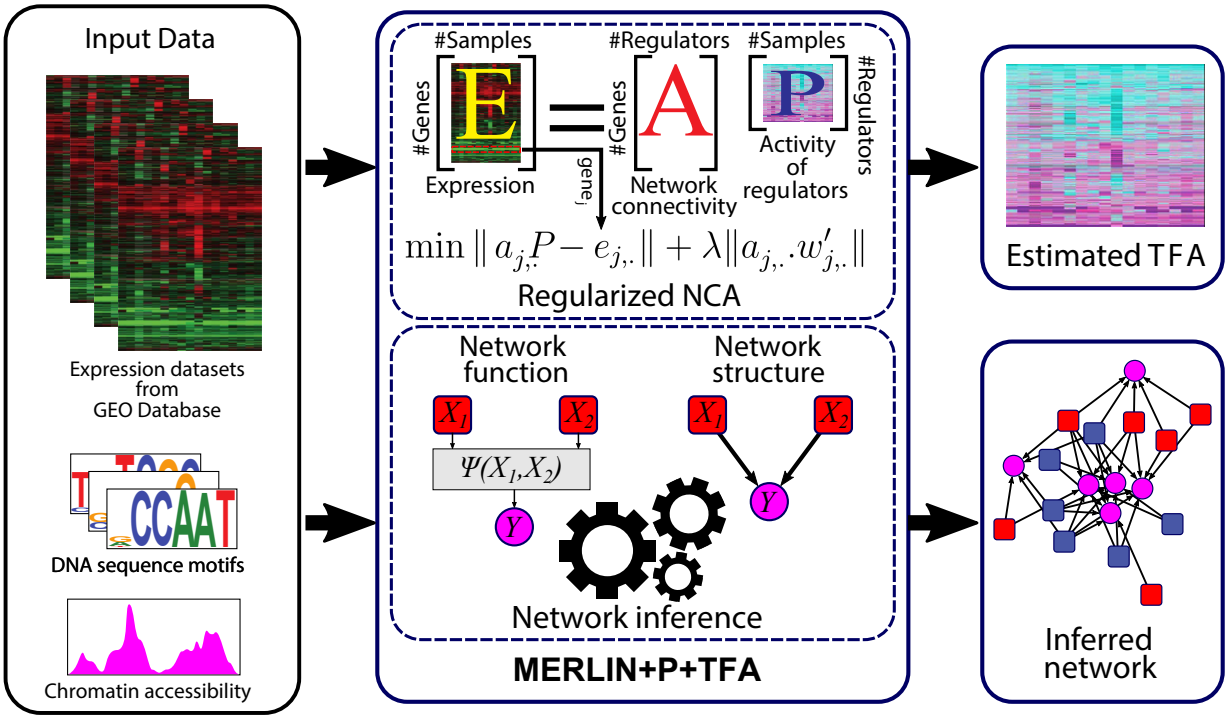
[71] David J. Wong, Helen Liu, Todd W. Ridky, David Cassarino, Eran Segal, and Howard Y. Chang. Module map of stem cell genes guides creation of epithelial cancer stem cells. *Cell Stem Cell*, 2(4):333 – 344, 2008.

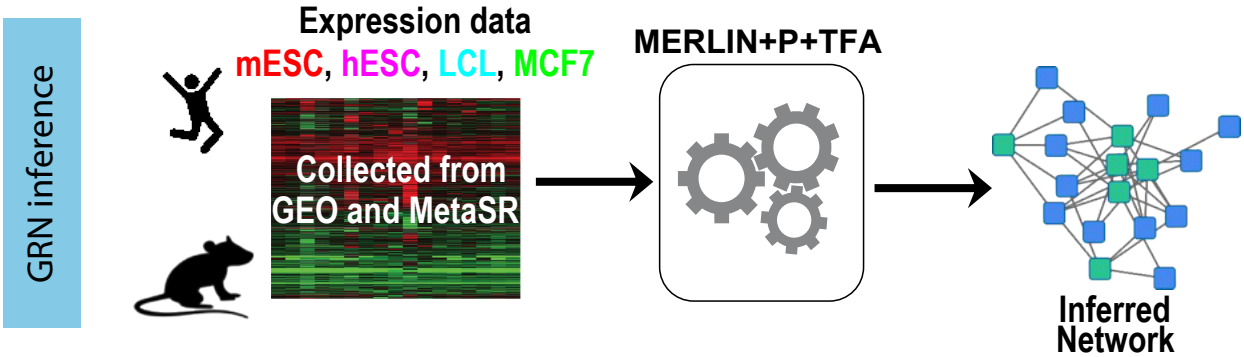
[72] Aravind Subramanian, Pablo Tamayo, Vamsi K. Mootha, Sayan Mukherjee, Benjamin L. Ebert, Michael A. Gillette, Amanda Paulovich, Scott L. Pomeroy, Todd R. Golub, Eric S. Lander, and Jill P. Mesirov. Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences*, 102(43):15545–15550, 2005.

[73] Arthur Liberzon, Chet Birger, Helga Thorvaldsdóttir, Mahmoud Ghandi, Jill P. Mesirov, and Pablo Tamayo. The molecular signatures database hallmark gene set collection. *Cell Systems*, 1(6):417 – 425, 2015.

[74] Alex K Shalek, Rahul Satija, Joe Shuga, John J Trombetta, Dave Gennert, Diana Lu, Peilin Chen, Rona S Gertner, Jellert T Gaublomme, Nir Yosef, et al. Single-cell rna-seq reveals dynamic paracrine control of cellular variation. *Nature*, 510(7505):363–369, 2014.

[75] Khoa A Tran, Stefan J Pietrzak, Nur Zafirah Zaidan, Alireza Fotuhi Siahpirani, Sunnie Grace McCalla, Amber S Zhou, Gopal Iyer, Sushmita Roy, and Rupa Sridharan. Defining reprogramming checkpoints from single-cell analyses of induced pluripotency. *Cell reports*, 27(6):1726–1741, 2019.


Input Data
Expression datasets from GEO Database
DNA sequence motifs
Chromatin accessibility
#Samples
#Genes
Expression
#Regulators
#Genes
Network connectivity
#Samples
#Regulators
Activity of regulators
gene_j
$\min \| a_{j,}P - e_{j,.} \| + \lambda \| a_{j,.}.w'_{j,.} \|$
Regularized NCA
Network function
Network structure
$\Psi(X_1,X_2)$
Network inference
MERLIN+P+TFA
Estimated TFA
Inferred network


GRN inference
Expression data
mESC, hESC, LCL, MCF7
Collected from GEO and MetaSR
MERLIN+P+TFA
Inferred Network

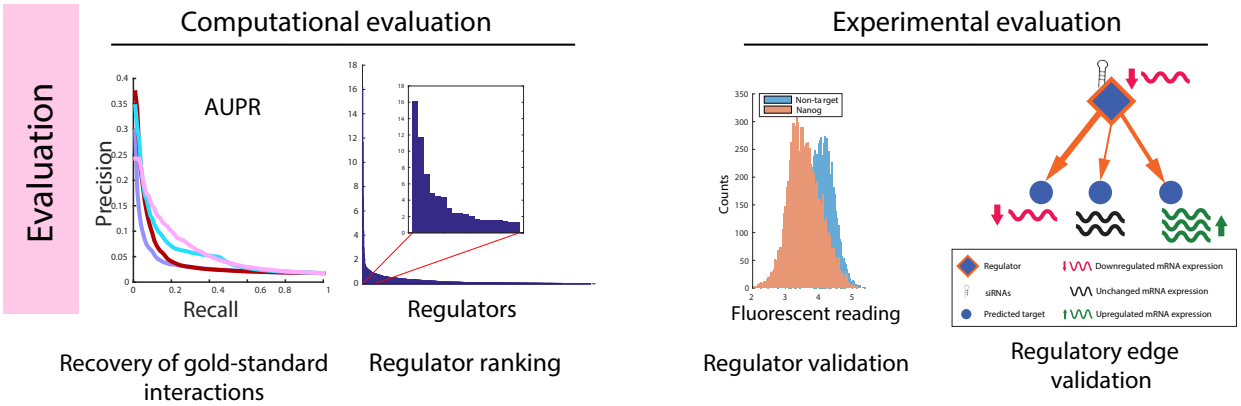

Evaluation
Computational evaluation
Experimental evaluation
AUPR
Precision
Recall
Regulators
Counts
Non-target
Nanog
Fluorescent reading
Regulator
siRNAs
Predicted target
Downregulated mRNA expression
Unchanged mRNA expression
Upregulated mRNA expression
Recovery of gold-standard interactions
Regulator ranking
Regulator validation
Regulatory edge validation

Figure 1. The MERLIN+P+TFA framework for inferring genome-scale gene regulatory networks (GRNs) with hidden transcription factor activity (TFA) levels. MERLIN+P+TFA has two steps: (1) Regularized network component analysis (NCA) step, (2) GRN inference with MERLIN+P which uses prior knowledge and mRNA and TFA of regulators. Regularized NCA extends the existing NCA method to incorporate the confidence levels in the input network for TFA estimation. This step takes an $m \times s$ genome-scale gene expression matrix, E of m and a noisy prior network connecting the genes to p regulators, as input and estimates the TFA P of the p regulators by minimizing the error $||E - AP||_2^2$. Here, "A" is the updated "connectivity matrix" of the regulators and target genes and is initialized based on the input prior network. A and P are estimated using an iterative algorithmy. In regularized NCA, the objective is extended to add a regularization accounting for the confidence of a regulator-target pair (w_{ji}) available in the prior. The output of MERLIN+P+TFA is the TFA levels and the final inferred network. MERLIN+P+TFA was applied to simulated and real expression datasets from yeast, human and mouse. The inferred GRNs were evaluated using multiple metrics (AUPR, predictable TFs, regulator ranking). The mouse embryonic stem cell (mESC) network was further used to nominate ESC-specific regulators and experimentlaly validated with regulator knockdown expreriments and target expression post regulator knockdown.

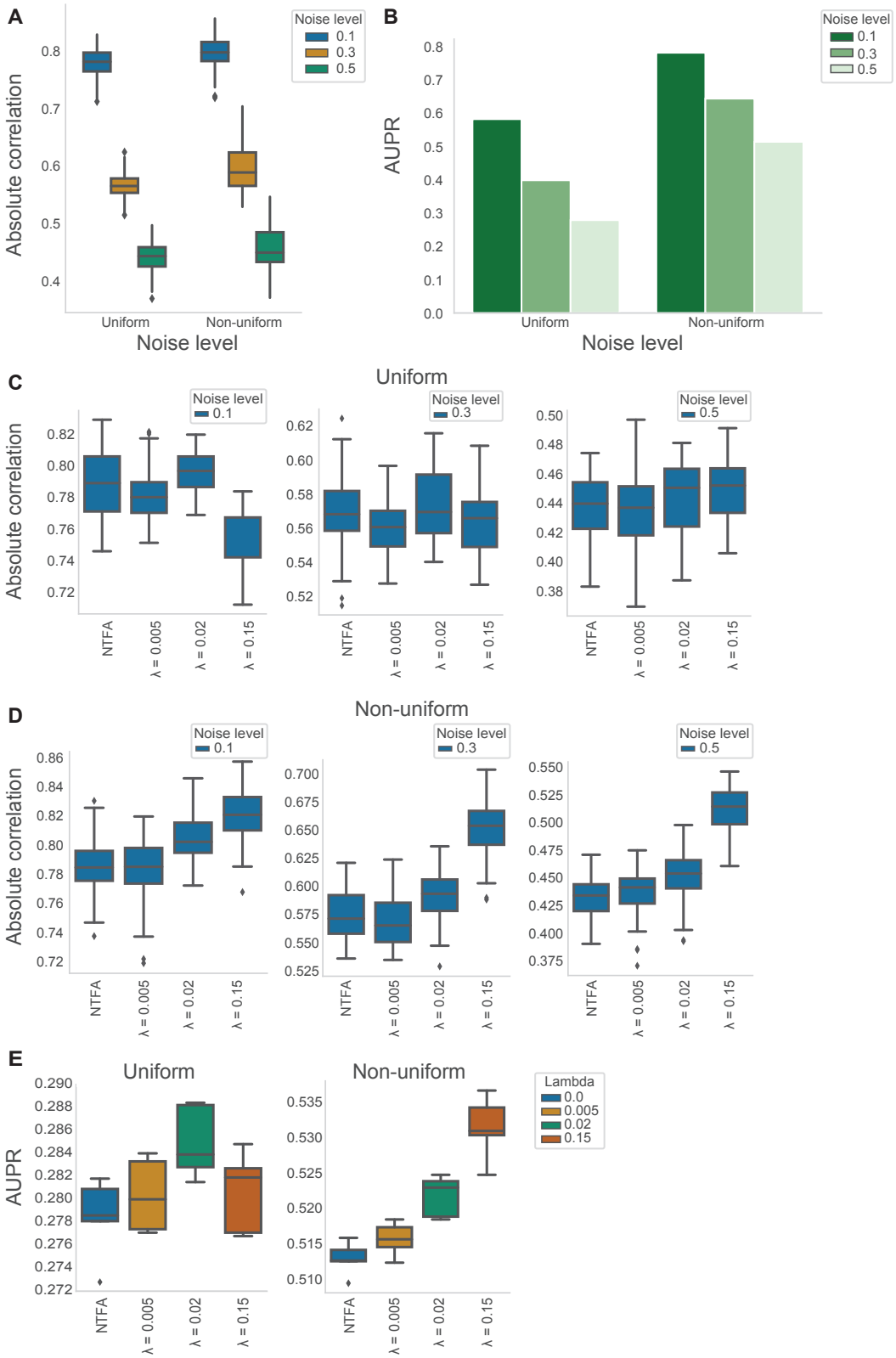

A
Absolute correlation
Noise level
Uniform
Non-uniform
Noise level
0.1
0.3
0.5
B
AUPR
Noise level
Uniform
Non-uniform
Noise level
0.1
0.3
0.5
C
Uniform
Absolute correlation
Noise level
0.1
0.3
0.5
NTFA
λ = 0.005
λ = 0.02
λ = 0.15
D
Non-uniform
Absolute correlation
Noise level
0.1
0.3
0.5
NTFA
λ = 0.005
λ = 0.02
λ = 0.15
E
Uniform
Non-uniform
AUPR
Lambda
0.0
0.005
0.02
0.15
NTFA
λ = 0.005
λ = 0.02
λ = 0.15

Figure 2. The effects of the prior network properties on the performance of NCA and regularized NCA with the simulated data. "Non-uniform" refers to the $(a = 5, b = 1)$ configuration. (**A-B**) The effect of including noise in the prior network on NCA's performance. (**A**) The absolute correlation between the TFA estimated from a noisy prior network (with 10%, 30%, or 50% noise) and the true TFA. (**B**) The area under the precision-recall curve (AUPR) values of the network inferred from the TFA estimated using the noisy prior network at a specific noise level. (**C-E**) The comparative performance of NCA and the regularized NCA (NCA+LASSO) in (**C-D**) estimating TFA for absolute correlation with 10%, 30%, or 50% noise and (**E**) inferring network given the prior network with 50% noise.

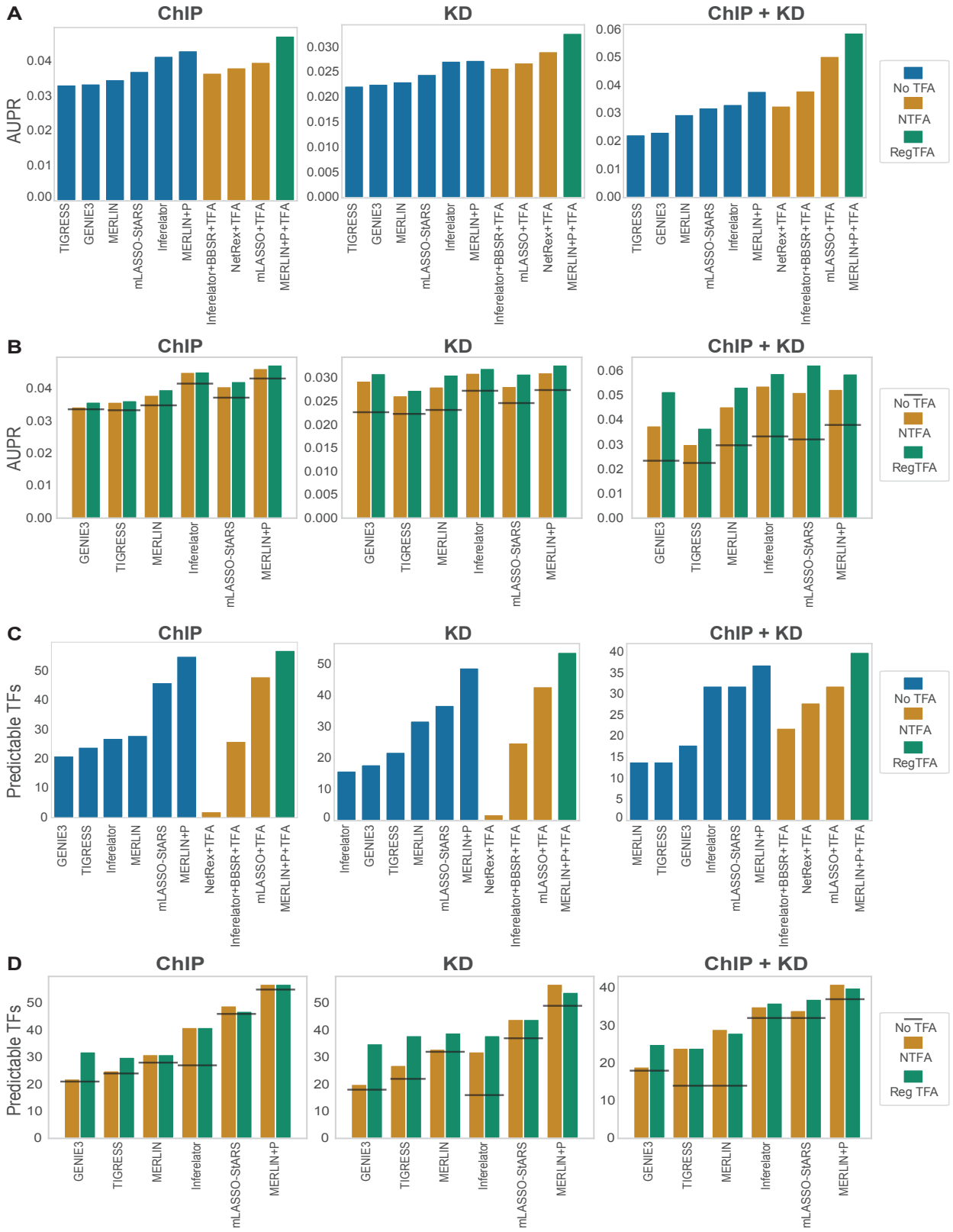

A
ChIP
KD
ChIP + KD
AUPR
TIGRESS
GENIE3
MERLIN
mLASSO-StARS
Inferelator
MERLIN+P
Inferelator+BBSR+TFA
NetRex+TFA
mLASSO+TFA
MERLIN+P+TFA
No TFA
NTFA
RegTFA
B
C
Predictable TFs
D
Reg TFA

Figure 3. Performance of GRN inference methods on the yeast dataset. (**A**) AUPR scores of methods that were applied without including TFA (No TFA), with unregularized TFA (NTFA), or using regularized TFA (RegTFA) with respect to the ChIP, KD, and the intersection of the ChIP and KD gold standard networks (ChIP + KD). Shown are the highest AUPRs across different regularizations. For {GENIE3, TIGRESS, MERLIN, MERLIN+P}, the unregularized AUPR scores correspond to the TFA estimated with NCA ($\lambda = 0.000$). For {mLASSO-StARS, Inferelator}, the unregularized AUPR scores correspond to the TFA internally estimated by the methods themselves. We exclude method NetREX since it could not be used without its internally estimated TFA. (**B**) AUPR scores of the inferred networks comparing regularized (RegTFA) and unregularized TFA (NTFA) across the three gold standards. No TFA is shown as a horizontal line for the respective algorithm. (**C**) The number of predictable TFs in the inferred networks. Shown are the highest value of this metric across different regularizations computed using top 30k edges. (**D**) Predictable TFs of inferred networks for regularized (RegTFA) and unregularized TFA (NTFA) across the three gold standards. The horizontal line depicts the No TFA setting.

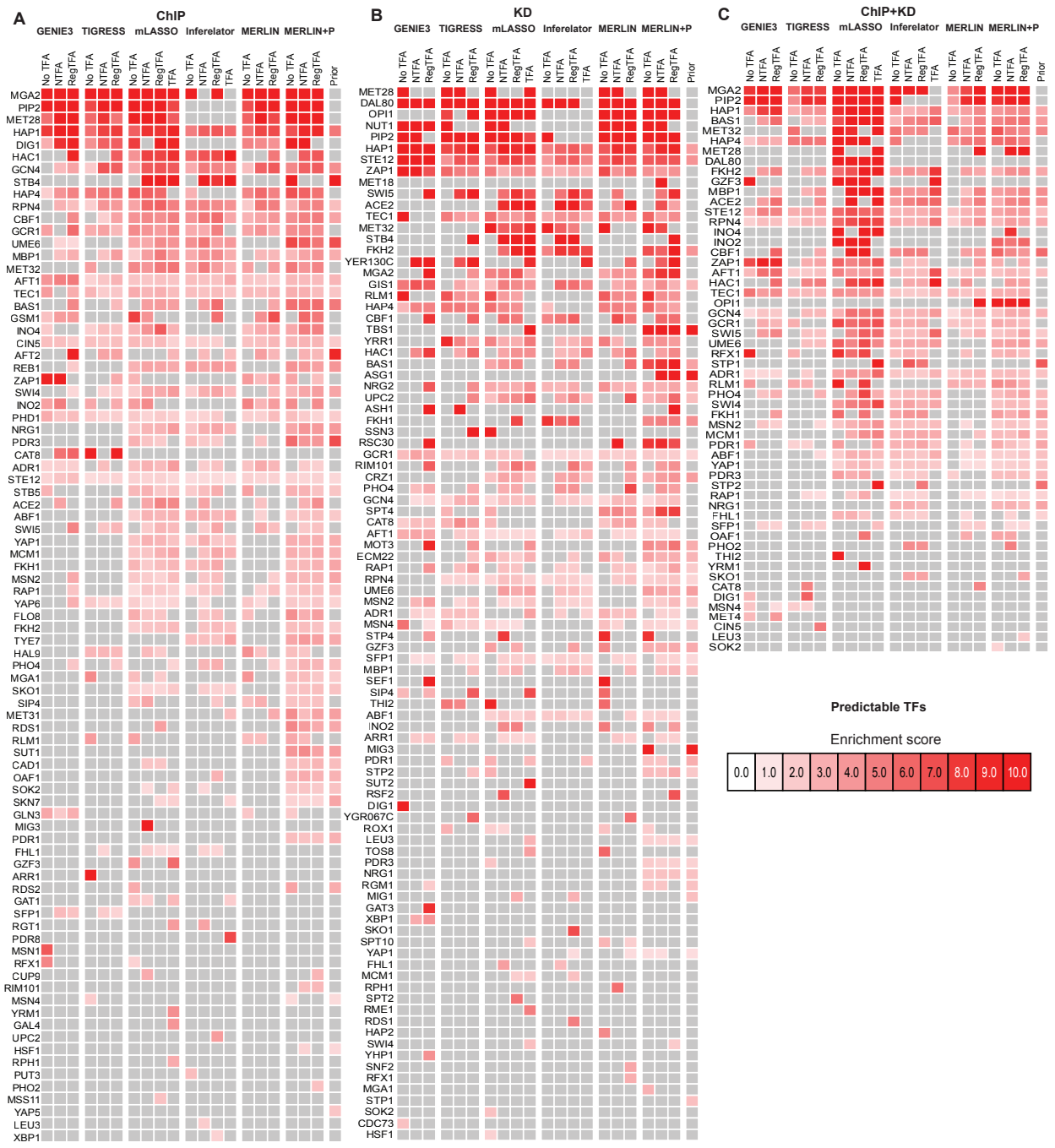

A
ChIP
GENIE3
TIGRESS
mLASSO
Inferelator
MERLIN
MERLIN+P
No TFA
NTFA
RegTFA
TFA
Prior
B
KD
C
ChIP+KD
Predictable TFs
Enrichment score
0.0
1.0
2.0
3.0
4.0
5.0
6.0
7.0
8.0
9.0
10.0

Figure 4. The enrichment scores of each TF (row) in the networks inferred by different algorithms for different TFA settings (NTFA, RegTFA, TFA) and different gold standard networks–(**A**) ChIP, (**B**) KD and (**C**) Intersection of the ChIP and KD (ChIP+KD). NTFA: no regularization; RegTFA: regularized TFA with the best metric; TFA: shown for mLASSO-StARS and Inferelator represents the TFA estimated by each algorithm's inbuilt TFA estimation process. The higher the enrichment score, the more significant the overlap is between the targets of a TF between the inferred and gold standard network.

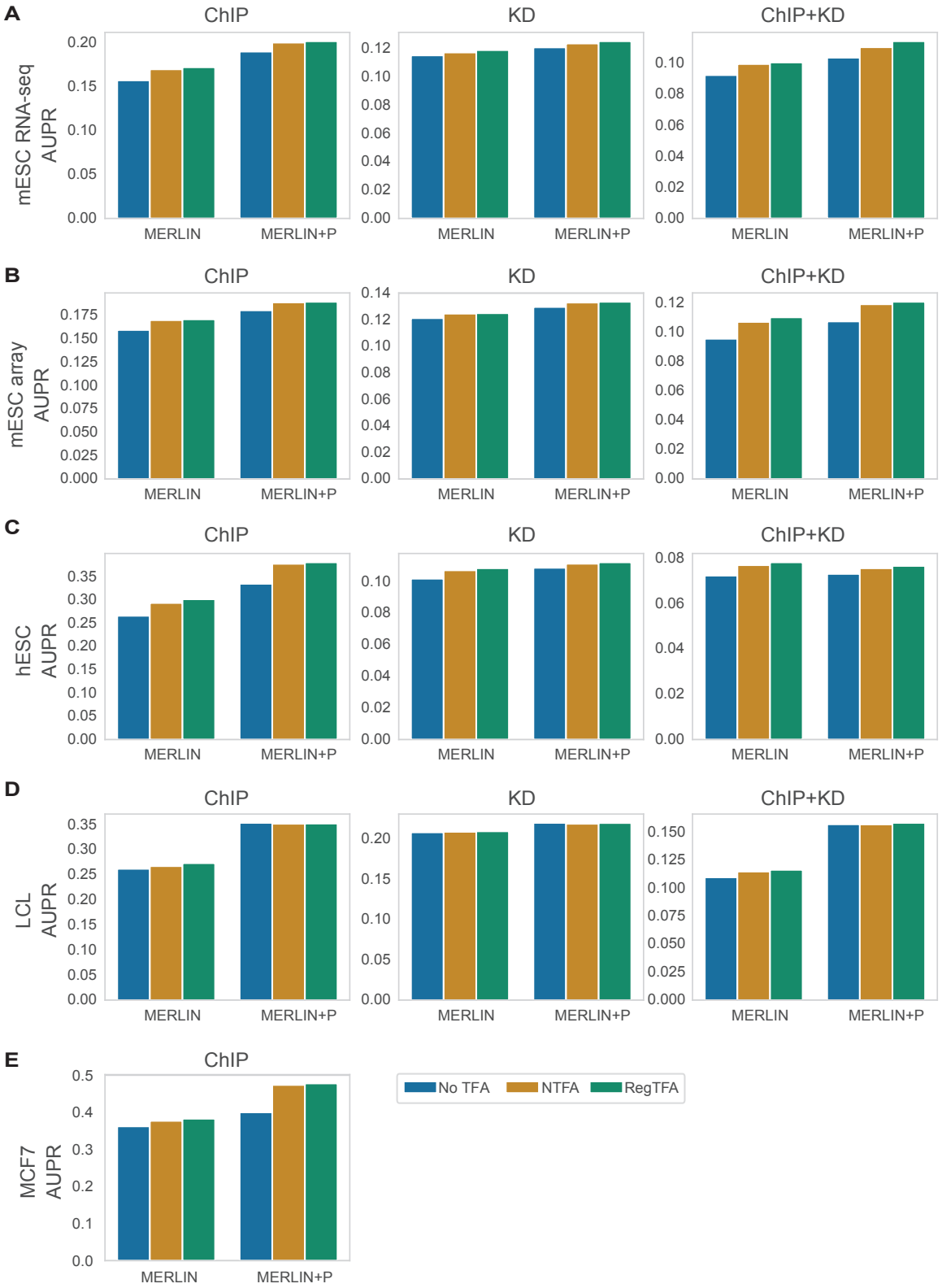

A
ChIP
KD
ChIP+KD
mESC RNA-seq AUPR
0.20
0.15
0.10
0.05
0.00
0.12
0.10
0.08
0.06
0.04
0.02
0.00
0.10
0.08
0.06
0.04
0.02
0.00
MERLIN
MERLIN+P
B
ChIP
KD
ChIP+KD
mESC array AUPR
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
0.12
0.10
0.08
0.06
0.04
0.02
0.00
MERLIN
MERLIN+P
C
ChIP
KD
ChIP+KD
hESC AUPR
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0.10
0.08
0.06
0.04
0.02
0.00
0.08
0.06
0.04
0.02
0.00
MERLIN
MERLIN+P
D
ChIP
KD
ChIP+KD
LCL AUPR
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0.20
0.15
0.10
0.05
0.00
0.150
0.125
0.100
0.075
0.050
0.025
0.000
MERLIN
MERLIN+P
E
ChIP
MCF7 AUPR
0.5
0.4
0.3
0.2
0.1
0.0
MERLIN
MERLIN+P
No TFA
NTFA
RegTFA

Figure 5. The AUPR scores of the networks inferred with MERLIN and MERLIN+P from five mammalian datasets: mESC RNA-seq (**A**), mESC array (**B**), hESC (**C**), LCL (**D**), MCF7 (**E**), on three gold-standard networks (ChIP, KD, and the intersection of ChIP and KD (ChIP+KD)). Shown are AUPR scores for each method for three TFA settings: expression data only (No TFA), NCA having $\lambda = 0$ (NTFA), and regularized TFA (RegTFA). The best performing regularized NCA result was selected for each comparison and is represented as "RegTFA".

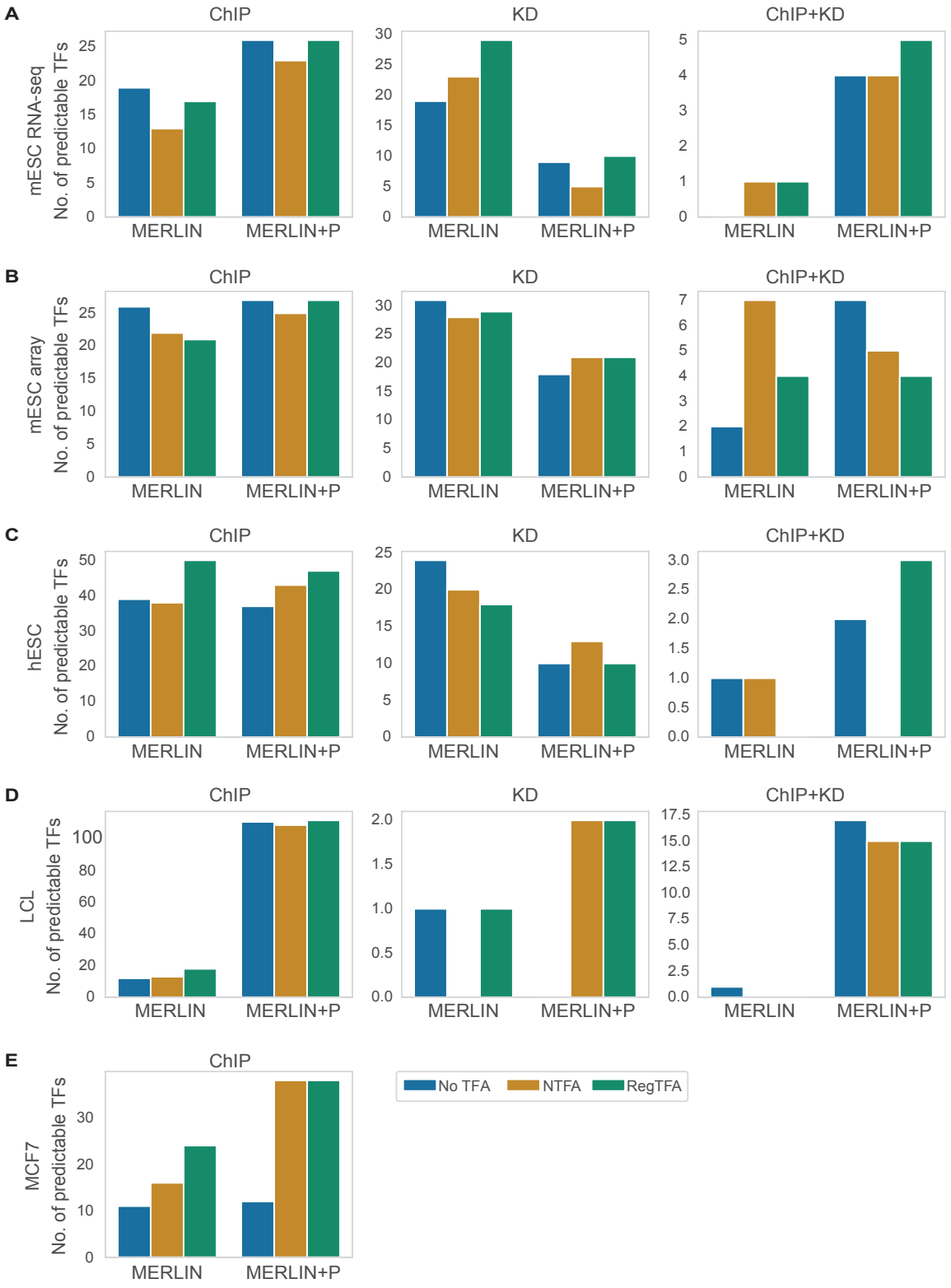

A
ChIP
KD
ChIP+KD
mESC RNA-seq
No. of predictable TFs
MERLIN
MERLIN+P
B
mESC array
C
hESC
D
LCL
E
MCF7
No TFA
NTFA
RegTFA

Figure 6. Number of predictable TFs in the networks inferred with MERLIN and MERLIN-P from five mammalian datasets, mESC RNA-seq (**A**), mESC array (**B**), hESC (**C**), LCL (**D**), MCF7 (**E**) on three gold-standard networks (ChIP, KD, and the intersection of ChIP and KD (ChIP+KD)). Shown are predictable TF counts for each method for three TFA settings: expression data only (No TFA), NCA having $\lambda = 0$ (NTFA), and regularized TFA (RegTFA). The best performing regularized NCA result was selected for each comparison and is represented as "RegTFA". Predictable TFs are computed on the top 30k edges.

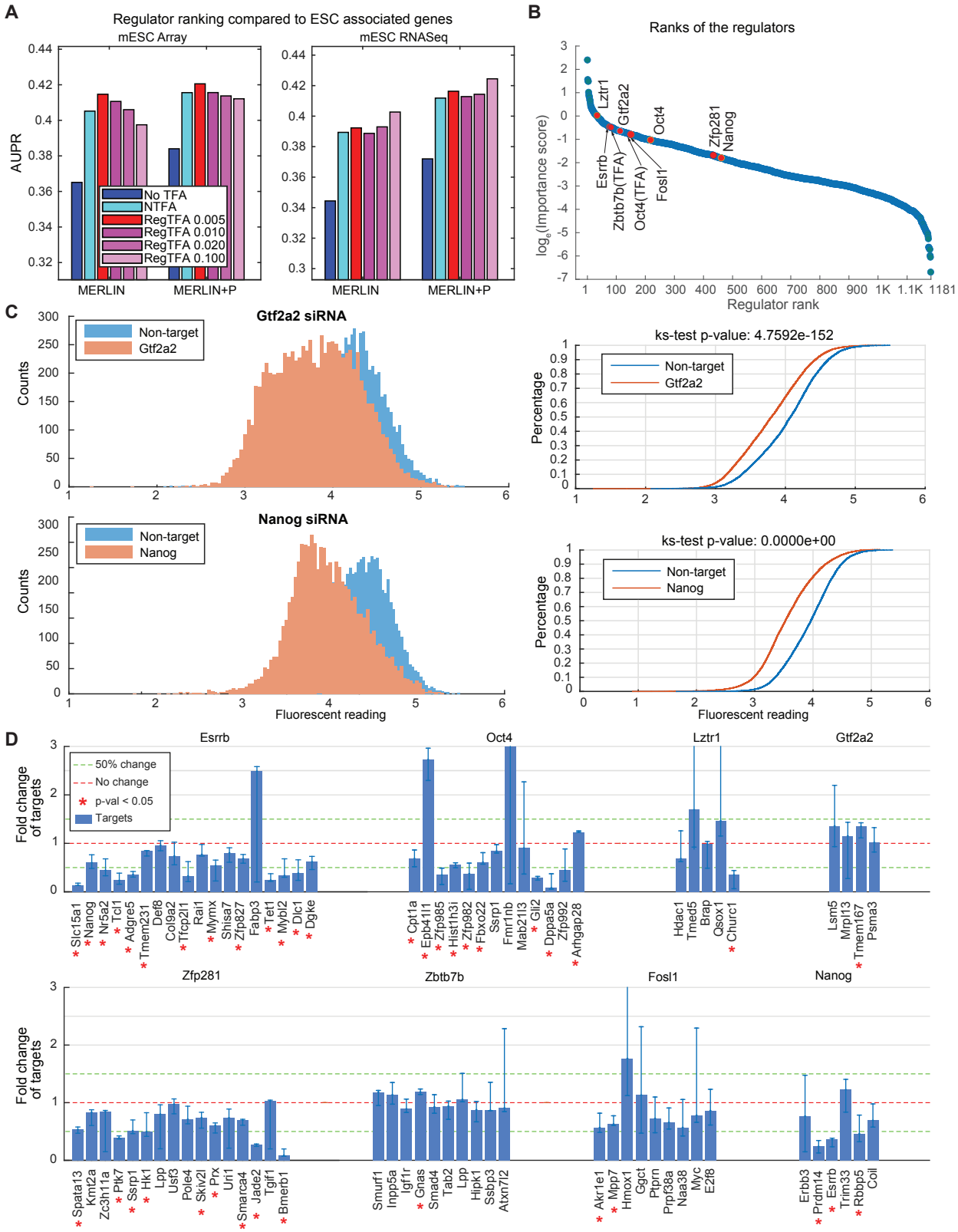
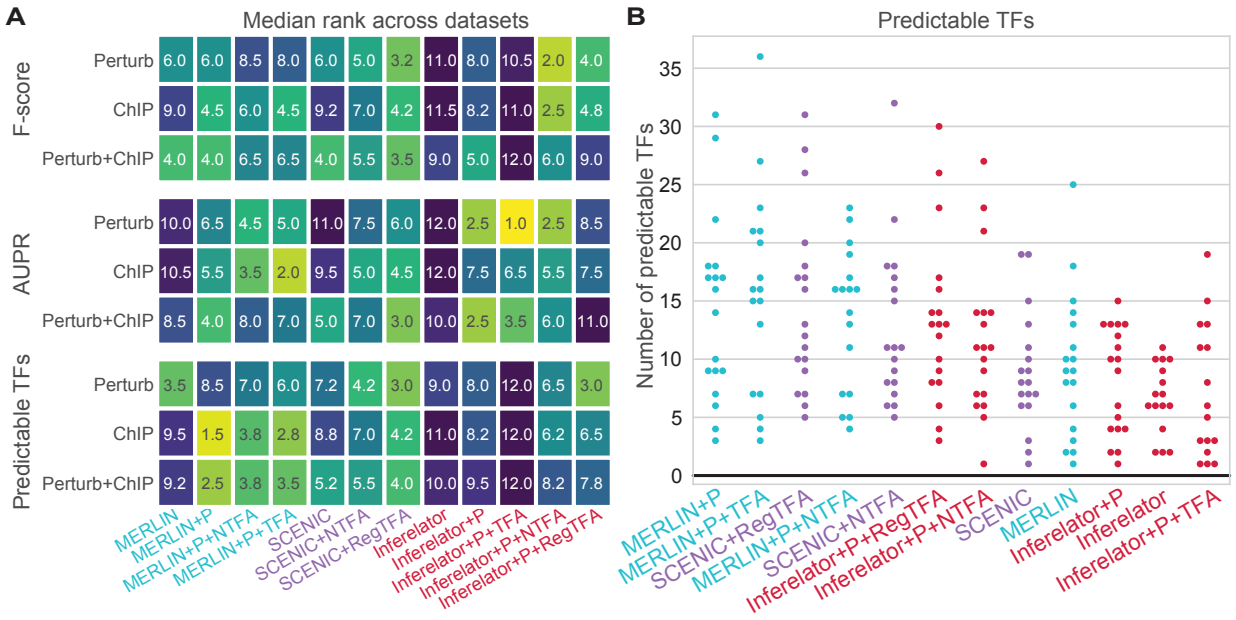

A
Regulator ranking compared to ESC associated genes
mESC Array
mESC RNASeq
AUPR
No TFA
NTFA
RegTFA 0.005
RegTFA 0.010
RegTFA 0.020
RegTFA 0.100
MERLIN
MERLIN+P
B
Ranks of the regulators
log_e(Importance score)
Regulator rank
Lztr1
Gtf2a2
Oct4
Zfp281
Nanog
Esrrb
Zbtb7b(TFA)
Oct4(TFA)
Fosl1
C
Gtf2a2 siRNA
Non-target
Gtf2a2
Counts
ks-test p-value: 4.7592e-152
Percentage
Nanog siRNA
Nanog
ks-test p-value: 0.0000e+00
Fluorescent reading
D
Esrrb
Oct4
Lztr1
Gtf2a2
50% change
No change
p-val < 0.05
Targets
Fold change of targets
Slc15a1 Nanog Nr5a2 Tcl1 Adgre5 Tmem231 Def8 Col9a2 Tfcp2l1 Rai1 Mymx Shisa7 Zfp827 Fabp3 Tet1 Mybl2 Dlc1 Dgke
Cpt1a Epb41l1 Zfp985 Hist1h3i Zfp982 Fbxo22 Ssrp1 Fmr1nb Mab21l3 Gil2 Dppa5a Zfp992 Arhgap28
Hdac1 Tmed5 Brap Qsox1 Churc1
Lsm5 Mrpl13 Tmem167 Psma3
Zfp281
Zbtb7b
Fosl1
Nanog
Spata13 Km12a Zc3h11a Ptk7 Ssrp1 Hk1 Lpp Usf3 Pole4 Skiv2l Prx Uri1 Smarca4 Jade2 Tgif1 Bmerb1
Smurf1 Inpp5a Igf1r Gnas Smad4 Tab2 Lpp Hipk1 Ssbp3 Atxn7l2
Akr1e1 Mpp7 Hmox1 Ggct Ptprn Prpf38a Naa38 Myc E2f8
Erbb3 Prdm14 Esrrb Trim33 Rbbp5 Coil

Figure 7. Experimental validation of the inferred mESC GRN. (**A**) The AUPR of the top N regulators in each inferred network of {MERLIN, MERLIN-P} for six different TFA settings (No TFA, NTFA, RegTFA with different regularizations) on the mESC array and RNA-seq datasets. The AUPR was computed using a gold-standard list of ESC related genes curated from the literature. (**B**) Regulators $(1, 181)$ in the MERLIN+P+TFA inferred network ranked based on importance score. Shown are rankings using regularized TFA $\lambda = 0.01$ (TFA0.010) from the mESC RNA-seq dataset. Four known ESC regulators (Esrrb, Nanog, Oct4, Zfp281) and 4 novel (Fosl1, Gtf2a2, Zbtb7b, Lztr1) TFs are highlighted in red cicles. "Oct4" is highlighted twice since it in the inferred GRN using gene expression (Oct4) and its estimated TFA (Oct4 (TFA)). (**C**) The effect of the siRNA knockdowns of "Gtf2a2" (a novel TF) and "Nanog" (a known TF). Left: histograms showing the distributions of Nanog fluorescence for Gtf2a2 (top) and Nanog (bottom) siRNA knockdown cells in the E14 cell line. The higher the intensity, the higher the protein expression of Nanog per cell, which is used as a proxy for measuring cellular pluripotency. The orange distribution is for the TF knockdown and the blue distribution is for "non-targeting siRNA control" experiments. For both Gtf2a2 and Nanog, the knockdown distributions (orange) is shifted significantly to the left with respect to the control distributions (KS test p-value $<$1E-10). Right: cumulative distribution function (CDF) of the respective histograms on the left. (**D**) Fold change of expression of targets on knocking down one of the 8 TFs as measured by RT-PCR. The targets with statistically significant fold change (t-test p-value< 0.05) are marked with asterisks. Please note that only the top $50,000$ edges of each inferred network were considered for generating the panels in this figure.



C

Algorithm	F-score	AUPR	Predictable TFs
MERLIN	7.0	10.0	8.8
MERLIN+P	5.5	6.0	3.5
MERLIN+P+NTFA	7.0	5.0	4.5
MERLIN+P+TFA	6.0	5.0	3.5
SCENIC	8.0	9.5	7.2
SCENIC+NTFA	6.0	6.5	5.2
SCENIC+RegTFA	4.0	4.5	3.8
Inferelator	11.0	12.0	10.5
Inferelator+P	7.5	4.0	9.2
Inferelator+P+TFA	12.0	3.5	12.0
Inferelator+P+NTFA	3.5	5.0	6.8
Inferelator+P+RegTFA	5.5	8.5	6.0

Figure 8. Assessing regularized TFA for GRN inference on scRNAseq datasets.(**A**) Algorithm median rank across 6 scRNAseq datasets based on performance on ChIP, Perturb, and intersection of the ChIP and Perturb (Perturb+ChIP) gold standards. Performance is measured with F-score, AUPR, and predictable TF metrics. (**B**) Predictable TF distribution for all scRNAseq datasets (Gasch, Jackson, A2S, FBS, Shalek, Han) and gold standard networks (ChIP, Perturb, Perturb+ChIP). Algorithms are ordered according to median rank performance (top performing algorithm placed on the left and the bottom performing algorithm placed to the right). (**C**) Overall median algorithm ranking across gold standards and scRNAseq datasets. Algorithm ranking is shown for F-score, AUPR, and predictable TFs metrics.