TRANSCRIPTIONAL REGULATORY GENOMICS: FROM MECHANISTIC MODELING TO CAUSAL INFERENCE

BY

PAYAM DIBAEINIA

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science
in the Graduate College of the
University of Illinois Urbana-Champaign, 2023

Urbana, Illinois

Doctoral Committee:

Professor Saurabh Sinha, Chair
Professor ChengXiang Zhai
Assistant Professor Mohammed El-Kebir
Associate Professor Jacqueline M. Dresch, Department of Biology, Clark University

ABSTRACT

Gene transcription refers to the process in which coding regions on the genome are copied into mRNA molecules through complex cellular mechanisms. This process is regulated through mechanisms that are encoded in the genome and are activated by cellular signals, enzymes, and proteins. Transcriptional regulation often involves a class of proteins called transcription factors regulating other genes. Although we have a limited understanding of regulatory mechanisms and associations in human and other species, discerning such characteristics of transcriptional regulation is of paramount importance in systems biology. The recent advancements in experimental techniques for high throughput measurements of cellular processes and their molecular signatures have led to increasingly growing biological databases including various “omics” datasets. These resources give rise to the emergence of novel computational models in systems biology that aim at understanding the genome of human and other species from data. These efforts include the development of data-driven methods for modeling transcriptional regulation using omics datasets. The general goal of such studies is to understand regulatory mechanisms and molecular interactions that drive transcriptional regulation. In practice, both the predictive accuracy and interpretability of these quantitative models are crucial to improve their efficacy. Especially, interpretability of the model is a key factor in various applications, from learning mechanistic regulatory insights to inferring causal regulatory relationships. This thesis is focused on the applications of interpretable computational models in learning and simulating transcriptional regulatory systems. In this Ph.D. thesis, I develop novel interpretable machine learning models for studying transcriptional regulations from two aspects: (1) learning biophysically-consistent regulatory mechanisms, (2) inference of causal regulatory associations. The first aspect of the study was pursued through quantitative and machine learning models that either explicitly encode regulatory mechanisms using biophysically-inspired functions or learn them in meaningful higher-order representations. The second aspect was achieved through an interpretation of non-linear machine learning models based on causal inference principles. Additionally, I leverage an existing mechanistic model for stochastic expression of genes to develop a novel framework for simulating gene expressions under causal regulatory networks at the cell-level resolution. This tool is useful for assessing the strength and weaknesses of causal regulatory inference algorithms.

To my parents, for their love and support.

ACKNOWLEDGMENTS

I would like to start by thanking my brilliant Ph.D. advisor, Professor Saurabh Sinha, who has constantly supported me in my academic endeavors and thought me invaluable lessons in science and life. He has patiently helped me to learn the principles of scientific research and has guided me to grow and build my skills. My Ph.D. studies in computer science was an amazing journey full of learning and experience which under the compassionate guidance of Saurabh has become one of the most precious chapters in my life. Besides, I would also like to thank my thesis committee, Professor ChengXiang Zhai, Professor Mohammed El-Kebir, and Professor Jacqueline M. Dresch whose thoughtful feedbacks and directions have helped me shape my thesis research. I am also thankful to Professor Tandy Warnow, Professor Aditi Das and Professor Jian Peng for their collaborations and continued advice and supports.

I would also like to express my gratitude toward my former and current fellow members in Sinha's Lab who their kindness and support has driven my work toward excellence. Of course, I am grateful to my dear family that helped me in this long journey with their unconditional support. And finally, I want to thank my partner and companion in life, Niloofar, who has always encouraged me and sincerely supported me in every single step of my Ph.D. journey.

The work presented in Chapter 2 was supported by the National Institutes of Health (grants R01GM114341, R35GM131819 to Saurabh Sinha, in part) and DOE Center for Advanced Bioenergy and Bioproducts Innovation (U.S. Department of Energy, Office of Science, Office of Biological and Environmental Research under award number DE-SC0018420). The work presented in Chapter 3 was supported by the National Institutes of Health (grants R35GM131819A and R01GM114341A, to Saurabh Sinha). The work presented in Chapter 4 was supported by the National Institutes of Health (grant R35GM131819, to Saurabh Sinha) and DOE Center for Advanced Bioenergy and Bioproducts Innovation (U.S. Department of Energy, Office of Science, Office of Biological and Environmental Research under award number DE-SC0018420). People silhouettes used in Figure 4.1B were obtained from `Vecteezy.com`.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

DNA contains the instructions required for cells to develop and perform their normal functions. Although all cells of a living organism share copies of the same DNA, cells can differentiate into various types that ultimately form distinct tissues and organs. Cell development and differentiation is controlled by a complex "Transcriptional Regulatory Program" which is triggered by cellular signals. Transcription is the process in which certain coding regions on the DNA, which are referred to as genes, are copied into mRNA molecules. These mRNA molecules will be later converted to protein molecules through a process called translation. Interestingly, although the non-coding regions of the DNA do not encode for proteins they are believed to play a key role in regulating the process of transcription, and such regulations underlies the cell differentiation process noted above. Enhancers are segments in the non-coding DNA regions that regulate the transcription of nearby or faraway genes. In metazoans, transcriptional regulation involves the binding of certain proteins called "transcription factors" (TF) to the enhancer regions and is followed by complex mechanisms and interactions among DNA, bound TFs and other proteins. Genetic and environmental factors can cause perturbations in this complex regulatory program which in turn can lead to different diseases such as cancer. Understanding transcriptional regulation not only elucidates the disease mechanisms, but also enables effective prevention, treatment and targeted drug design.

This thesis aims at understanding transcriptional regulation in a data-driven manner, first, from a mechanistic perspective to decipher the mechanisms of gene regulation, and second, from a causal perspective to identify perturbations in regulatory programs caused by an intervention such as disease. In the remainder of this chapter, we will first discuss the computational efforts to model transcriptional regulations. Next, we will formally define "Gene Regulatory Networks" (GRN) and will discuss the possibility of learning these networks from single-cell RNA sequencing data.

1.1 COMPUTATIONAL MODELS OF TRANSCRIPTIONAL REGULATION

Developing quantitative models for transcriptional regulation through mathematical and machine learning models has received significant attention over the past few years [1, 2, 3]. Especially, a significant effort has been made to incorporate prior biophysical knowledge into these models in order to obtain models that are interpretable and can explain regulatory mechanisms. Various computational studies have reached this goal by hard-coding mathematical functions, which contain some representations for known regulatory mechanisms, into their model architecture (e.g., GEMSTAT [4]). Nearly all of these biophysics-inspired functions rely on some properties of the TFs' cognate sites on the enhancer sequence; therefore, quantitative models commonly take as input the enhancer sequence annotated by potential transcription factor binding sites (TFBS). Identification of TFBSs on enhancer sequence is achieved through matching different segments of the enhancer sequence against TFs' binding specificities represented by *position weight matrices* (PWM).

In 1985, Shea and Ackers [5] proposed a framework for quantitative modeling of transcriptional regulation based on enumerating possible configurations of an enhancer where each configuration is defined by indicating the bound or non-bound status of TFBSs on the enhancer. In their model, relative probability of a configuration is computed by Boltzmann distribution principles from the TF-DNA binding energies and TF-TF interaction energies present in the configuration. Shea and Ackers framework is often referred to as a thermodynamics based model of transcriptional regulation due to its reliance on statistical thermodynamics principles. Since then, several studies have employed variations of Shea and Ackers model for quantitative modelling of transcriptional regulation and attempted to modify, extend and improve the underlying thermodynamics model of this framework [4, 6, 7, 8].

In chapter 2, we will focus on a well-studied regulatory system in *Drosophila melanogaster* and we will interrogate different formalisms of thermodynamics models to evaluate their interpretations of regulatory mechanisms encoded in enhancers. Moreover, we will introduce a novel convolutional neural network model for sequence encoding of transcriptional regulation and will show that interpretability can be embedded inside the model architecture. In particular, we focus on the distance-dependent pairwise interactions between TFBSs in the enhancer and show that our model, which has a simple attention mechanism to capture this effect, provides interesting interpretations for such pairwise interactions.

1.2 GENE REGULATORY NETWORKS

Proteins, including transcription factors that bind to the enhancer regions, are the final products of a two-step process consisting of gene transcription and RNA translation. This view of gene expression enables us to define directional associations $A \rightarrow B$ between gene A and B, when the proteins produced from the transcription and translation of gene A play a regulatory role in the transcription of gene B. Compiling all such associations results in a Gene Regulatory Network (GRN) that summarizes all the regulatory relationships between genes. An extensive effort has been made toward developing computational methods and algorithms to infer GRNs from biological data [9]. These methods include statistical and Machine Learning (ML) algorithms that aim at inferring GRNs from gene expression data or from integrated multi-omics data [9].

The advent of single-cell RNA-sequencing (scRNA-seq) technologies has opened new doors for GRN inference from transcriptomics data. In contrast to bulk RNA sequencing where gene levels are profiled in an aggregated pool of heterogeneous cells, scRNA-seq profiles the expression of genes in individual cells, and therefore provides a high resolution profile on genes' expression. This higher resolution provides additional source of variation to be explained by the models, more samples for training complex ML models and greater statistical power. On the other hand, this relatively new technology significantly suffers from technical noise. Dropout is one of the most important types of technical noise in scRNA-seq which introduces significant missing values (i.e., false zero expressions) to the gene expression profiles. While cell-level resolution offered by scRNA-seq is beneficial, technical noise present in this technology is detrimental for GRN inference. There is an ongoing effort to develop novel GRN inference methods which have higher tolerance toward technical noise in scRNA-seq data and can achieve higher gains from the increased resolution in these data sets.

In chapter 3, we will borrow an existing stochastic model for gene regulation and transform it to a model for transcriptional regulation at the single-cell level. Following an existing paradigm for numerical simulations of such a stochastic model, we will introduce a novel open-source package, SERGIO, for simulations of gene regulatory networks at the single-cell resolution. This simulator is especially useful for benchmarking GRN inference algorithms since the lack of information about the true underlying GRNs prevents a reliable evaluation of GRN inference algorithms using real data sets. In addition to the benchmarking of GRN inference algorithms, SERGIO is useful for evaluating the performance of a wide variety of the emerging algorithms in the field of single-cell transcriptomic analysis. The applications of this tool extend beyond benchmarking of single-cell analysis tools and it can be used to analyze the expression dynamics under GRN perturbations. We will showcase this capability of SERGIO by employing this tool to study the regulation of T-cell differentiation.

Another important challenge in GRN inference from bulk and single-cell transcriptomics data is the presence of confounding variables. Confounders can potentially induce correlations between gene pairs neither of which has any regulatory role in the transcription of the other one. For example, two genes that are not directly associated might be correlated due to a third gene that regulates both of them, or due to a third gene that regulates one of them, but itself is regulated by the other. This implies that correlation-based metrics are not necessarily sufficient for inference of causal regulatory associations. Some early GRN inference algorithms attempted to address this issue by employing additional criteria alongside correlation-based metrics to distinguish direct from non-direct regulatory associations (e.g., ARACNE [10]). More recent GRN inference algorithms have relied on machine learning models that commonly employ regression or regularized regression (e.g., GENIE3 [11]). The general goal of these methods is to model the expression of a target gene as a function of the other genes' expression, and infer a GRN based on downstream interpretation of these models. However, it should be noted that machine learning models often rely on correlation signals, and therefore suffer from the same limitations discussed above. In general, it can be shown that when two genes are perfectly correlated, but only one of them regulates a third gene, no quantitative method that only relies on expression data can identify the true regulator of the third gene. Despite these fundamental challenges, a causal perspective in GRN inference can potentially alleviate the problems arising from confounders.

In chapter 4, we will study GRN inference from a causal perspective. In particular, we will focus on differential GRN inference between two groups or conditions, such as treatment versus control group, or disease versus healthy condition. We will define a causal quantity that measures the changes in causal gene associations and will propose a procedure for estimating this quantity from transcriptomic data. Through extensive benchmarking analysis using simulated single-cell expression data, we will show that this method significantly outperforms existing algorithms for differential GRN inference that are based on differential correlation metrics. We will also showcase the application of our method in understanding the differential regulatory program that underlies Alzheimer's disease (AD). We will achieve this by applying our method to one of the largest publicly available single-cell expression data sets for AD to date.

CHAPTER 2: MECHANISTIC MODELING OF GENE REGULATION

This chapter is a reproduction of the work published in the Nucleic Acids Research journal [12].

2.1 ABSTRACT

Deciphering the sequence-function relationship encoded in enhancers holds the key to interpreting non-coding variants and understanding mechanisms of transcriptomic variation. Several quantitative models exist for predicting enhancer function and underlying mechanisms; however, there has been no systematic comparison of these models characterizing their relative strengths and shortcomings. Here, we interrogated a rich data set of neuroectodermal enhancers in *Drosophila*, representing cis- and trans- sources of expression variation, with a suite of biophysical and machine learning models. We performed rigorous comparisons of thermodynamics-based models implementing different mechanisms of activation, repression, and cooperativity. Moreover, we developed a convolutional neural network (CNN) model, called CoNSEPT, that learns enhancer "grammar" in an unbiased manner. CoNSEPT is the first general-purpose CNN tool for predicting enhancer function in varying conditions, such as different cell types and experimental conditions, and we show that such complex models can suggest interpretable mechanisms. We found model-based evidence for mechanisms previously established for the studied system, including cooperative activation and short-range repression. The data also favored one hypothesized activation mechanism over another and suggested an intriguing role for a direct, distance-independent repression mechanism. Our modeling shows that while fundamentally different models can yield similar fits to data, they vary in their utility for mechanistic inference. CoNSEPT is freely available at: https://github.com/PayamDiba/CoNSEPT.

2.2 INTRODUCTION

Transcriptional regulation in metazoans is mediated by proteins called transcription factors (TF) that bind to specific sites in regulatory regions called enhancers [13], via TF-DNA interactions and cooperative DNA binding [14]. Many TFs that occupy their respective binding sites interact with each other and with the transcription start site over long and short distances to influence the recruitment of transcription machinery and transcription initiation [15]. These simultaneous interactions establish a complex regulatory code that drives a gene's expression in varying cellular conditions or cell types.

Gene regulatory mechanisms encoded in an enhancer can be fairly complex and have been the subject of numerous studies, notably the detailed experimental dissection of developmental enhancers in Drosophila [16, 17, 18]. Such studies have significantly advanced our understanding of regulatory mechanisms. For example, certain TFs that are known to inhibit transcription ("repressors") have been found to function only if bound at short distances from activator binding sites [19, 20, 21]. As another example, TFs that are responsible for promoting transcription ("activators") have been shown in some cases to contribute synergistically to the gene's expression [22, 23], possibly due to cooperative DNA-binding by TFs to adjacent binding sites. These complex regulatory mechanisms may be mirrored in rules underlying the arrangement of binding sites in an enhancer, a phenomenon sometimes called cis-regulatory "grammar" [24, 25, 26, 27]. Precise characterization of the sequence-function relationship encoded in enhancers therefore requires interpreting how a collection of binding sites for one or more TFs works together and how such combinatorial action is influenced by site arrangements as well as varying TF concentrations in different cellular contexts. The challenge goes beyond a general understanding of the underlying principles (e.g., "TF X is a short-range repressor" or "TFs X and Y activate synergistically"): often, one seeks a quantitative model capable of predicting a given enhancer's regulatory output in varying cellular conditions and the effect of sequence variations such as disease-related non-coding polymorphisms within the enhancer [28]. Indeed, such "sequence-to-expression models" are an active area of research today [1].

The most direct efforts to deciphering the cis-regulatory code of enhancers have been through experiments that record expression readouts of many variants of an enhancer [7, 26] or many enhancers under similar control [24, 29, 30]. Often, the variants are synthetic constructs that manifest a diversity of TF binding site composition and arrangement; in some cases, their functions (expression level) are determined in a single cellular context [24] while occasionally the expression readout is obtained in varying cellular contexts, e.g., nuclei of the early *Drosophila* embryo [6, 17, 30, 31] or *Ciona* embryos. Such data sets are then analyzed through a specialized mathematical model whose structure incorporates mechanistic hypotheses and whose parameters represent quantitative details of those mechanisms, such as activation strength and distance-dependence of cooperative and repressive interactions among binding sites [1, 2, 6, 7, 8, 32, 33, 34]. The most effective frameworks for such mathematical mechanistic modeling have been based on equilibrium thermodynamics [4, 26, 35], although non-equilibrium models have also been motivated and proposed [36, 37, 38].

Different thermodynamics-based models implement regulatory mechanisms in different ways. For example, some models accord the activation by a TF exclusively to its DNA-binding ability [7, 8], while others [4, 33] model activation by additional free parameters (beyond those for TF-DNA binding) that explicitly represent long-range interactions between activators and the basal transcription machinery (BTM). Similarly, different models employ various representations of short-range repression, e.g., via quenching of a bound activator or mediating local chromatin remodeling through recruitment of co-repressors [4, 6, 7, 8], and some even accommodate longer-range repression via direct inhibitory interaction between the bound repressor and BTM [4, 39]. Typically, an enhancer dataset is analyzed using a mathematical model with a pre-determined structure (qualitative mechanisms) and its parameters are tuned to fit the data. It is rare for the same study to investigate models with varying assumptions about regulatory mechanisms to determine those best supported by the data. Here, we sought to bridge this gap by asking if various existing formalisms differ in their ability to model sequence-to-expression relationships.

In addition to thermodynamics-based models, statistical and machine learning (ML) models have also provided useful quantitative descriptions of cis-regulatory encoding [40, 41]. Such models typically avoid strong pre-conceptions about underlying mechanisms, providing a "data-driven" approach to quantitative modeling of enhancers, as a counterpoint to the "hypothesis-driven" approach of thermodynamics-based models. Recently, Avsec et al. [42] trained a deep neural network model on TF-DNA binding data and showed how interrogating the trained model can reveal mechanistic insights. While neural network models have been frequently applied to TF-DNA binding and epigenomic data [43, 44, 45], their utility for sequence-to-expression modeling remains to be demonstrated. This is partly because data sets with direct expression measurements on enhancers [4, 7, 26] in diverse conditions are of relatively modest sizes and the models tend to be heavily parameterized. Thus, the second major motivation of this study was to test if ML models and especially neural network models provide a practical alternative to thermodynamics-based models for enhancer sequence-function relationships, and to assess their relative merits and weaknesses.

Motivated by the above considerations, we tested a suite of quantitative models, including linear models, thermodynamics-based models, and a newly developed convolutional neural network (CNN) model, on a rich sequence-expression data set previously reported by Sayal et al. [26]. The data include expression measurements of an enhancer that drives expression of the *rhomboid* (*rho*) gene in the D. melanogaster embryo, along with several synthetic variants of the enhancer and several endogenous enhancers with similar regulatory function.

The spatial expression patterns driven by these enhancers are known to be regulated by two activators, Dorsal (DL) and Twist (TWI) and one repressor, Snail (SNA) [16, 26]. Importantly, the dataset not only represents variation of enhancer sequences, it also includes changes in cellular conditions.

We used rigorous model comparisons and prior mechanistic studies of this regulatory system to evaluate the suite of quantitative models. The thermodynamics-based models we tested included two different activation mechanisms, three repression mechanisms, and the presence or absence of cooperative activation via proximal binding sites. We tested linear and generalized linear models that combine binding site contributions additively, as well as an extension where pairwise TF interactions were allowed. Finally, we developed and tested a new neural network model for the expression driven by a given enhancer sequence in varying cellular conditions. This model, called "CoNSEPT" (**Co**nvolutional **N**eural Network-based **S**equence-to-**E**xpression **P**rediction **T**ool), utilizes user-provided DNA binding motifs and condition- or cell type-specific concentrations of TFs, and can thus quantify the regulatory role of each TF. Importantly, it learns salient rules of binding site arrangement in a purely data-driven manner, without presuming any particular distance-dependence function of TF-TF interaction as is commonly done in thermodynamics-based models.

Our modeling shows that the three broad categories of models are competitive with each other in terms of their ability to fit the enhancer-expression data set. We found that convolutional neural networks can be reliably trained on relatively modest-sized training data and can learn aspects of cis-regulatory grammar in a fully data-driven manner. To our knowledge, this is the first demonstration of a CNN predicting expression variation across sequences as well as across conditions. Our thermodynamics-based modeling showed that explicitly modeling the strength of activators is advantageous compared to ascribing activator strength solely to its DNA-binding, but that different biochemical mechanisms of short-range repression cannot be reliably distinguished based on the dataset. Both thermodynamics-based and neural network models detected a significant role for cooperative interaction between activator sites. Intriguingly, both types of models suggested a potential role for a direct repression mechanism that is not short-range, the predominant theory of repressor action for this system. Our baseline linear models showed good agreement with data but were, by design, limited in offering mechanistic insights into the cis-regulatory code, including rules of binding site arrangements and interactions. Overall, this work conveys a positive outlook for the modeler, who has at their disposal a variety of tools of varying complexity with which to understand a regulatory system at the level of enhancer sequences

2.3 MATERIALS AND METHODS

2.3.1 A gene expression data set with cis and trans variations

We analyzed a data set generated and first modeled by Sayal et. al. [26]. It includes expression levels driven by a well-studied enhancer of the gene *rho* (Figure 2.1A, henceforth called the *rho* enhancer) in the early *D. melanogaster* embryo. This enhancer is regulated by three transcription factors (TFs): Dorsal (DL), Twist (TWI) and Snail (SNA). DL and TWI are known to activate and SNA represses *rho* expression [16, 26]. Binding sites of these TFs in the *rho* enhancer are well mapped and shown in Figure 2.1A. The expression levels of rho driven by the wild-type (WT) enhancer were measured by Sayal et al. as a function of cellular (nuclear) position along the ventral-dorsal (V-D) axis of the embryo (Figure 2.1B), using reporter assays. The enhancer's expression profile is quantitatively represented by a 17-dimensional vector, where the 17 dimensions represent uniformly spaced positions or "bins" along the V-D axis from the ventral end to 40% of the V-D axis length. Concentration profiles of the three TFs are also available, as 17-dimensional vectors analogous to the enhancer expression profiles (Figure 2.1B. Moreover, similar expression profiles were generated for 37 synthetic variants of the WT *rho* enhancer, where each variant was constructed by mutagenesis of one or multiple TF binding sites (Figure 2.1C) (Our nomenclature for the enhancers is different from that of Sayal et al., see Appendix A, Supplementary Table A.1.). Thus, the data set captures expression variation across different trans contexts (cells at different V-D axis positions, with varying TF levels) as well as different cis contexts (WT enhancer and synthetic variants).

Figure 2.1D reports on a selection of the synthetic enhancers – those representing DL and/or TWI site deletions, showing the difference in activation by each enhancer compared to the WT enhancer. Interestingly, deleting the two sites with smallest individual effects ("D3" and "D4") simultaneously has the largest effect among all variants. Two other sites – "D2" and "T1" – individually have at least as much contribution as D3 and D4 individually, but their simultaneous deletion has a substantially smaller effect than the D3-D4 double deletion. This and other aspects of Figure 2.1D suggest non-linear regulatory contributions (Figure 2.1E) from sites in the rho enhancer, and present an interesting challenge for current mathematical models of cis-regulatory encoding: can existing models capture the subtle variations of function encoded in these variant enhancers, and if so, can they reveal new insights about the underlying regulatory mechanisms?

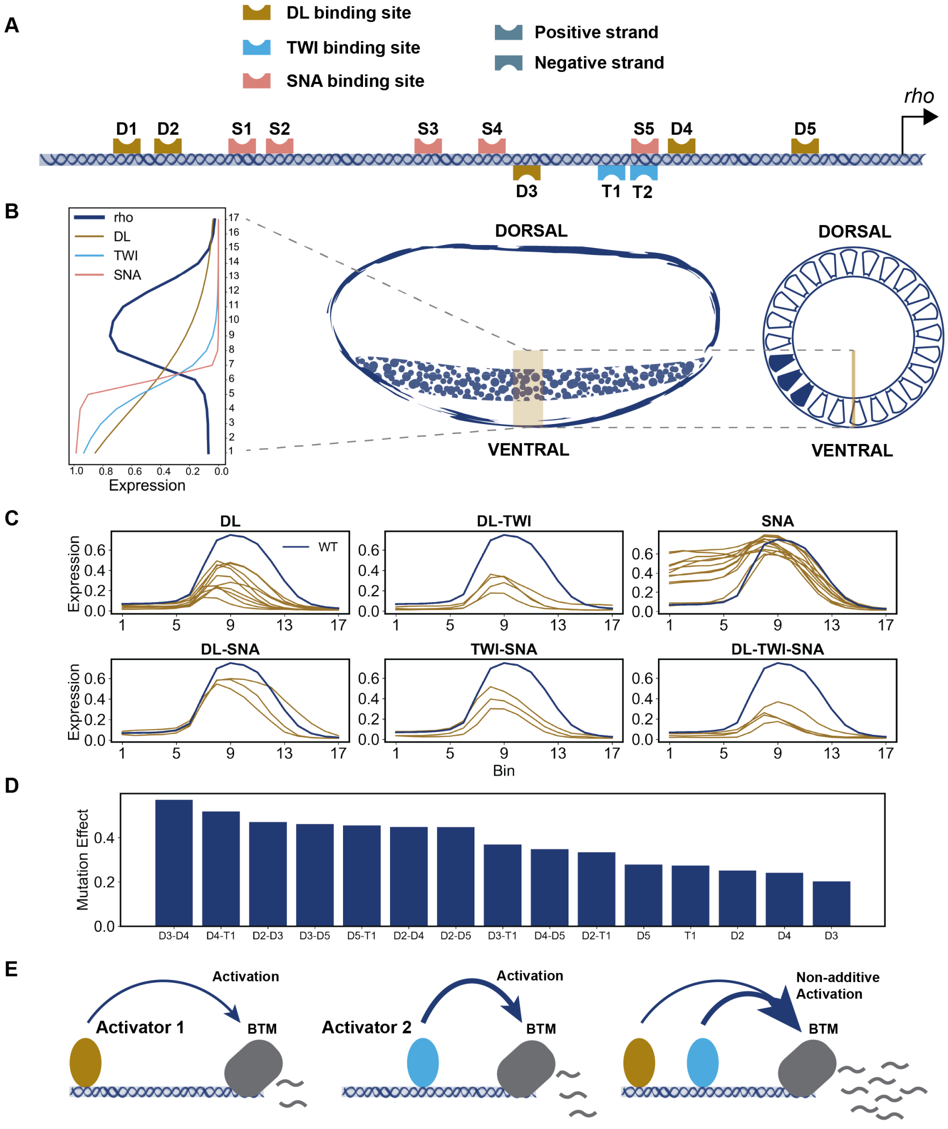

Figure 2.1: Overview of the data used in this study. (A) Schematic of the wild-type (WT) enhancer of the *Drosophila rho* gene. Binding sites of the three TFs were identified using the PWMs employed in [26]. All annotated sites agree with those found in [26] and except for D1 and S2 all the sites match with the in vitro footprinted sites characterized previously [26]. (B) The levels of the three regulators and the expression of rho driven by the wild-type enhancer in 17 equidistant points along 0-40% of ventral-dorsal (V-D) axis. (C) The expression of *rho* driven by perturbed enhancers (shown in brown) representing mutagenesis of binding sites of one or more TFs. (D) Each activator's site deletion (or combination thereof) is expected to reduce peak expression of rho (at bin 8 on the V-D axis); we therefore defined the effect of a variant enhancer (Y-axis) as the difference between the expression driven by it and the wild-type expression at this position of the axis. The effect of T2 single site deletion is not shown due to its overlap with the SNA site S5. (E) Schematic of synergistic activation, where the activation driven by two bound activators (right) is greater than the sum of their individual activation effects (left and middle).

To meet the above challenge, we trained diverse mathematical models that map TF concentration profiles and enhancer sequence to the enhancer's expression profile, for all 38 enhancers (WT and 37 variants, henceforth called the "training set") simultaneously. The accuracy, or "goodness-of-fit", of a model was measured by the root mean squared error (RMSE) between predicted and real expression profiles along the V-D axis, averaged over all enhancers; this is referred to as the "train error" below. The data set also includes expression profiles of 13 other enhancers that have expression profiles similar to *rho* (Appendix A, Supplementary Figure A1); these are orthologs of the *rho* enhancer from other *Drosophila* species or enhancers of other *D. melanogaster* genes with a neuroectodermal expression pattern similar to *rho*. We used these 13 enhancers, which represent greater sequence diversity than do the 38 enhancers in the "training set" (above), as the "test set". The average RMSE of a model on these enhancers is referred to as "test error" below. By fitting, evaluating and comparing various models that differ in their explicit encoding of biophysical mechanisms, we hoped to draw inferences about specific mechanisms that are supported by the data set.

2.3.2 Linear models

We tested three variants of linear models, viz., basic Linear Model (LM), Generalized Linear Model (GLM), and Generalized Linear Model with Quadratic terms (GLMQ).

LM Model: Expression (E) driven by an enhancer is a weighted sum of contributions from TFs that bind to the enhancer:

$$E = \sum_{t \in T} W_t[t]F_t + W_b \tag{2.1}$$

where T is the set of TFs, W_t is a TF-specific weight that reflects its activating or repressive role and strength, W_b is a "basal" expression parameter, $[t]$ is the concentration of TF t, and F_t reflects the total binding site presence of t on the enhancer, defined as:

$$F_t = \sum_{s \in S_t} exp(LLR(s) - LLR(S_{t,opt})) \tag{2.2}$$

where S_t is the set of all putative binding sites of t in the enhancer, $S_(t, opt)$ represents the strongest possible binding site of t and $LLR(x)$ denotes the log likelihood score of site x, calculated using the given Position Weight Matrix (PWM) of t and a provided background nucleotide distribution. This definition of site strength follows [46].

GLM Model: Expression driven by an enhancer is a sigmoid function of the LM model:

$$E = \frac{1}{1 + exp(-\sum_{t \in T} W_t[t]F_t + W_b)} \tag{2.3}$$

GLMQ Model: This defines the total contribution of TFs as a non-linear function of their concentration $[t]$ multiplied by their site strength F_t and then applies a sigmoid function to model saturation:

$$E = \frac{1}{1 + exp(-\hat{E})} \tag{2.4}$$

$$\hat{E} = \sum_{t \in T} W_t[t]F_t + \sum_{t \in T} V_t([t]F_t)^2 + U_{DL-TWI}[DL][TWI]F_{DL}F_{TWI} + W_b \tag{2.5}$$

where V_t is a free parameter associated with $t-t$ cooperative interaction. In addition to linear terms $[t]F_t$ and quadratic terms $([t]F_t)^2$ for each TF, this includes a term for heterotypic cooperativity of DL and TWI, as suggested in the literature [26, 47], with a tunable weight U_{DL-TWI}.

2.3.3 GEMSTAT Model

We tested seven thermodynamics-based models named after their main mechanistic aspects, namely Activation by Binding (AB), Activation by Potency (AP), repression by Quenching (Q), repression by Neighborhood Remodeling (NR), Direct repression (DIR), Cooperative binding (COOP), and No Cooperative binding (NO-COOP) models. All of the thermodynamics-based models explored in this work are implemented in GEMSTAT [4], except the AB model, for which the implementation of Sayal et al. was used [26]. Thermodynamics-based modeling of gene expression involves enumerating all "microstates" (henceforth, states) of the enhancer under thermodynamic equilibrium. A state is defined as a configuration specifying the bound or non-bound status of each TFBS. Therefore, an enhancer containing n TFBSs has 2^n states. In GEMSTAT, for each of these states the Basal Transcriptional Machinery (BTM) may be in the bound or non-bound state, making $2^(n+1)$ states. We define "ON" states of the system as those where BTM is bound; other states are called "OFF" states. The expression driven by the enhancer is assumed proportional to the probability of the system being in ON state:

$$Expression \propto Pr(ONstate) = \frac{\sum_{s \in S_{ON}} Q(s)}{\sum_{s \in S_{ON}} Q(s) + \sum_{s \in S_{OFF}} Q(s)} \tag{2.6}$$

where S_{ON} and S_{OFF} denote the set of all ON and OFF states of the enhancer, respectively, and $Q(s)$ is the Boltzmann weight that prescribes the relative probability of state s in equilibrium. (The denominator is the partition function.) For simplicity, we set the constant of proportionality in the above equation to 1.

The Boltzmann weight of state s, $Q(s)$, is calculated as the product of terms representing each molecular interaction in that state; these interactions include TF-DNA interactions at binding sites, BTM-DNA interaction at promoter, TF-BTM interactions representing activation or repression effects of TFs, and TF-TF interactions representing cooperativity or antagonism between proximally DNA-bound TF pairs. See Appendix A, Supplementary Table A.3 for the parameters used in different GEMSTAT models in this study.

TF-DNA interaction: The weight (term contributed to $Q(s)$) of a TF-DNA interaction at site s for a TF t is given by:

$$q(s,t) = [t]k_t exp(LLR(s) - LLR(S_{t,opt})) \tag{2.7}$$

where k_t is a TF-specific free parameter that is learned from the data and other terms are as defined above.

TF-BTM interaction: The weight (term contributed to $Q(s)$) of a TF-BTM interaction is a TF-specific positive constant which is > 1 for activators (making the ON state more favorable than corresponding OFF state) and < 1 for repressors. This TF-specific constant is referred to as the TF's "potency" in the Results section. In this study, repressor's potency is present only in DIR model representing long-range repression. We employed the GEMSTAT's "limited contact" scheme of activation, where at most one bound activator can interact with the BTM in any state.

TF-TF interaction: Any activator-activator or repressor-repressor pair (i.e., DL-DL, TWI-TWI, SNA-SNA, DL-TWI) bound within 50 bp of each other is modeled as interacting, with the weight contributed to $Q(s)$ being a learnable free parameter. Such interactions may be configured to be excluded from or included in the model e.g., when comparing the "COOP" and "NO-COOP" settings of GEMSTAT. Separately, an activator-repressor pair (i.e., DL-SNA, TWI-SNA) bound within 100 bp of each other is modeled as interacting, with a learnable weight (≤ 1). Such interactions represent short-range repression by quenching, following Sayal et al [26], and may be configured to be excluded or included in the model.

BTM-DNA interaction: This interaction is modeled by a single learnable parameter.

2.3.4 Activation by Binding (AB) Model

We used the implementation of Sayal et al. [26] for the AB model. Here, the bound or non-bound status of the BTM is not part of the state definition, and expression is assumed proportional to the probability of states with at least one bound activator that is not repressed by a bound TF nearby. The Boltzmann weight of a state is defined as the product of terms representing TF-DNA and TF-TF cooperative interactions, defined as in GEMSTAT. Terms involving BTM interactions are not part of the model and in particular TF-BTM interaction parameters that capture activating and repressive influences in the GEMSTAT model are not included. See [26] for details. We used the "binned" interaction scheme of their model, using one bin and setting the ranges for TF-TF interactions to match those of GEMSTAT (50 bp for cooperative interactions, 100 bp for short-range repression).

2.3.5 Alternative repression mechanisms in GEMSTAT

In the GEMSTAT model used for testing the AP mechanism of activation, repression is modeled by short-range ($\leq$ 100 bp) activator-repressor interactions (DL-SNA, TWI-SNA). This repression mechanism is referred to as "Q" (for "quenching"). An alternative to this is the "Neighborhood Remodeling" (NR) mechanism of short-range repression, where any repressor site may be in one of three possible states (rather than two): "non-bound", "bound-only" and "bound-effective" [4]. In the bound-effective state, the bound repressor modifies its neighborhood on the DNA such that the neighboring chromatin becomes inaccessible for other TFs to bind. This modification is assumed to occur within a fixed distance d_r from the bound repressor site. States where a repressor site is in the "bound-effective" state and another site (for any TF) within d_r distance is in the bound state are considered invalid. The "bound-only" state is akin to the usual "bound" state of a site, with no restrictions on possible states of neighboring sites. A site in the bound-effective state contributes an additional factor β_r (a TF-specific free parameter) to the Boltzmann weight of the overall state; a higher value of β_r (> 1) leads to lower fractional occupancy of proximal activator sites, thus achieving greater repression. In this study we used a range parameter of d_r =100 bp. A third alternative to the “Q” and “NR” mechanism is the “DIR” (for “direct” repression) mechanism, modeled by a repressor-BTM interaction regardless of distance of repressor binding site from a bound activator site, thus making this a “long-range” repression mechanism [4].

2.3.6 CoNSEPT model

CoNSEPT (Convolutional Neural Network-based Sequence-to-Expression Prediction Tool) is a neural network model that predicts the enhancer activity as a function of enhancer sequence and TF concentration levels. The model is parameterized by user-provided PWMs (motifs) representing TF binding preferences.

First, the enhancer sequence (of length L) is scanned with user-defined PWMs to score the presence of each motif along the enhancer. The scanning module computes the complementary sequence (negative strand) of the input enhancer and converts both strands into a one-hot encoded representation by replacing each nucleotide (A, C, G, or T) with a 4-dimensional vector as follows:

$$A = [1, 0, 0, 0], \qquad C = [0, 1, 0, 0], \qquad G = [0, 0, 1, 0], \qquad T = [0, 0, 0, 1] \tag{2.8}$$

User-defined PWMs are formatted into a set of $K \times 4$ matrices, m_t for each TF t, representing the probability of each nucleotide appearing at each position of the TF’s binding site of length K. In this study, we used $K = 10$, and to do so we had to expand the TWI and SNA PWMs we obtained from Sayal et al. [26] by three and two bases, respectively, with a probability of 0.25 over A, C, G, and T.

Both encoded strands are then scanned with each motif using a convolution operation:

$$b_+^t = S_+ * m_t \tag{2.9}$$

$$b_-^t = \sim (\sim S_- * m_t) \tag{2.10}$$

$$\tag{2.11}$$

where b_+^t and b_-^t represent the binding score profiles of TF t on the positive and negative strands, respectively, S_+ and S_- denote the positive and negative encoded strands. Moreover, $\sim (.)$ represents a “flip” operation that reverses the sequence. The flip operation mimics how the negative strand is scanned for motif presence in previous work [4, 8, 26, 34], and as far as we know, this is the first time it has been included in a neural network model of sequence-function.

Next, for each TF t, the positive and negative binding score profiles, represented by an $\dot{L} \times 2$ matrix ($\dot{L} = L - K + 1$), are passed to a 2-dimensional max-pool layer that extracts the strongest binding site score from the two strands within a window of a certain width. Since we have three TFs (DL, TWI, and SNA) in this study, this step gives us three $\ddot{L}$-dimensional vectors B^t, where $\ddot{L} < \dot{L}$ is the reduced length due to max-pooling. These vectors are then integrated with the TFs' concentration values to obtain "occupancy" vectors F_t:

$$F_t = B^t \times [t] \tag{2.12}$$

where F_t is the vector representing occupancy of TF t along the enhancer, $[t]$ denotes the concentration of TF t in a particular cellular context, and "×" represents element-wise product.

Next, CoNSEPT incorporates user-specified prior knowledge of TF-TF interactions. To this end, for a specified interacting pair, (t_1,t_2), an $\ddot{L} \times 2$ "TF pair feature matrix" is constructed by stacking F_{t_1} and F_{t_2}. The specified interactions may correspond to homotypic or heterotypic cooperativity or short-range repression. Each TF pair feature matrix is passed to a separate 2-dimensional convolutional kernel that moves along the enhancer length and captures the short-range patterns in occupancies, producing an $\dddot{L}$-dimensional vector Ω_i ($\dddot{L} < \dot{L}$ is the reduced length due to convolution):

$$\Omega_i = (F_{t_{i,1}} \& F_{t_{i,2}}) * \kappa_i \tag{2.13}$$

where i corresponds to a specified TF-TF interaction between $t_{i,1}$ and $t_{i,2}$, κ_i denotes the convolution kernel for this interaction and "&" is the stacking operation. The outputs Ω_i of all convolutional kernels are stacked into a $\dddot{L} \times N$ matrix Ω, where N denotes the total number of user-defined TF-TF interactions.

In this study, we used DL-DL, TWI-TWI, and SNA-SNA interactions, consistent with the homotypic cooperativities in COOP model, also DL-TWI interaction, consistent with the heterotypic cooperativity in COOP model, and SNA-DL and SNA-TWI interactions, consistent with the short-range repressions in COOP model. The output of the convolutional kernels, Ω, is activated by a non-linear function. We also tested passing this activated output into two additional convolutional layers with different number of kernels and activated by a non-linear function after each layer to capture longer-range interactions. The activated output of the last convolutional layer goes into a dropout layer that is widely used for regularizing neural network models [48]. We did not use dropout in the previous layers to maintain any positional feature on the enhancer that might contribute to short-range and long-range regulations. The output of the dropout layer is passed to a non-overlapping max-pool layer that extracts the strongest signals. Finally, the output of this pooling layer is linearly combined through a fully connected layer and goes into a final activation function that outputs the expression value. For this last activation function, we tested sigmoid and tanh functions suitably modified to ensure positive expression values upper-bounded by one. We use the following naming conventions to fully demonstrate CoNSEPT's architecture:

BS: TF binding-site scanning module
$PFC_{\alpha,\beta}$: stacking of occupancy vector of each pair of TFs and passing the resulting TF pair feature matrix to a 2-dimensional convolutional kernel of size $(\alpha, 2)$ and stride $(\beta, 2)$. This unit generates the output Ω described above. In this study we used $\alpha = \beta$; i.e. non-overlapping convolution.
$P_{\gamma,\delta}$: a max-pool layer of size γ and stride δ
CN_k: a 1-dimensional convolutional layer with k kernels of size 3 and stride 1.
FC: a fully connected layer
DR_p: a dropout layer with probability of p
LN: a layer-normalization layer [49]
σ: an activation function
$\times[TF]$: multiplication by TF concentration values to obtain occupancy as described above

Architecture: $BS - LN - P_{\gamma,\delta} - \times[TF] - PFC_{\alpha,\alpha} - \sigma_1 - \{CN_{k_1} - \sigma_1 - LN\}_c - DR_p - P_{3,3} - FC - \sigma_2$

Above, $\{.\}$ represents an optional block of convolutional kernels followed by activation and normalization and c denotes the number of blocks used in the model. Note that for $c = 0$, the additional convolutional layers inside the braces are not used. In this study, we tested 2016 different models for various settings of $\gamma, \delta, \alpha, \sigma_1, c, k_1, k_2, p,$ and σ_2 and selected the best model based on the performance on a validation data set. See Appendix A, Supplementary Table A.4 for the parameter settings tested.

2.3.7 Training CoNSEPT

To train a CoNSEPT model we find the settings of parameters θ that minimize the mean squared error between the model output ($\hat{E}$) and the ground-truth expression (E) over the training data:

$$\theta = arg\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} (E_i - \hat{E}_i)^2 \tag{2.14}$$

where N is the total number of training samples, each representing a different combination of enhancer and cellular context (bin along DV axis). For optimization, we employed a stochastic gradient descent algorithm using Adam optimizer [50] for 1000 epochs and a batch size of 20. The learning rate of the gradient descent was scheduled to be decreased during the training. An ensemble of 2016 CoNSEPT models with different hyperparameter settings was constructed and the optimal hyperparameter setting was selected based on training and validation set performances (see the Results section).

2.3.8 Synthetic constructs for evaluation of DL-TWI synergistic activation

In order to guide the training of CoNSEPT models towards capturing the cooperative activation by DL and TWI, we used data from Shirokawa et al. [47]. We constructed a DNA sequence mimicking that tested by Shirokawa et al., consisting of a TWI and a DL binding site located 6 bp apart on a construct of length 35 bp with a sequence shown in Figure 2.2.

5' NNNNNNNAACATATGAANNNNNNGGGAAAATCCNN 3'

TWI binding site **DL binding site**

Figure 2.2: Schematic of the DNA sequence used in this study for in-silico evaluatation of DL-TWI synergistic activation ("N" denotes a dummy base). This DNA sequence is similar to the synthetic constructs previously tested by Shirokawa et al. [47].

The DL binding site is slightly different than that used by Shirokawa et al. [47] since we used the consensus DL binding site implied by the PWMs we obtained from Sayal et al. study [26]. Also, the TWI site is four bp longer than that in Shirokawa et al. [47] due to the padding of TWI PWM in our study.

Similar to the experiments of Shirokawa et al. [47], we next created a sequence with five consecutive repeats of the above block. Since CoNSEPT was trained on enhancers of length 635 bp (See Appendix A, Supplementary Note A.1.2), we expanded this construct of length

175 bp to 635 bp by adding dummy bases ("N") at both ends. To eliminate potential biases, we repeated this expansion with 80 different random distributions of dummy bases at the two ends; therefore, we obtained 80 different constructs of length 635 bp that only differ in their distribution of dummy bases at the two ends.

Next, we selected the hyper-parameter settings of CoNSEPT with the highest validation set performance (referred to as "best-validated" CoNSEPT model) among the ensemble of 2016 settings described above. Using these hyper-parameter settings, we re-trained an ensemble of 200 CoNSEPT models on the training data with different random initializations of free parameters. We examined the prediction of each of these 200 trained models on the synthetic constructs defined above, assuming DL and TWI concentrations in a 3:1 expression ratio (similar to that in experiments of Shirokawa et al. [47]) as well as in the presence of only one of the TFs and a basal level in the absence of both TFs. For each model, the predicted expression values were averaged over the 80 constructs to obtain four expression values corresponding to the four conditions of TF presence/absence – Basal, DL, TWI, DL & TWI.

2.3.9 Synthetic constructs for evaluation of distance-dependent interactions

To characterize distance-dependent interactions learned by CoNSEPT models, we examined their predictions on additional synthetic constructs containing pairs of DL-TWI, DL-SNA, TWI-SNA or DL-DL binding sites at progressively increasing separations (from 5 bp to 195 bp), located at a random location in a 635 bp long sequence. Thus, a collection of 3900 synthetic enhancers was tested, corresponding to 39 different inter-site spacings and 100 different locations of the site pair in the enhancer. The TF binding sites were set to consensus sites of corresponding PWMs and the remaining bases of the enhancers were set to the dummy base "N" (see above). It is worth mentioning that we did not evaluate TWI-TWI distance dependent interactions since the training enhancers contain a maximum of two binding sites for this TF (Figure 2.1A), potentially preventing the reliable learning of their distance-dependent interactions.

We used the trained CoNSEPT models to predict expression driven by each synthetic construct with relative levels of DL, TWI, and SNA set to of 0.4, 0.3, and 0.4 respectively (reflecting the 6th of the 17 "bins" along the V-D axis; this bin was selected as it represents the dorsal boundary of SNA expression; Figure 2.1B). For each TF pair at a specific inter-site spacing, we averaged the predicted expression over all 100 constructs with different placements of the TF pair. We evaluated the GEMSTAT model (NR/COOP) on the same constructs containing pairs of TFs; however, we replaced the dummy bases "N" with random draws from the set of four nucleotides (A, C, G, and T) with equal probability.

2.3.10 Data and code availability

We used the publicly available dataset (including enhancer sequences, gene expressions, TF levels, and PWMs) first published and analyzed by Sayal et al. [26]. The GEMSTAT version allowing "limited contact" scheme used in this study is available at: `https://github.com/PayamDiba/Manuscript_tools/tree/main/GEMSTAT_direct_limited`. CoNSEPT is freely available as a python package at: `https://github.com/PayamDiba/CoNSEPT`.

2.4 RESULTS

2.4.1 Linear models provide a good phenomenological baseline

We first trained a linear model (LM) to assess the baseline explanatory power of statistical models on the Sayal et al. data set. In these models [41], expression driven by an enhancer is the sum of contributions from all TFs (Figure 2.3A). The contribution of a TF is the product of the TF's concentration, its binding site strength at the enhancer estimated using its Position Weight Matrix (PWM) (see Methods) and a tunable parameter that represents the TF's regulatory strength and direction (activator/repressor). We also tested a generalized linear model (GLM) [32, 51], where expression is a sigmoidal function of the sum of TF contributions (see Methods), such that the response of a gene is less sensitive to the concentration of its regulators at very low or high concentrations. Both LM and GLM have only one free parameter per TF and are the simplest of the models evaluated here. To impose our prior knowledge of TF roles, the trainable weights for DL and TWI were restricted to positive values and the weight for SNA was restricted to negative values. For LM we computed the globally optimal parameters (on the training set) while for GLM an ensemble of 100 models was trained.

Figure 2.3B shows the train and test errors (RMSE) and correlation coefficient between real and predicted expression profiles, for LM as well as the ensemble of GLM models. The GLM model with the smallest train error (henceforth is referred to as "best-fit" model) was selected from the ensemble and its performance was compared against the LM model (Figure 2.3C). The GLM model clearly shows better fits than LM model in terms of error (RMSE)

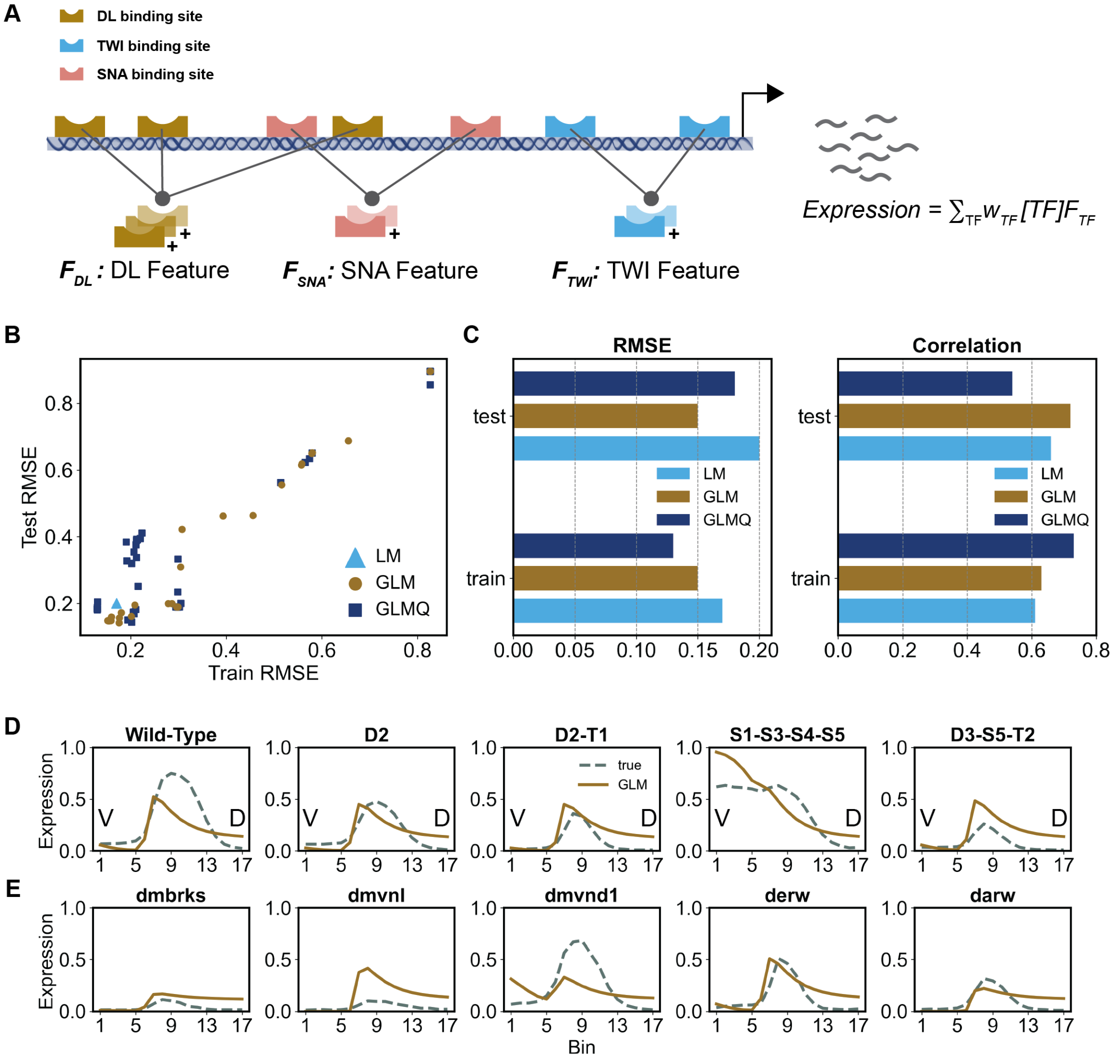

Figure 2.3: Linear and generalized linear models of gene expression. (A) Schematic representation of linear model (LM). Each TF's "feature" is obtained by aggregating the strengths of all of its binding sites in the enhancer. Expression is modeled as the weighted (w) sum of all TF features multiplied by their corresponding concentration profile ([.]). (B) Train and Test RMSE for LM and the ensemble of GLM (generalized linear model) and GLMQ (GLM with quadratic terms) models. (C) RMSE and correlation scores of LM and best-fit (smallest train RMSE) GLM and GLMQ models in train and test data. (D) Predictions of best-fit GLM model on select train enhancers (brown curves) shown versus true expressions (blue curves). Left-most panel corresponds to the Wild-Type enhancer and the remaining panels correspond to perturbed enhancers titled by the site deletion they represent (e.g., "D2-T1" corresponds to simultaneous deletion of DL site D2 and TWI site T1). (E) Predictions of best-fit GLM model on select test enhancers (brown curves) shown versus true expressions (blue curves).

and correlation on both train and test data sets. Examining its predictions more closely (Figure 2.3D,E), we find that the GLM model often correctly predicts the main features of an enhancer's readout, e.g., location of expression peak along the axis, but is also prone to predicting excessive expression at the dorsal end, which is due to inaccurate estimation of the basal transcription level (the intercept term in the linear function).

The above models consider each enhancer as a "bag of sites" [52] where multiple TFs and TF sites contribute additively to the regulatory output. In an attempt to capture any non-additive contributions from pairs of TFs, as is believed to arise from cooperative activity [23], we extended the GLM model ("GLMQ") to include quadratic terms that represent products of TF concentrations (see Methods). Though the training RMSE improved compared to GLM due to the additional parameters, the test RMSE and test correlation deteriorated substantially (Figure 2.3C). Notably, GLMQ shows a negative trained coefficient corresponding to DL-TWI interaction (see Appendix A, Supplementary Table A.2 for trained parameters) which contradicts the previous findings of the synergistic activity between these two activators reported in literature [47]. In summary, cooperative interactions were not reliably learnt by simply adding quadratic terms to the generalized linear model. This is not surprising, since this approach to modeling TF cooperativity does not consider dependence of the interactions on inter-site distances [14].

2.4.2 Thermodynamics-based models reveal biochemical mechanisms

We next employed thermodynamics-based models of gene expression [4, 26] that explicitly incorporate biochemical mechanisms of gene regulation, including TF-DNA binding affinities, activation and repression mechanisms, synergistic activity of multiple TFs, and distance-dependent interactions between TFs bound at proximal sites. Different models, representing different combinations of mechanisms, were implemented as variations of the same thermodynamics-based sequence-to-expression modeling framework, called GEMSTAT [4]. Evaluation of different GEMSTAT model variations was performed with the same parameter fitting techniques, and the compared models typically shared many parameters, differing only in the desired mechanistic aspect, making their comparison more controlled.

Activation Mechanism

We first examined and compared two different activation mechanisms that have been implemented in past modeling studies. In both mechanisms, activation is due to binding of activator TFs to the enhancer and stronger binding leads to greater activation. In one class of models, e.g., that used by Sayal et al. in their original analysis of the data set [26], the contribution of an activator binding site depends only on the binding affinity of the site for its cognate TF and the TF's concentration. (For now, we ignore effects of other binding sites on this site's contribution.) These two factors together determine the fractional "occupancy" of the site by the TF, and its contribution to expression depends solely on its occupancy. In the second class of models, the site's contribution additionally depends on the particular TF's "potency", which may be different for different TFs. That is, two sites with the same fractional occupancy by their respective TFs may contribute to the activation to different extents. This adds additional freedom to the model, in the form of one extra tunable parameter per TF. The mathematical formalisms of the two mechanisms outlined here, called AB (Activation by Binding) and AP (Activation with Potency), are illustrated in Figures 2.4A,B and explained in Methods.

We sought to determine if the AP and AB models differ in their ability to explain the data set. In the AP model, an activator's potency was modelled by stipulating an interaction between a DNA-bound activator and the basal transcriptional machinery (BTM), as in GEMSTAT. (This interaction is represented by a single free parameter per TF.) Other mechanistic details such as cooperative binding, short-range repression, etc. are identical between the two models. For a rigorous comparison, we trained ensembles of 100 AB models and 100 AP models on the train set of 38 enhancers and evaluated them on the test set of 13 enhancers. Figure 2.4C shows the accuracy of all models on train and test sets and Figure 2.4D reports on the best-fit model (smallest train error) in both ensembles. In Figure 2.4D we note that the AP model achieves lower error on the training set, which is not surprising given that it has additional parameters (TF's potency and basal transcription level). More importantly, it achieves a substantially lower error (RMSE of 0.16 versus 0.32) and higher correlation (correlation of 0.59 versus 0.20) on the test set than the AB model, where the additional parameters do not confer an advantage. This performance gap on test data is apparent at the ensemble level also (Figure 2.4C), suggesting that it is not an artifact of the optimization step. For both models, the test error values are overall higher than training error, but this is likely to be because the test enhancers are biologically more distinct from the training enhancers than the variation within the training set. The gap between training and test errors is much smaller for the AP model, which has more free parameters, which is contrary to what we would expect if the gap was primarily due to overfitting.

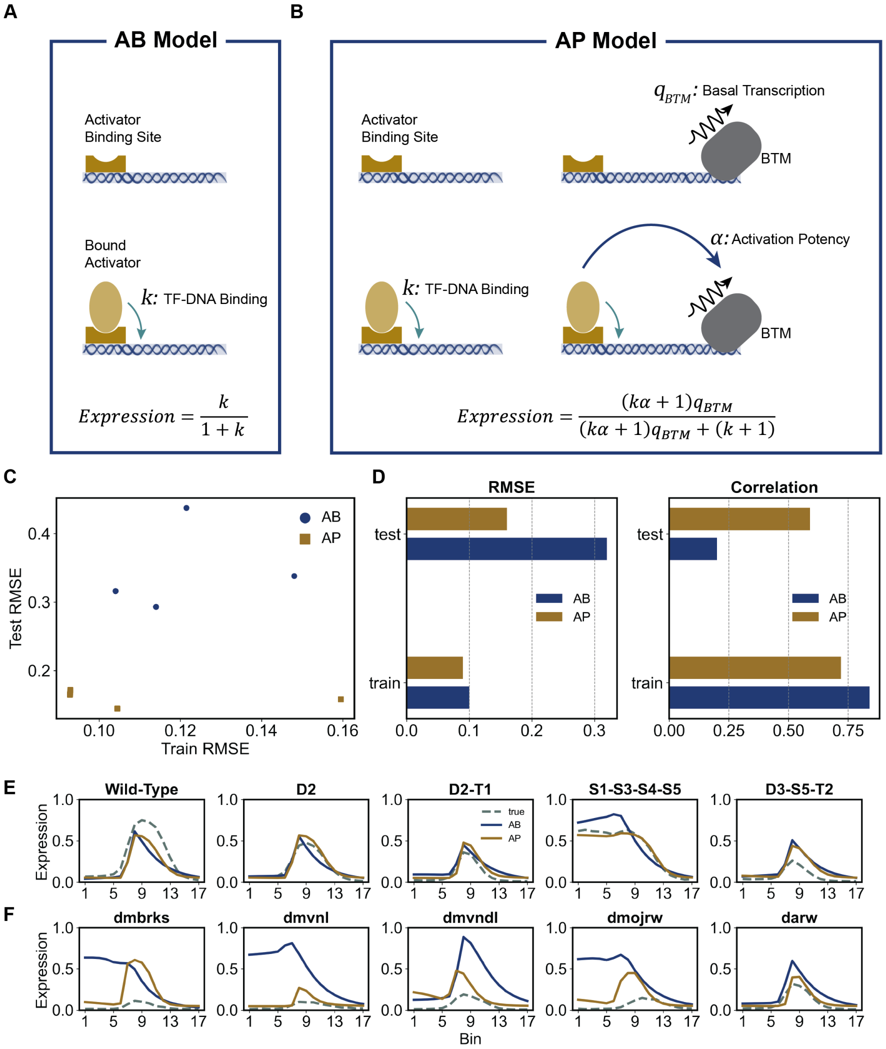

Figure 2.4: Evaluations of activation mechanisms. (A) Schematic representation of AB model for an enhancer containing only one binding site for an activator TF. Shown are the two possible configurations for this enhancer depending on whether the activator is bound or not. The statistical weights (relative probabilities) of the two configurations are k (bound) and 1 (not bound), and expression is proportional to probability of the bound configuration. (B) Schematic representation of AP model for an enhancer containing only one binding site for an activator TF. Shown are the four possible configurations for this enhancer depending on whether the activator and the BTM are bound (or not). Expression is proportional to total probability of the two configurations in which BTM is bound. (C) Train and Test RMSE for the ensemble of 100 AB and 100 AP models. (There is extensive overlap of models (points) in each ensemble.) (D) RMSE and correlation scores of best-fit AB and AP models in train and test data. (E) Predictions of best-fit AB and AP models on select train enhancers (solid curves) are shown along with true expression profiles (dashed curves). (F) Predictions of best-fit AB and AP models on select test enhancers (solid curves) are shown along with true expression profiles (dashed curves).

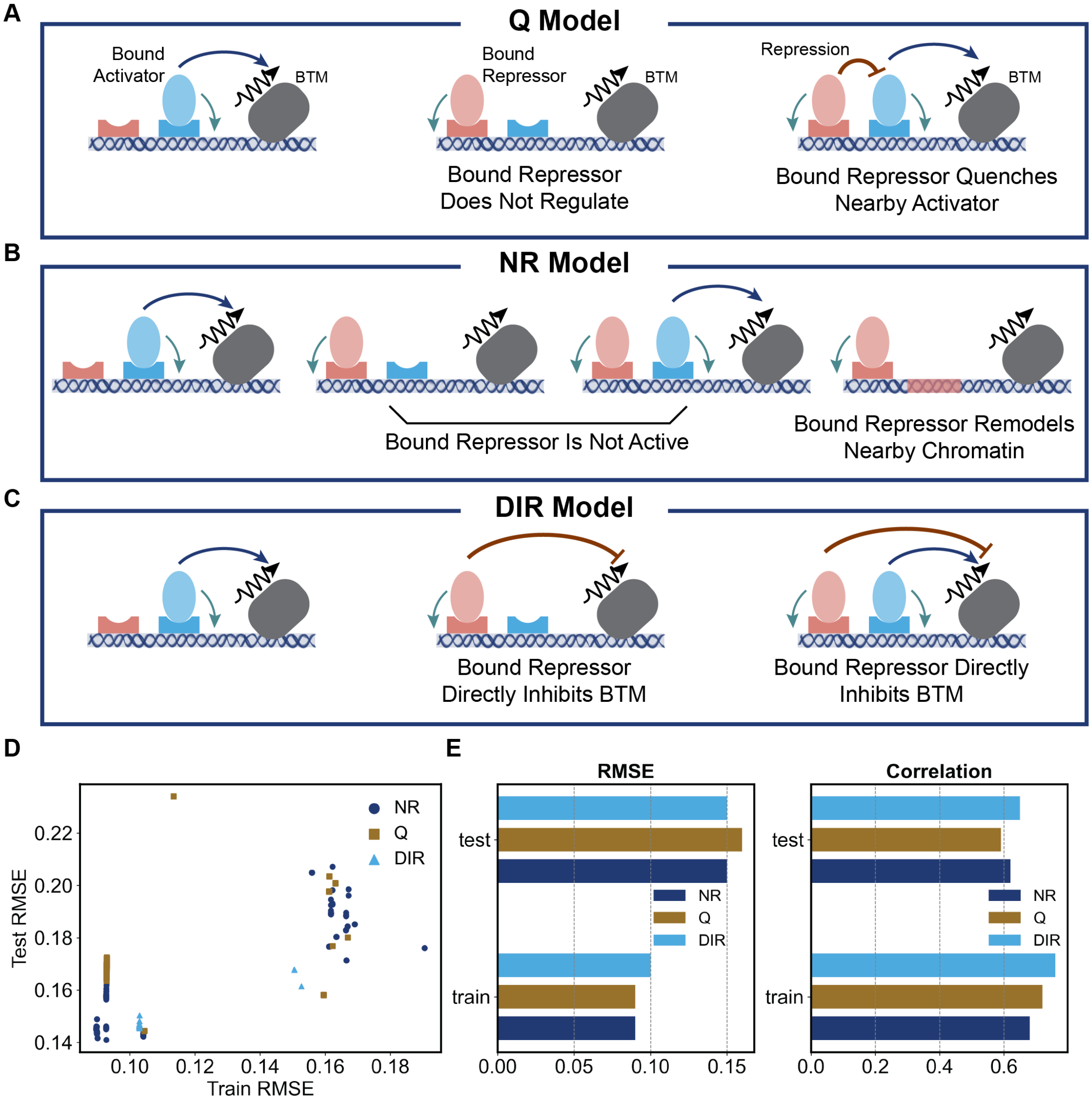

Figure 2.5: Evaluations of repression mechanisms. (A) Schematic representation of Q model that employs short-range quenching repression mechanism. A bound repressor diminishes the activity of nearby bound activators. (B) Schematic representation of NR model that employs short-range neighborhood remodeling repression mechanism. A bound repressor can be in either active or inactive state. An inactive bound repressor does not interfere with the binding of activators to nearby regions or with the activity of the nearby bound activators, while an active bound repressor prevents activators from binding to nearby regions. Any configuration with an active bound repressor and an activator bound nearby is considered invalid. (C) Schematic representation of DIR model that employs direct repression mechanism. A bound activator directly diminishes the activity of recruited BTM but does not interfere with the binding of activators. (D) Train and Test RMSE for the ensemble of 6000 models for each of Q, NR, and DIR models. (E) RMSE and correlation scores of best-fit NR, Q, and DIR models in train and test data.

A few illustrative examples of our evaluations are shown in Figure 2.4E,F, where each panel compares AP and AB model predictions to the real expression profiles. While both models exhibit similar accuracy on training enhancers (Figure 2.4E), the AB model predicts ectopic expression in the ventral region (bins 0-5) for test enhancers 'dmbrks' (D. melanogaster enhancer of gene brk) and 'dmvnl' (D. melanogaster enhancer of gene vn), while the AP model correctly predicts the neuroectodermal peak (bins 7-10) of expression for all shown test enhancers (Figure 2.4F, also see Appendix A, Supplementary Figure A.2). In light of the above observations, we infer that the data set supports the AP model over the AB model, arguing for separate "potency" for each activator TF, beyond its DNA-binding strength, as an important aspect of the underlying activation mechanism.

Repression Mechanism

Previous studies have shown that SNA is a "short-range" repressor whose effect is mediated by a co-repressor named CtBP [53, 54]. CtBP can bind to histone deacetylases, which in turn causes DNA to wrap around the histone more tightly and prevents nearby TFs from binding to DNA [55, 56]. It implies that a DNA-bound repressor co-bound with CtBP is likely to be the only recruited TF within a small window ($\sim$ 100 bp) around its binding site. Such a mechanism for short-range repression is implemented in GEMSTAT [4], and we will refer to it as "neighborhood remodeling" ("NR", Figure 2.5B) below. In this model, the bound repressor makes the neighboring chromatin (within 100 bp in our tests) inaccessible for activators to bind at. An alternative formulation of short-range repression is the "quenching" mechanism ("Q", Figure 2.5A) [7, 26], which states that a bound repressor will diminish or "quench" the effectiveness of an activator bound nearby; in the thermodynamic model this is achieved by a decreased equilibrium probability of the configuration where both the activator and repressor are bound (see Methods). We tested if the data set can discriminate between the NR and Q models of short-range repression using their implementations within the GEMSTAT framework. We also tested a model with so-called "direct" repression ("DIR", Figure 2.5C), which, unlike NR and Q, is not a short-range mechanism. Here, the regulatory effect of a bound repressor is due to interactions with the BTM (similar to how activation is modeled in the AP model), and the binding site does not have to be within a short range of any activator binding site for its repressive effect to materialize. Although literature evidence points to short-range repression by SNA [7], we considered it worthwhile to test the direct repression model as a simple phenomenological baseline against which more realistic short-range repression models may be compared in light of the available data. DIR and NR are the least complex of the three models, modelling repression using three free parameters for

SNA (DNA-binding, repression potency, and homotypic cooperativity), while Q uses four free parameters (see Methods).

Figure 2.5D compares the performance of trained ensembles of NR, Q and DIR models and Figure 2.5E summarizes the performance of the best-trained model in each ensemble. We found both short-range repression models (NR and Q) to yield comparable fits and predictive ability, with the NR model being slightly better in terms of both correlation (0.62 vs 0.59) and RMSE (0.16 vs 0.15) on test enhancers (with one less free parameter than NR model). Interestingly, predictions of the DIR model are as accurate as NR – they show the same RMSE and better correlation on test enhancers, and a significantly better correlation (0.76 vs. 0.68) on train enhancers while using the same number of free parameters. Moreover, correlation values on test enhancers are substantially better with the DIR model than with the Q model. This suggests that in addition to short-range repression, SNA may have long-range repressive effect on transcription of *rho*. We revisit this hypothesis below when we discuss the predictions of our Neural Network-based model.

Cooperative Activation Mechanisms

An important mechanism studied in the context of enhancer function is that of cooperative DNA binding by multiple TFs at proximally located binding sites [14], which results in a synergistic effect greater than the sum of the individual site contributions [23, 47, 57]. To test if such effects are reflected in the data, we compared a version of GEMSTAT that models cooperativity between activators ("COOP" model, Figure 2.6B) with one that does not ("NO-COOP", Figure 2.6A). To implement cooperativity, GEMSTAT includes TF-TF interaction energy terms in configurations where two TFs are bound within a certain distance (set to 50 bp here) of each other. Such terms were presumed for DL-DL, TWI-TWI, and DL-TWI interactions, each represented by a free parameter, in the COOP model. (Both tested versions use the NR model for repression and include homotypic interaction for the repressor SNA.)

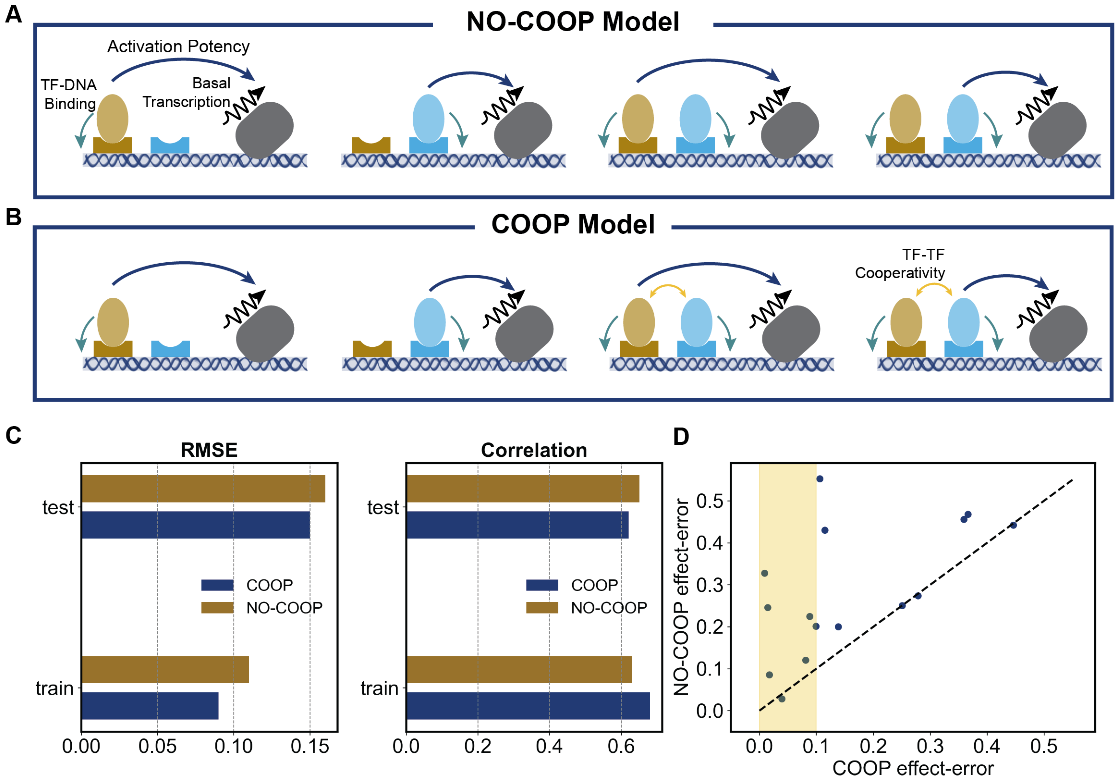

Figure 2.6: Evaluations of synergistic activation through cooperative binding. (A,B) Schematic representation of the difference between COOP (B) and NO-COOP (A) models. Shown are four of the possible configurations where at least one activator is bound and the BTM is bound. In the COOP model, the two configurations where both activators are bound have favorable interaction between them, resulting in higher probability of these configurations and hence increased expression. (Both COOP and NO-COOP employ short-range neighborhood remodeling repression mechanism; not shown.) (C) RMSE and correlation scores of best-fit COOP and NO-COOP models on training and test data. (D) Effect-error of best-fit COOP versus NO-COOP models on the perturbed enhancers containing deletions of activators' binding sites (similar to the enhancers shown in Figure 2.1D). Effect error is defined as the absolute difference between the predicted and true effect of the perturbation; effect of a sequence perturbation (activator site deletion) is defined as decrease in peak (bin 8) expression due to the perturbation. Highlighted region shows perturbed enhancers for which COOP has a small effect error (close to the average training RMSE).

We trained ensembles of models for COOP and NO-COOP separately. The best-trained COOP and NO-COOP models show train error of 0.09 and 0.11 and test error of 0.15 and 0.16 respectively (Figure 2.6C), thus providing some evidence in favor of the former. To tease apart their differences further, we used a complementary evaluation metric: the ability of trained models to predict the effects of activator site deletions (Figure 2.1D). We defined a model's "effect-error" as the difference between the predicted effect of the site deletion(s) represented by an enhancer and the real effect captured in the data set.

(Here, “effect” is calculated in the same way as in Figure 2.1D, i.e., the difference in peak expression between wild type and variant enhancer.) Figure 2.6D compares the effect-error of the best-trained COOP and NO-COOP models on each enhancer shown in Figure 2.1D, where one or two activator sites in the wild type *rho* enhancer have been mutagenized. For all enhancers, the effect-error in COOP is smaller or as low as that in NO-COOP. In particular, in the highlighted region where the effect-error of the COOP model is relatively small (smaller than 0.1), we note several enhancers where the NO-COOP model’s effect-error is substantially worse. These results support the hypothesis that cooperative DNA-binding at proximally located activator binding sites plays a significant role in regulatory function of the *rho* enhancer.

2.4.3 CoNSEPT: a neural network model of enhancer function

For our final modeling of the data set, we implemented a model called CoNSEPT (Convolutional Neural Network-based Sequence-to-Expression Prediction Tool), that can accommodate highly non-linear contributions of TF binding sites to overall enhancer function (Figure 2.7). Our primary goal was to test if a convolutional neural network (CNN) model, which does not explicitly incorporate known rules of cis-regulatory encoding, can learn such rules (regulatory mechanisms, including distance-dependent interactions between sites) from the data. This tool first uses pre-determined TF motifs (PWMs) to scan both strands of an enhancer to identify putative binding sites and estimate their strengths (“PWM scores”). It then integrates these strengths with respective TF concentration values to assess each TF’s presence at varying locations in the enhancer, analogous but not identical to occupancy in thermodynamics-based models (see Methods). Next, it assembles the presence scores of each TF pair into a feature matrix, which is passed to a separate convolutional filter to capture short-range interactions between the TF pair. Outputs of these filters are aggregated and passed to the subsequent convolutional layers to capture long-range interactions. Finally, the output of the last convolutional layer is combined linearly to predict the expression driven by the enhancer (see Methods for details).

We used the training data comprising the WT *rho* enhancer and its 37 variants to train CoNSEPT models; eleven of the 13 enhancers in the previously defined test data were used for evaluating trained CoNSEPT models. (Two enhancers were removed from the original test data due to their significant length difference with the training enhancers; see Methods.) We used data on three additional enhancers (variants of the *rho* enhancer, regulated by the same TFs) available from Sayal et al. [26] as a separate “validation set” for tuning hyper-

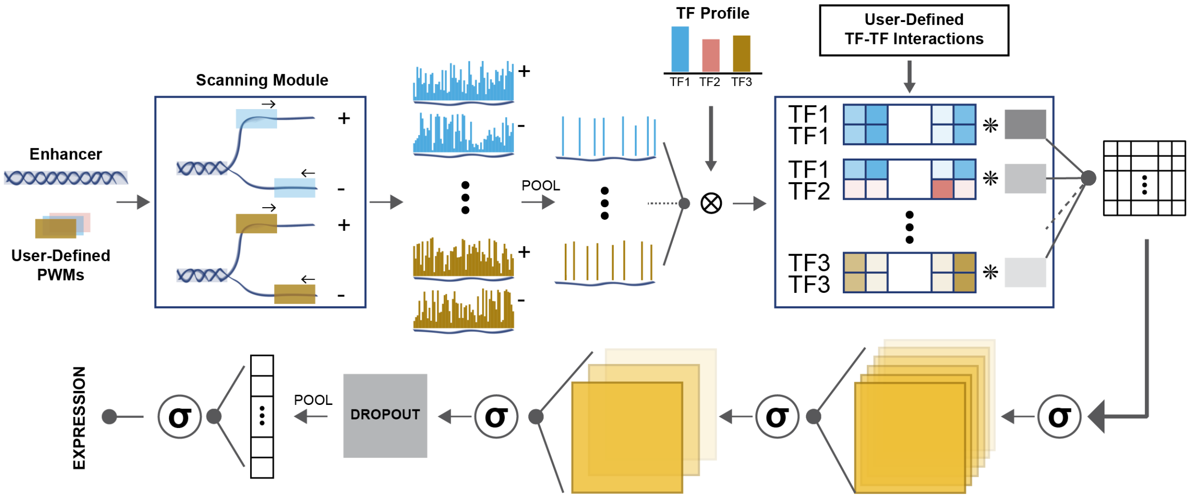

Figure 2.7: Architecture of CoNSEPT. First, the input enhancers are scanned with user-defined PWMs and passed to a pooling layer to extract the strongest matches. Next, TF profiles are integrated (multiplied) with the extracted scores and pair-wise features are built according to the user-defined TF–TF interactions. Each pair-wise feature is passed to a separate kernel capturing short-range regulatory interactions between pair of TFs. (Note: a TF pair may be homotypic.) The concatenated output of interaction convolutional kernels is passed to a sequence of activation functions (σ) and optional additional convolutional layers. The final activated output of the convolutional layers is passed to a dropout layer followed by a fully-connected layer and an activation function to predict the expression.

parameters of CoNSEPT models. Previous applications of neural networks for regulatory sequence interpretation had the benefit of large training data sets [43, 45], and the challenge for us was to train a CNN on a far smaller data set without "overfitting".

An ensemble of CoNSEPT models with varying architectures and hyperparameters (see Appendix A, Supplementary Table A.4 for different settings used) was trained on training data and evaluated on validation data (Figure 2.8A). A subset of the ensemble exhibits better training and validation accuracy than the best-trained GEMSTAT model (NR repression, COOP), and we selected among these the one with the smallest validation error (circled) for further analysis (see Appendix A, Supplementary Table A.5 for the settings of this model). Figure 2.8B compares the RMSE and correlation values for the predictions of this model on train, validation and test data sets with predictions of GLM and GEMSTAT models learned above as the best representatives of the linear and thermodynamics-based models. CoNSEPT predictions on the test set are at least as accurate as GLM and GEMSTAT in terms of RMSE, and moderately better in terms of correlation values (Examples of its improved prediction are shown in Figure 2.8C,D, also see Appendix A, Supplementary Table A.6 for p-values of Spearman's correlation coefficients, which reveal that prediction of both CoNSEPT and GLM significantly correlate with the true expressions for 10 out of 11 test enhancers.). Its high accuracy values on the training set are not surprising given its higher number of free parameters (1537, 11, and 4 for CoNSEPT, GEMSTAT and GLM models respectively), but its competitive test accuracy suggests that despite the vastly greater model complexity and limited training data, CoNSEPT model fitting does not suffer from overfitting any more than NR model does.

2.4.4 CoNSEPT learns distance-dependent interactions between TF binding sites

We saw above that CoNSEPT learns to predict expression from sequence with almost no pre-determined notion of cis-regulatory grammar such as activation, repression or pairwise interactions between sites. We next interrogated the trained model to determine if biophysically plausible cis-regulatory rules underlie its highly parameterized non-linear form. We were particularly interested in whether it learns distance-dependent pairwise interactions between TF binding sites. Recall that the linear models above do not allow such interactions and our thermodynamics-based formulations must be "hard-coded" with a particular form of distance-dependent interaction.

Prior to interrogating the CoNSEPT model, we refined it based on an additional data set that specifically captures synergistic interactions between DL and TWI binding sites. Shirokawa et al. [47] tested a synthetic enhancer consisting of 5 repeating blocks of a sequence with DL and TWI binding sites six bp apart (Figure 2.8E); a reporter assay was used to show that expression driven in the presence of both DL and TWI (in 3:1 ratio) is greater than the sum of expression by these TFs separately, suggesting synergistic activation by these two TFs [47]. Our model refinement demanded that the CoNSEPT model with architecture and hyper-parameter setting as determined above (circled model in Figure 2.8A) additionally recapitulate the synergistic activation encoded by the construct of Shirokawa et al. (We also required expression driven by TWI alone to be at least as high as that driven by DL, another observation made by the authors.) This refinement step yielded three CoNSEPT models (parameterizations), whose predictions on the synthetic enhancer of Shirokawa et al. are shown in Figure 7F. (See Methods for details of refinement steps.)

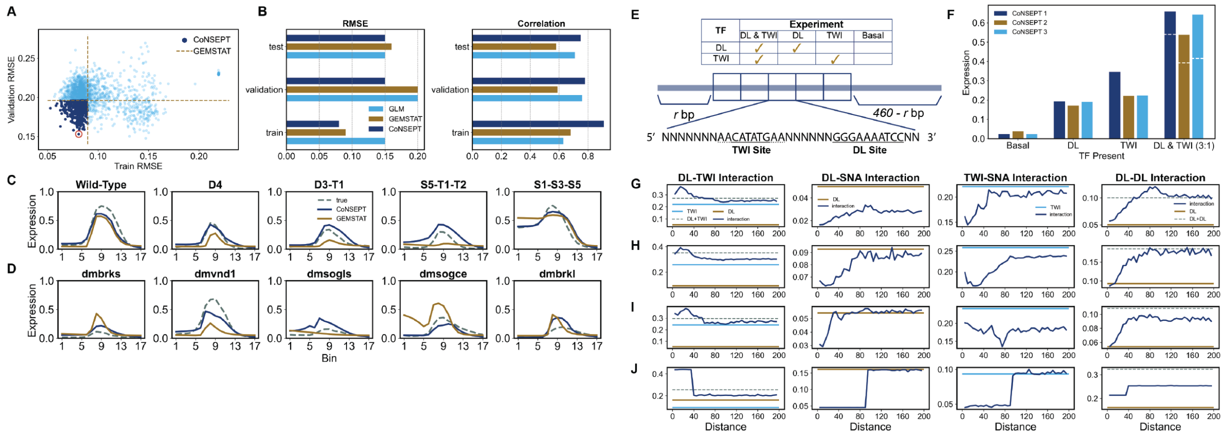

Figure 2.8: (A) Train and validation RMSE for the ensemble of CoNSEPT models. Best-fit GEMSTAT (NR, COOP) model was used as a baseline (dashed lines). Among the CoNSEPT models with a better performance than the baseline, we selected the one with the smallest validation RMSE (circled). (B) RMSE and correlation scores of the selected CoNSEPT model on training, validation and test data. (C,D) Predictions of the selected CoNSEPT model and the baseline GEMSTAT model on select train (C) and test(D) enhancers for which CoNSEPT improves predictions. (E) *In silico* synthetic construct adopted from [47] consists of 5 repeating blocks each containing one DL and one TWI binding site with a length of 35 bp. The 5-block construct with a length of 175 bp is randomly padded by r (random variable) "dummy" bases on the left and $460 - r$ dummy bases at the right end to get a longer construct of length 635 bp consistent with the length of train enhancers. The top table shows the four trans contexts for which this construct's expression was predicted: "DL & TWI" where both TFs are present with a 3:1 concentration ratio, "DL" where TWI concentration is zeroed out, "TWI" where DL concentration is zeroed out, and "Basal" where neither of the factors are present. (F) Predictions of the three selected CoNSEPT models on the synthetic construct shown in E. The white dashed lines on DL & TWI bars correspond to the sum of the expression driven by DL and TWI individually. (G-J) Predictions of the three selected CoNSEPT models (G-I) and the baseline GEMSTAT model (J) on constructs containing only one or two TF binding sites with different inter-site spacings (0-200 bp; x-axis). Each column corresponds to a pair of interacting TFs. The dark blue curves show predicted expression on constructs containing two binding sites for the TF pair. The brown and light blue curves show predicted expression on construct with one DL site or one TWI site respectively. Dashed lines show the expression obtained by summing the predicted expression driven by individual factors (DL+TWI in first columns and DL+DL in fourth column).

We next used the three models derived above to predict the expression driven by constructs harboring a single pair of TF binding sites (DL-DL, DL-TWI, DL-SNA, or TWI-SNA) at varying inter-site distances (see Methods). The results, shown in Figure 2.8G-I, reveal the distance-dependent interactions between TF binding sites, as learnt by CoNSEPT models. Firstly, all three models predict synergistic activation by a DL-TWI site pair (first column) over short distances < 40 bp with a maximum effect at $\sim$20 bp inter-site spacing consistent with previous experimental findings [23]. The GEMSTAT model (NR, COOP), shown in the bottom row for comparison, shows a similar trend; indeed, GEMSTAT training required that cooperative binding be of a fixed strength within 50 bp and absent for greater inter-site spacing. On the other hand, the trend was learnt by CoNSEPT models entirely from training data. Notably, the DL-TWI spacing has been shown to be a key element of functional organization of neuroectodermal enhancers [58].

We next examined learnt distance-dependencies for DL-SNA interactions (Figure 2.8G-I, second column). All three models correctly predict SNA as a repressor, with two predicting its effect to be exclusively short-range, decreasing linearly as the distance from DL (activator) site increases from 0 to 40 bp or 80 bp. The third model (CoNSEPT 1, Figure 2.8G) also shows the 80 bp short-range effect but predicts an additional long-range effect that extends to the maximum spacing interrogated. The GEMSTAT model (bottom row) is consistent with a short-range effect, as observed above (Figure 2.5), but its distance-dependence function (range of 100 bp) was hard-coded into the model. Interestingly, our GEMSTAT modeling had also indicated the possibility of longer-range effects of SNA ('DIR', Figure 2.5), akin to that seen with the CoNSEPT 1 model. Similar trends were seen with TWI-SNA interactions, with two of the models predicting a short-range effect (range of 60 or 90 bp) (Figure 2.8G,H, third column) and one model (CoNSEPT 3) also indicating a longer-range effect. The consistent observation of long-range (beyond 100 bp) repressive effects of SNA sites, through CoNSEPT models (DL-SNA as well as TWI-SNA interactions in Figure 2.8G-I) as well as GEMSTAT models (Figure 2.5), suggest that mechanisms other than the documented short-range repression by SNA may be at play in neuroectodermal enhancer function.

Intriguingly, all three CoNSEPT models predict (Figure 2.8G-I, fourth column) that the activation by two DL sites is less than twice the activation by a single site at shorter inter-site spacing (< 50 bp). The GEMSTAT model's predictions are similar (bottom row) – when asked to tune a parameter representing DL-DL interaction (limited to sites within 50 bp), the model learnt an antagonistic interaction. The models mostly did not find evidence of synergistic interaction between DL site pairs at any spacing. In fact, the CoNSEPT models predict that two adjacently placed DL sites drive no more activation than either site alone, while the GEMSTAT model, forced to assume fixed interaction strength within the distance threshold (50 bp), differs in this prediction. We do not know of a plausible candidate mechanism for the prediction of antagonistic DL-DL site interactions at short distances ($<$ 50 bp), and the finding needs more direct confirmation and dissection in the future. We conducted additional analyses to provide an interpretation of what CoNSEPT model learns about overlapping binding sites and about the relationship between TF binding affinities and gene expression (Appendix A, Supplementary Figure A.3 and Supplementary Note A.1.1). These analyses suggest that CoNSEPT learns a non-linear saturating relationship between TF binding site strength and gene expression.

2.5 DISCUSSION

Sequence-to-expression modeling can reveal mechanistic details of an enhancer's regulatory function as encoded in its sequence. The significance of such modeling is well argued in the literature [28] and, with rapidly advancing technology for multiplexed assays of enhancer function [59, 60, 61], a rigorous method for mechanistic inference and cis-regulatory decoding is clearly needed. We were interested in a version of this problem where the model captures variation of function across different enhancers as well as across cellular conditions; the latter requires that the model utilize cellular TF concentrations in making predictions. Fortunately, prior work offers several ways forward, from linear models [41, 51] to thermodynamics-based models [4, 6, 7, 26, 62, 63]. Each of these modeling approaches has been shown to explain one or more expression data sets and reveal useful insights into their underlying biology. Typically, each approach has its own mechanistic assumptions baked into the implementation, with quantitative details of assumed mechanisms being left as trainable parameters. To our knowledge, no study has attempted to fit models of fundamentally different form, i.e., with different mechanistic assumptions (or lack thereof), to the same data set to assess their relative potential to explain the data and provide mechanistic clues. This existing gap was the primary motivation behind our work.

We tested linear and generalized linear models, thermodynamics-based models with varying biophysical assumptions, as well as a newly implemented convolutional neural network on the same data set, evaluating their ability to fit the data and generalize to unseen (but related) enhancers. We found these different modeling approaches to show similar predictive ability overall in terms of the RMSE score, although the neural network model (CoNSEPT) yields higher correlation coefficient on average (Figure 2.8B) compared to other models (See Appendix A, Supplementary Table A.7). We found that linear models, which are simple to implement and use, provide good fits to data but are by construction unable to discover non-linear combinatorial contributions of multiple regulators. On the other hand, equilibrium thermodynamics-based models allow incorporating various hypotheses regarding combined action of multiple regulators in ways that depend on inter-site distance.

In evaluating different thermodynamics-based frameworks [4, 26], we found that modeling the effects of activators via two separate interactions (TF-DNA binding and TF-BTM interactions) provides a substantial benefit in terms of predictive power (Figure 2.4), compared to modeling activator-DNA binding alone, at least within the context of our evaluations. The choice of separating the DNA-binding aspect of a TF's function from its expression activation aspect is not arbitrary – these two aspects are typically encoded in different domains of the protein, and such separation is common when modeling repressor TFs [7, 8]. Activator potency is often attributed to interactions between TF and the basal transcriptional machinery (potentially mediated by co-factors), whose details are expected to differ from one TF to another. Note that we used the implementation of Sayal et al. for the AB model but set its TF-TF interaction rules to a simpler form than those of the original study [26], in order to match those in the AP model (implemented in GEMSTAT). We expect that replicating the more complex settings of interaction rules of Sayal et al. will yield better fits to the data, but make direct comparison (of AP and AB) challenging.

We tested several additional thermodynamics-based formalisms within a common framework (GEMSTAT), using the same parameter training algorithms. We saw evidence of cooperative DNA-binding at proximal sites by TF pairs (Figure 2.6), as has been reported extensively in the literature [14], including through thermodynamics-based models [4, 64]. Importantly, DL-TWI synergistic action has been experimentally observed [23, 47]. When testing three different modes of repression, we were surprised to note that a "direct repression" mechanism, where a bound repressor's regulatory contribution does not depend on how far its binding site is from an activator's site, has the same predictive ability as the two "short-range repression" mechanisms tested. In light of literature evidence for short-range repression [65, 66], this finding suggests that discerning true mechanisms from modeling of experimental results *ex post facto* may not be adequately powered, and that active learning approaches [67, 68] that suggest the most informative future experiments may be called for.

While thermodynamics-based models allow interactions between TF binding sites to depend on their relative arrangement in the sequence, the form of the dependence has to be pre-determined. For example, commonly, cooperative DNA binding and short-range repression are assumed to have a fixed strength as long as inter-site distance is less than a pre-set threshold [6, 7, 8] and absent otherwise. This assumption reduces the complexity of the model, though it may oversimplify the distance-dependence of TF interactions. (Sayal et al. [26] considered a more flexible form for this dependence by using separate tunable interaction parameters for different intervals of inter-site spacings.) Moreover, typical thermodynamics-based models allow interactions only in configurations where the two TF bound to proximal sites have no other bound TF in between; this is necessitated by considerations of computational efficiency. These constraints of thermodynamics-based models led us to encode a more flexible model of distance-dependent TF-TF interactions (cooperative as well as antagonistic) in CoNSEPT, through layers of convolution kernels that process all putative site pairs near each other. By examining the trained model's predictions on a simple synthetic enhancer with varying inter-site distances, we were then able to extract the distance-dependence function learned by it (Figure 2.8G-J) in a data-driven manner. The learned function is mostly in agreement with that used in the thermodynamics-based models, e.g., both approaches suggest cooperative effects of DL-TWI site pairs within 40 bp of each other [23], as well as repressive effects of SNA sites located within 80 bp of an activator site. Interestingly, one of the three CoNSEPT models is consistent with "direct repression" (not limited to short-range effects), an observation made also when fitting thermodynamics-based models with this form of repression.

A key contribution of this work is the design and implementation of CoNSEPT, which is a general-purpose tool for sequence-to-expression modeling using convolutional neural networks. Our work demonstrates the feasibility of training such complex models (thousands of free parameters) on limited data sets (hundreds rather than thousands of samples), and we have tested that it can handle other data sets of much larger size (tens of thousands of samples, data not shown). It stands in contrast to other available implementations of neural networks for regulatory genomics, which are targeted to modeling epigenomic [45, 69, 70] and genome-wide TF-DNA binding [42, 44] data, or do not explicitly model the dependence of sequence function on cellular descriptors such as TF levels [71]. This feature allows CoNSEPT to make predictions for varying cellular conditions. Moreover, its reliance on pre-defined PWMs is different from previous applications of neural networks in regulatory genomics where TF motifs were learned from the data using convolutional kernels [44, 45, 70]. We believe that using known PWMs, which are often reliably characterized experimentally [72], reduces the number of free parameters and lowers the chance of overfitting.

The work most closely related to the CoNSEPT model is the neural network model presented by Liu et al. [73] who used it to model the expression driven by enhancers related to the *eve* gene of *Drosophila*. Their model is a recasting of the thermodynamics-based model of Kim et al. [64] and includes all of the latter's mechanistic aspects. At the same time, the model of Liu et al. is specialized for the biological system (*eve* enhancers) studied by them and encodes distance-dependence of interactions with pre-determined form and parameters motivated by that system. CoNSEPT, on the other hand, departs from the thermodynamics-based formalism and takes a more data-driven approach to capturing cis-regulatory "grammar" and distance-dependent interactions.

In summary, our work shows that there are multiple formalisms capable of explaining the sequence-function mapping encoded in enhancers, with complementary strengths and varying reliance on prior mechanistic knowledge. Although the mechanisms that we find are specific to the regulation of *trhomboid* and other neuroectodermal enhancers, and might not necessarily generalize to the regulation of other genes, our work gives a recipe for understanding the regulatory mechanisms in a data-driven or model-driven manner. In addition to their explanatory role, the models tested here can also be useful for predicting the expression driven by unseen sequences and cellular contexts; this ability has several applications in down-stream analysis such as predicting the effect of a particular TF's knockout, site mutagenesis, or effects of single nucleotide polymorphisms on the expression [74, 75]. We also showed that convolutional neural networks can be a reliable expression prediction tool capable of learning non-linear regulatory mechanisms from modest-sized training data.

CHAPTER 3: SIMULATIONS OF REGULATORY NETWORKS

This chapter is a reproduction of the work published in the Cell Systems journal [76].

3.1 ABSTRACT

A common approach to benchmarking of single-cell transcriptomics tools is to generate synthetic data sets that statistically resemble experimental data. However, most existing single-cell simulators do not incorporate transcription factor-gene regulatory interactions that underlie expression dynamics. Here we present SERGIO, a simulator of single-cell gene expression data that models the stochastic nature of transcription as well as regulation of genes by multiple transcription factors according to a user-provided gene regulatory network. SERGIO can simulate any number of cell types in steady-state or cells differentiating to multiple fates. We show that data sets generated by SERGIO are statistically comparable to experimental data generated by Illumina HiSeq2000, Drop-seq, Illumina 10X chromium and Smart-seq. We use SERGIO to benchmark several single-cell analysis tools, including GRN inference methods, and identify Tcf7, Gata3, and Bcl11b as key drivers of T-cell differentiation by performing in silico knockout experiments. SERGIO is freely available at: https://github.com/PayamDiba/SERGIO.

3.2 INTRODUCTION

Single-cell transcriptomics technologies are revolutionizing biology today [77, 78, 79, 80], and have led to the rapid development of computational tools for analyzing the resulting data sets [81, 82, 83, 84]. These tools, developed for a wide array of tasks such as clustering [85, 86, 87], trajectory inference [88, 89] and gene regulatory network (GRN) reconstruction [85, 90, 91], as well as pre-processing operations such as imputation [92, 93, 94], adopt complementary strategies whose relative merits and weaknesses are not clear *a priori*. In some cases, single-cell data sets annotated using domain knowledge [95, 96] allow objective evaluations of different strategies, but this is not a scalable approach to systematic benchmarking. A promising alternative approach is to synthesize single-cell expression data sets that mimic real data in their statistical properties and for which underlying biological relationships, e.g., cell type labels, regulatory influences, etc., are known by construction. One advantage of such synthetic data sets is the ability to systematically modify the biological and technical parameters underlying the data in order to understand their effects on a tool's performance, as well as the ability to obtain replicates of data sets for robust statistical estimates of performance.

Simulation tools ("simulators") for single-cell expression data have been reported in various forms. Several studies offering novel analysis tools use in-house simulators to benchmark those tools [84, 97, 98, 99, 100, 101, 102], while other studies specifically develop simulators for use by the community [103, 104, 105, 106, 107, 108]. Most of these simulators are geared towards capturing the noise characteristics of technologies such as single-cell RNA-seq (scRNA-seq), by first estimating statistical quantities describing real data sets and then sampling single-cell expression profiles from probability distributions that mirror those quantities. A crucial aspect of biology missing in current simulators is the gene regulatory network (GRN): the set of transcription factor (TF)-gene relationships that underlies the dynamics and steady states of gene expression in each cell. In other words, when sampling an expression value for a gene in a cell, these simulators do not account for the fact that the gene is expressed under the control of one or more TFs, whose concentrations in the cell have a major role in determining the target gene's expression. We thus sought to develop a single-cell expression simulator that is guided by an underlying GRN, not only because of the biological realism that it represents, but also because this is the only direct way to benchmark tools specifically designed for GRN reconstruction. Some existing tools do attempt to induce gene-gene relationships in synthetic data using multi-gene statistical models for sampling purposes [104, 109], but these attempts do not explicitly incorporate the special properties of gene regulatory networks that have been reported in the literature [110, 111, 112, 113], including non-linear response to TFs, intrinsic fluctuations in expression and propagation of such "biological noise" along the GRN.

In the realm of "bulk" transcriptomics, GRN-driven simulations are already the norm, as exemplified by the simulation tool called GeneNetWeaver (GNW) [114], which was used in a community-wide effort to benchmark numerous GRN reconstruction tools [115, 116, 117, 118]. GNW is not meant to simulate scRNA-seq data, and though some studies have employed workarounds to use it for this purpose [90, 119], it is believed that such synthetic data do not exhibit the statistical characteristics of contemporary single-cell data sets [119]. Furthermore, such workarounds do not offer key features necessary for a single cell expression simulator, such as simulation of multiple cell types and cells differentiating from one cell type to another.

In this work, we develop a simulator tool that (1) uses a principled mathematical description of transcriptional regulatory processes to synthesize single-cell expression data associated with a specified GRN, (2) includes stochasticity of gene expression as an integral part of the process, thus capturing biological noise expected to manifest in cell-to-cell variability, and (3) incorporates various types of measurement errors ("technical noise") that are typical of single-cell technologies. The new tool, called SERGIO (**S**ingle-cell **E**xp**R**ession of **G**enes **I**n silic**O**), is freely available as a stand-alone software package. It borrows some of its modeling assumptions from the widely used GNW simulator, but relinquishes the more complex features of GNW, such as a thermodynamics-based model of regulation and explicit modeling of translation processes, which would have necessitated use of poorly-understood parameters during simulation and slowed down simulations of large GRNs.

SERGIO uses a stochastic differential equation (SDE) called the chemical Langevin equation [120] to simulate a gene's expression dynamics as a function of the changing (or fluctuating) levels of its regulators (TFs), as prescribed by a fixed GRN. It performs such simulations for any pre-specified number of genes in parallel and generates single-cell expression "profiles" (expression values of all genes) by sampling from these temporal simulations in steady-state, thus mimicking established cell types. It allows users to specify the number of cell types to be simulated, via steady-state levels of a few "master" regulators in the GRN. SERGIO also allows users to simulate single-cell expression data from a specified differentiation program, for which it samples cells from transient portions of temporal simulations. In this simulation mode, SERGIO explicitly models the splicing step with an additional SDE, resulting in simulations of unspliced and spliced transcript levels. SERGIO subjects the synthesized expression data to a multi-step transformation where technical noise is incorporated in a manner reflecting noise in real scRNA-seq data.

SERGIO is a stand-alone simulator tool for single-cell transcriptomics that offers all of the above-mentioned features while basing its simulations on a given GRN. Here, we outline key aspects of its model and implementation and show that it may be used to generate realistic data sets that resemble experimental data obtained from popular scRNA-seq technologies, by several statistical measures. We then showcase its use to benchmark a number of popular single-cell analysis tools. We find that while modern tools are able to accurately identify cell types and differentiation trajectories from suitable data sets, their ability to reconstruct gene regulatory relationships remains severely limited. To demonstrate the use of SERGIO beyond benchmarking studies, we apply it to simulate expression T-cell differentiation data at single-cell resolution, using two different draft GRNs, show that the simulated data matches a recently published data set for this differentiation process, and also examine the effects of specific perturbations on the process.

3.3 MATERIALS AND METHODS

3.3.1 Steady-State Simulations

The Chemical Master Equation (CME) offers a paradigm for modelling stochastic transcription of genes [121, 122]. Gillespie's stochastic simulation algorithm [123, 124] enables the computation of trajectories according to CME. However, simulating trajectories of CME for large biomolecular systems is computationally expensive [122]. Chemical Langevin equation (CLE) [120] that is derived from CME under two additional assumptions provides a more tractable system of differential equations compared to CME; therefore, CLE can be integrated with numerical methods in a computationally efficient manner and allows for practical simulations. It has been shown that Gillespie's stochastic simulations of CME and numerical simulations of CLE generate trajectories that have comparable mean, variance and correlations [122]. Hence, we model the dynamics of the concentration of genes using systems of stochastic differential equations (SDE) that have been previously employed in GeneNetWeaver (GNW) [114, 116] and which are derived from the chemical Langevin equation (CLE) [120]. The time-course of mRNA concentration of gene i is modeled by:

$$\frac{dx_i}{dt} = P_i(t) - \lambda_i x_i(t) + q_i(\sqrt{P_i(t)}\alpha + \sqrt{\lambda_i x_i(t)}\beta) \tag{3.1}$$

where x_i is the expression of gene i, P_i is its production rate, which reflects the influence of its regulators as identified by the given GRN (details below), λ_i is the decay rate, and q_i is the noise amplitude in the transcription of gene i. α and β are two independent Gaussian white noise processes. This model relies on the assumption that there is no delay between TF-mediated regulation and mRNA production. In order to obtain the mRNA concentrations as a function of time, we integrate the above stochastic differential equation for all genes:

$$(x_i)_t = (x_i)_{t_0} + \int_{t_0}^{t} (P_i(t) - \lambda_i x_i(t))dt + \int_{t_0}^{t} q_i\sqrt{P_i(t)}dW_\alpha + \int_{t_0}^{t} q_i\sqrt{\lambda_i x_i(t)}dW_\beta \tag{3.2}$$

where W_α and W_β are two independent stochastic Wiener processes. We integrate this equation in pre-defined time steps of duration , according to Euler–Maruyama method [125] using the Itô scheme:

$$(x_i)_{t+\Delta t} = (x_i)_t + (P_i(t) - \lambda_i x_i(t))\Delta t + q_i\sqrt{P_i(t)}\Delta W_\alpha + q_i\sqrt{\lambda_i x_i(t)}\Delta W_\beta \tag{3.3}$$

$$\Delta W_\alpha = \sqrt{\Delta t}\mathcal{N}(0,1), \qquad \Delta W_\beta = \sqrt{\Delta t}\mathcal{N}(0,1) \tag{3.4}$$

Each iteration yields the mRNA concentrations of all genes at time step $t + \Delta t$ using each gene's own concentration and all of its regulators' concentrations at time step t. We model each gene's production rate, P_i, as the sum of contributions from each of its regulators (as prescribed by the GRN):

$$P_i = \sum_{j \in R_i} p_{ij} + b_i \tag{3.5}$$

where R_i is the set of all regulators of gene i, b_i is the basal production rate of gene i, and p_{ij} is the regulatory effect of gene (TF) j on gene i. The latter is modeled as a non-linear saturating Hill function of the mRNA concentration of the TF [126]:

$$p_{ij} = K_{ij}\frac{x_j^{n_{ij}}}{h_{ij}^{n_{ij}} + x_j^{n_{ij}}}; \qquad \textit{if regulator } j \textit{ is an activator of gene } i \tag{3.6}$$

$$p_{ij} = K_{ij}(1 - \frac{x_j^{n_{ij}}}{h_{ij}^{n_{ij}} + x_j^{n_{ij}}}); \qquad \textit{if regulator } j \textit{ is a repressor of gene } i \tag{3.7}$$

where K_{ij} denotes the maximum contribution of regulator j to target gene i, n_{ij} is the Hill coefficient that introduces non-linearity to the model and is the regulator concentration that produces half-maximal regulatory effect (half-response). If gene i is a user-designated "master regulator" (MR), i.e., no gene regulates it, then its production rate P_i is entirely determined by basal production rate b_i which is a user-defined parameter. For simplicity, we set $b_i = 0$ for genes other than master regulators. K_{ij} and h_{ij} are user-defined parameters, and the type of each interaction (activation or repression) is also user-specified. The h_{ij} parameter is set to be the average of the regulators' expression among the cell types to be simulated. The parameters α and β in equation 1 characterize the intrinsic noise associated with the production and decay processes of the mRNA transcript of gene i. Moreover, the intrinsic noise in the transcription of regulators propagates along the GRN and thus influences the production rate P_i to become an extrinsic noise source in the transcription of gene i. We support three forms of noise:

1. Dual Production Decay ("dpd"): the form of stochastic noise that is shown in equation 3.1.

2. Single Production (“sp”): including only the noise term associated with the production process (equivalently, set $\beta = 0$).
3. Single Decay (“sd”): including only the noise term associated with the decay process (equivalently, set $\alpha = 0$)

We note that the current version of SERGIO is not capable of simulating GRNs containing auto-regulatory edges or cycles. This is because of a topological sorting algorithm in SERGIO that enables the automatic selection of half-response parameters and fails in the presence of cycles in GRN. It is a shortcoming as cycles are often present in real GRNs, but it can be resolved by eliminating the dependency of SERGIO on the topological sorting algorithm.

3.3.2 Sampling Single-Cells

We use the above system of equations (equation 3.3-3.7) to simulate the time-course of each gene’s expression in a cell, starting with a given initial value, and record expression values of all genes at randomly selected time points after the simulation has reached steady state. Invoking the ergodic assumption [127], we treat the expression profiles at these time points to represent single-cell profiles. In order to speed up the simulation, we estimate the steady-state concentrations of all genes given the input parameters and initialize the time-course simulation with those values. Also, we ensure that a sufficient number of time steps, which is controlled by a user-defined parameter, are simulated in the steady state prior to sampling cells.

3.3.3 Cell Types

The above simulation is performed for each “cell type” separately. We define a cell type (or cell state) by the average concentration of master regulators. A cell type differs from another cell type by the average concentration of one or more of the master regulators among the population of cells belong to each cell type. This can be controlled by the basal production rate b for master regulators. SERGIO takes as input the basal production rate of all master regulators in each of the cell types to be simulated.

3.3.4 Estimating Steady-State Concentrations

In steady state-simulations, SERGIO approximates the steady-state concentrations of all genes in all cell types prior to starting the simulations. Then, SERGIO initializes all the concentrations with their corresponding estimated stead-states values. This is particularly useful for speeding up the simulations since initial concentrations are so close to the values to which the numerical integration is supposed to converge. To do so, SERGIO applies a topological sort algorithm on the gene regulatory network in order to layer the graph. After layering the GRN graph using topological sorting, all the regulators of the genes of any layer reside in the preceding layers. Sergio starts estimation of steady-state concentrations (as well as half-response parameters of the hill functions) from the top most layer and continues layer-by-layer until the very last layer of genes. Therefore, all information required for estimating the steady-state concentrations (and half-responses) of genes in the current layer is already available from the user-defined parameters and the estimated concentrations of the genes in the previous layers. Below we demonstrate how Sergio estimates steady-state concentrations.

We start with equation 3.1 that describes the rate of changes in mRNA concentration x_i of gene i. In order to reach the steady-state regime, we need to have $dE[x_i]/dt = 0$, where $E[.]$ denotes the expectation operator. Since dx_i/dt is well defined and is bounded, we have $dE[x_i]/dt = E[\frac{dx_i}{dt}]$. So, we get:

$$E[\frac{dx_i}{dt}] = 0 \Rightarrow \tag{3.8}$$

$$E[P_i(t) - \lambda_i x_i(t) + q_i(\sqrt{P_i(t)}\alpha + \sqrt{\lambda_i x_i(t)}\beta)] = \tag{3.9}$$

$$E[P_i(t)] - \lambda_i E[x_i] + q_i E[\sqrt{P_i(t)}]E[\alpha] + q_i E[\sqrt{\lambda_i x_i(t)}]E[\beta] = 0 \tag{3.10}$$

Also recall that α and β are two Gaussian white noise processes which have a zero mean, so we get:

$$E[P_i(t)] - \lambda_i E[x_i] = 0 \quad \Rightarrow \quad E[x_i] = \frac{E[P_i(t)]}{\lambda_i} \tag{3.11}$$

If gene i is a master regulator, according to equation 3.5 its production rate is solely determined by its user-defined production rate b_i and therefore we can accurately estimate the expected steady-state concentration x_i:

$$E[x_i] = \frac{E[b_i]}{\lambda_i} = \frac{b_i}{\lambda_i}; \qquad \textit{if gene } i \textit{ is a master regulator} \tag{3.12}$$

However, if gene i is not a master regulator, according to equation 3.5 we get:

$$E[x_i] = \frac{E[P_i]}{\lambda_i} = \frac{\sum_{j \in R_i} E[p_{ij}]}{\lambda_i}; \qquad \textit{if gene } i \textit{ is not a master regulator} \tag{3.13}$$

where R_i is the set of all regulators of gene i, and p_{ij} is a hill function. We use the following approximation for calculating $E[p_{ij}(x_j)]$:

$$E[p_{ij}(x_j)] \approx p_{ij}(E[x_j]) \tag{3.14}$$

Note that this a loose approximation as it clearly over-estimates or under-estimates the true value of $E[p_{ij}(x_j)]$, yet it is good enough for our ultimate goal which is initializing mRNA concentrations. Substituting this back into the equation 3.13 we obtain an estimate of the steady-state concentration of gene i:

$$E[x_i] = \frac{\sum_{j \in R_i} p_{ij}(E[x_j])}{\lambda_i}; \qquad \textit{if gene } i \textit{ is not a master regulator} \tag{3.15}$$

At the time we calculate the concentration of gene i we have already calculated the steady-state concertation of all of its regulators (which reside in the preceding layers of the sorted GRN), hence we have all the information required for this calculation. Note also that the goal of the above estimation is not to find steady-state concentrations *per se*, but to find values close to these concentrations so as to reduce simulation time by starting the simulations at these concentrations.

3.3.5 Estimating Half-Response Parameter

We model each interaction with a Hill function that has a half-response parameter h that needs to be pre-specified for the simulations. If we use a small value of half-response ($h \ll [TF]$), then the Hill function becomes a constant function independent of TF concentration:

$$K \frac{[TF]^n}{[TF]^n + h^n} \approx K \tag{3.16}$$

On the other hand, if we use a very large value for half-response ($h \gg [TF]$), then the Hill function does not have its non-linear saturating form anymore (especially for $n = 1$, where it becomes linear):

$$K \frac{[TF]^n}{[TF]^n + h^n} \approx K (\frac{[TF]}{h})^n \tag{3.17}$$

So, we considered it important to set the half-response parameter to a value that yields the saturating non-linear behavior of Hill functions, across different cell types. While it is difficult for user to define this parameter (because, for an arbitrary TF→target interaction, user may not know the expected concentration of TF *a priori*), SERGIO can determine reasonable values of the half-response parameter using the concentration of all regulators of the current target (that reside in the previous layers obtained by topological sorting algorithm). In the current implementation of SERGIO, for each interaction, we set half-response to the mean expression of the regulator in all cells (belonging to multiple cell types). This guarantees that we see a large range of response over different cells and cell types. As explained in "Estimating Steady-State Concentrations" section above, at the time we get to a new gene, the expression of all of its regulators (which live in upper levels of sorted GRN) have been already simulated because of the topological sorting algorithm. This enables the calculation of half response parameter based on the expression of regulators. This design choice was made so as to take the burden of setting the half-response parameter away from the user, that also necessitates the topological sorting of the GRN graph and thus demands an acyclic graph.

3.3.6 Simulation of Differentiation Trajectories

In addition to simulating one or more "cell types" in steady state, SERGIO may be used to simulate cells on the differentiation trajectory from one cell type to another, i.e., between two steady states. More generally, given a "differentiation graph" where nodes represent cell types and directed edges indicate differentiation from one cell type to the other, SERGIO can simulate expression profiles of cells spanning different stages of differentiation specified by the graph. Such cells are either in one of the steady states represented by nodes or have departed away from the steady-state of their "parent" cell type of an edge and are migrating toward the steady-state of the corresponding "child" cell type. The differentiation is presumed to commence when one or more master regulators change their expression from that in the steady state of the parent cell type, e.g., due to a signaling event [128] or due to a noise-driven switch [129]. Thus, given a differentiation graph and average expression levels of master regulators for each cell type (nodes), we simulate each differentiation trajectory (edge) as follows: 1) Cells representing the parent cell type are sampled from the corresponding steady state. 2) Production rates (P_i) of master regulators are changed from those specified for the parent cell type to those of the child cell type, and time-course simulations are performed following equations 3-4 as explained above. As these simulations proceed, all genes ultimately converge to their steady-state concentrations in the child cell type. 3) Cells (expression profiles) are sampled at random from the entire simulation, including cells in the parent and child cell types (steady states) as well as cells on the differentiation trajectory (transient states). Multiple such time-course simulations are performed and the sampled cells are randomly chosen from the entire collection of such simulations. Also, after each simulation reaches the steady-state of the child cell type, it may be continued for a user-defined number of additional steps. This controls the ratio of the cells in the steady states of the differentiation graph to the number of cells in differentiating (transient) states.

Simulations of differentiation trajectories in SERGIO generate not only the total mRNA concentration of each gene (in a time-course), but the changing levels of spliced and unspliced mRNA transcripts separately. To this end, we express the rate of change in the concentration of unspliced and spliced RNA using ordinary differential equations (ODEs), following prior work [130, 131]. Furthermore, we introduce noise terms to these ODEs in a manner similar to steady-state simulations (equation 3.1). Thus, the time-course of the spliced (s) and unspliced (u) transcript level of gene i is modeled as:

$$\frac{du_i}{dt} = P_i(t) - (\lambda_i + \mu_i)u_i(t) + q_i^u(\sqrt{P_i(t)}\alpha + \sqrt{(\lambda_i + \mu_i)u_i(t)}\beta) \tag{3.18}$$

$$\frac{ds_i}{dt} = \mu_i u_i(t) - \gamma_i s_i(t) + q_i^s(\sqrt{\mu_i u_i(t)}\phi + \sqrt{\gamma_i s_i(t)}\omega) \tag{3.19}$$

where $P_i(t)$ is the production rate of pre-mRNA (unspliced transcript) that includes regulatory interactions, λ_i and μ_i are the degradation and splicing rate respectively of pre-mRNA and q_i^u is the noise amplitude associated with the transcription of pre-mRNA. For simplicity, we assume the degradation rate λ_i of pre-mRNA is zero and all of its decay is due to splicing (user-defined parameter μ_i). Also, γ_i is the degradation rate of spliced mRNA and q_i^s is the noise amplitude associated with the transcription of spliced mRNA. α, β, ϕ and ω are independent Gaussian white noise processes. All the three form of stochastic noise ("dpd", "sp", "sd") described for steady-state simulations are also supported in dynamics simulation. Moreover, production-rate is modeled as in steady-state simulations (equations 3.5-3.7 above). Both of the SDEs in equations 3.11-3.12 are integrated according to Euler–Maruyama scheme to obtain time-courses of unspliced and spliced mRNA concentrations.

3.3.7 Technical Noise

SERGIO adopts methods similar to Splatter [107] for adding technical noise to the simulated single-cell expression data. One module introduces the phenomenon of "outlier genes", which refers to the empirical observation that a small set of genes appear to have unusually high expression measurements across cells in typical scRNA-seq data sets. A second module incorporates the noted phenomenon of different cells having different total counts (library size), that follows a log-normal distribution. A third module introduces "dropouts", which refers to the observation that a high percentage of genes are recorded at zero expression in any given cell, indicating an experimental failure to record their expression rather than true non-expression. These three modules may be invoked optionally, and in any combination and order specified by the user. Each of the modules outlined below adds a single type of technical noise to the data set provided to it.

Outlier genes: Each gene is designated as an outlier with a user-defined probability. If so, its expression (in every cell) is multiplied by a factor sampled from a log-normal distribution, otherwise the expression is left unchanged. In the steady-state mode, outlier genes are introduced as following:

$$\forall i \in \{1..G\}: \quad \mathbb{I}_i^O \sim Ber(\pi^O), \;\; f_i^O \sim ln\mathcal{N}(\mu^O, \sigma^O) \tag{3.20}$$

$$\forall i \in \{1..G\}, \forall c \in \{1..C\}: \quad x_i^c \leftarrow \mathbb{I}_i^O f_i^O x_i^c + (1 - \mathbb{I}_i^O) x_i^c \tag{3.21}$$

where G and C denote the total number of simulated genes and cells respectively, and x_i^c denotes the simulated expression of gene i in cell c. $\mathbb{I}_i^O$ is a binary variable indicating if gene i is an outlier, and is sampled from a Bernoulli distribution with parameter π^O. Also, μ^O and σ^O are user-defined mean and standard deviation of the lognormal distribution from which the outlier scaling factor f_i^c is sampled. We follow a similar scheme in differentiation mode by changing both unspliced and spliced transcripts:

$$\forall i \in \{1..G\}: \quad \mathbb{I}_i^O \sim Ber(\pi^O), \;\; f_i^O \sim ln\mathcal{N}(\mu^O, \sigma^O) \tag{3.22}$$

$$\forall c \in \{1..C\}, \forall i \in \{1..G\}: \quad u_i^c \leftarrow \mathbb{I}_i^O f_i^O u_i^c + (1 - \mathbb{I}_i^O) u_i^c, \;\; s_i^c \leftarrow \mathbb{I}_i^O f_i^O s_i^c + (1 - \mathbb{I}_i^O) s_i^c \tag{3.23}$$

where u_i^c and s_i^c denote the simulated unspliced and spliced concentrations respectively of gene i in cell c.

Library size: For every cell (library) a library size parameter is sampled from a lognormal distribution, and expression values of all genes in the cell are scaled by a constant factor such that the resulting total cell depth matches the sampled library size. In steady-state mode we have:

$$\forall c \in \{1..C\}: \quad L_c \sim ln\mathcal{N}(\mu^L, \sigma^L) \tag{3.24}$$

$$\forall i \in \{1..G\}, \forall c \in \{1..C\}: \quad x_i^c \leftarrow \frac{L_c}{\sum_{j\in\{1..G\}} x_j^c} x_i^c \tag{3.25}$$

where μ^L and σ^L are the user-defined mean and standard deviation of the lognormal distribution from which the desired library size of cell c is sampled. Following the same approach, we scale both unspliced and spliced transcripts in the differentiation mode:

$$\forall c \in \{1..C\}: \quad L_c \sim ln\mathcal{N}(\mu^L, \sigma^L) \tag{3.26}$$

$$\forall i \in \{1..G\}, \forall c \in \{1..C\}: \quad u_i^c \leftarrow \frac{L_c}{\sum_{j\in\{1..G\}} u_j^c + s_j^c} u_i^c, \;\; s_i^c \leftarrow \frac{L_c}{\sum_{j\in\{1..G\}} u_j^c + s_j^c} s_i^c \tag{3.27}$$

Dropout: To introduce dropouts to the simulated data, we first assign a probability to the expression of each gene in each of the simulated cells not being a dropout. This probability is modeled as a logistic function of the expression of the gene in that cell, so that a high expression value is less likely to be zeroed out. This probability is then used as the parameter of a Bernoulli distribution from which a binary variable is sampled to indicate whether the gene is not a dropout in the cell. Dropout is introduced to the steady-state simulations as following:

$$y_0 = q^{th} percentile \ of \ Y \tag{3.28}$$

$$\forall i \in \{1..G\}, \forall c \in \{1..C\}: \quad \pi_{i,c}^D = \frac{1}{1 + exp(-k(Y_{i,c} - y_0))}, \;\; \mathbb{I}_{i,c}^D \sim Ber(\pi_{i,c}^D) \tag{3.29}$$

$$x_i^c \leftarrow \mathbb{I}_{i,c}^D x_i^c \tag{3.30}$$

where Y is the expression matrix in logarithmic scale:

$$Y = log(X+1), \quad X = \{x_i^c; \; \forall i \in \{1..G\}, \forall c \in \{1..C\}\} \tag{3.31}$$

Also, k and q are two user-defined parameters that determine the logistic probability π^D. In real single-cell data, dropout impacts unspliced and spliced transcripts independently. To model this in the differentiation mode, we employ a similar approach as we use for steady-state simulations but add dropout to spliced and unspliced expressions independently:

$$y_0 = q^{th} percentile\ of\ Y \tag{3.32}$$

$$\forall i \in \{1..G\}, \forall c \in \{1..C\}: \quad \pi_{i,c}^{D,U} \frac{1}{1 + exp(-k(Y_{i,c}^{U} - y_0))}, \ \pi_{i,c}^{D,S} \frac{1}{1 + exp(-k(Y_{i,c}^{S} - y_0))} \tag{3.33}$$

$$\mathbb{I}_{i,c}^{D,U} \sim Ber(\pi_{i,c}^{D,U}), \ \mathbb{I}_{i,c}^{D,S} \sim Ber(\pi_{i,c}^{D,S}) \tag{3.34}$$

where denotes the total mRNA expression matrix in logarithmic scale:

$$Y = log(X + 1), \ X = X^U + X^S \tag{3.35}$$

$$X^U = \{u_i^c; \ \forall i \in \{1..G\}, \forall c \in \{1..C\}\}, \ X^S = \{s_i^c; \ \forall i \in \{1..G\}, \forall c \in \{1..C\}\} \tag{3.36}$$

Also, we define:

$$Y^U = log(X^U + 1), \ Y^S = log(X^S + 1) \tag{3.37}$$

Conversion to UMI counts: In steady-state simulations, we generate UMI counts (UC) by sampling from a Poisson distribution whose mean is the simulated expression level of the gene in the cell:

$$\forall i \in \{1..G\}, \forall c \in \{1..C\}: \quad UC_{i,c} \sim Poisson(x_i^c) \tag{3.38}$$

In differentiation simulations, spliced (UC^S) and unspliced (UC^U) mRNA counts are independently sampled from a Poisson distribution whose mean is the simulated expression level of the gene in the cell:

$$\forall i \in \{1..G\}, \forall c \in \{1..C\}: \quad UC_{i,c}^U \sim Poisson(u_i^c), \ UC_{i,c}^S \sim Poisson(s_i^c) \tag{3.39}$$

3.3.8 Controlling Spliced to Unspliced Count Ratio

We can estimate and control the ratio of the expected spliced to unspliced expression in the stationary region of the stochastic transcription process. Using equation 3.19, under steady-state condition we get:

$$\frac{dE[s_i]}{dt} = E[\frac{ds_i}{dt}] = 0 \Rightarrow \mu_i E[u_i] - \gamma_i[s_i] = 0 \tag{3.40}$$

$$\frac{E[s_i]}{E[u_i]} = \frac{\mu_i}{\gamma_i} \tag{3.41}$$

SERGIO, as an input, takes the decay rate μ of the unspliced mRNA as well as the splicing ratio $E[s_i]/E[u_i]$; these two together determine the degradation rate of spliced RNA γ according to the last equation. This enables the user to simulate gene expressions with any desired spliced to unspliced ratios in order to reproduce different experimental settings.

3.3.9 Synthetic Data Set Generation

We now describe how we set simulation parameters to generate the data sets analyzed in this study. We sampled four gene regulatory networks (GRNs) from the known regulatory networks in S. cerevisiae and E. coli using GNW [114] using the "random seed" argument to select genes and the "random among top 20%" setting for neighbor selection. Two of the networks consist of 100 genes and were separately sampled from Ecoli, a third network containing 1200 genes was sampled from E. coli, and the fourth network comprising 400 genes was sampled from S. cerevisiae. We also used GNW to designate each TF-gene edge as either an activating or a repressive interaction. Auto-regulatory edges were removed from the sampled networks and cycles were broken at a randomly selected edge, since SERGIO does not support these two graph properties. These four networks were used to simulate 15 data sets described in Table 3.1. Except for DS13 and DS14 that were simulated in only one "replicate", fifteen replicates of each data set were created that had identical simulation parameters and differed due to the stochastic noise and random sampling. For all data sets, interaction strengths K_{ij} (equations 3.6-3.7) were uniformly sampled from the range 1 to 5. Each cell type to be simulated was specified by the expression state (high or low) of each master regulator (MR); the basal production rate (b_i in equation 3.5) of each MR was sampled from a pre-defined range that depends on the expression state and varies among different data sets (see Appendix B, Supplementary Table B.). We used a hill coefficient of 2 for all interactions in all data sets. We used the same noise amplitude parameter $q = 1$ and the same decay parameter $\lambda = 0.8$ for all genes in all steady-state data sets. In dynamics simulations, we used an unspliced noise parameter $q^u = 0.3$ and a spliced noise parameter $q^s = 0.07$ for all genes. Also, we used an unspliced transcript decay rate of $\mu = 0.8$ and a spliced transcript decay rate of $\gamma = 0.2$ that maintains a ratio of spliced to unspliced expression of a gene at $\sim$4. We used "dpd" setting of intrinsic noise and an integration time step of 0.01 for both steady-state and dynamics simulations.

We compared the simulated expression matrices (genes x cells) of DS1, DS2, DS3, and DS9 to a single-cell RNA-seq data set from the mouse cerebral cortex [132], which includes 3005 cells from nine cell types, as a reference for adding technical noise. We sampled the mouse cortex data set by drawing cells of each type at random: for cell types with less than 300 cells, we retained all the cells, while for the other cell types we randomly sampled 300 cells such that a total of 2500 single cells were sampled. Our simulations generated expression values for 100, 400 or 1200 genes depending on the data set, hence we randomly sampled from the real data set the same number of genes as present in the synthetic data. Moreover, we used five other published single-cell RNA-seq data sets as a reference for adding technical noise to data sets, which respectively contain 16383 cells of mouse MGE, CGE, cortical and subcortical regions [133], 3745 cells of human kidney [134], 12874 of Human Peripheral blood mononuclear cells (PBMC) [135], 10360 cells of human lung [136], and 6002 cells of mouse heart [96]. These resulted in the data sets DS4-8 respectively, and for each of these we repeated the same sampling strategy described above to create comparison data sets, sampling 1200 genes at random while preserving all the cells in each sample. We also we used a published single-cell RNA-seq data set containing 24185 cells from developing mouse dentate gyrus [137] as a reference for adding technical noise to the data set DS13. We generated comparison data sets by sampling 100 genes at random, while all single-cells were preserved.

To add technical noise, we used the above-mentioned modules for outlier genes, library size effect and dropouts in that order, and finally converted the expression levels to UMI counts. For each data set, we manually tuned the input parameters (see Appendix B, Supplementary Table B.1) to each of the technical noise modules to obtain a comparable noise level between the synthetic and real data.

3.3.10 Simulations with Cooperative Regulation

We also developed an in-house version of SERGIO that includes cooperative regulation. In this mode of simulation, production rate of a target gene i is calculated as:

Table 3.1: Description of the Synthetic Datasets Used in This Study

Dataset ID	**Network ID**	**Species**	**#Genes**	**#Cells**	**#Regulators**	**#Edges**	**#Cell Types**	**Differentiation**	**Matched Against**
DS1	2	E. coli	100	2700	10	258	9	no	mouse cerebral cortex (Illumina HiSeq 2000)
DS2	3	yeast	400	2700	37	1155	9	no	mouse cerebral cortex (Illumina HiSeq 2000)
DS3	4	E. coli	1200	2700	127	2713	9	no	mouse cerebral cortex (Illumina HiSeq 2000)
DS4	4	E. coli	1200	2700	127	2713	9	no	mouse MGE, and ... (10X Chromium)
DS5	4	E. coli	1200	2700	127	2713	9	no	human kidney (10X chromium)
DS6	4	E. coli	1200	2700	127	2713	9	no	human PBMC (10X chromium)
DS7	4	E. coli	1200	2700	127	2713	9	no	human lung (Drop-seq)
DS8	4	E. coli	1200	2700	127	2713	9	no	mouse heart (Smart-seq2)
DS9	1	E. coli	100	900	10	137	3	yes	–
DS10	1	E. coli	100	1200	10	137	4	yes	–
DS11	1	E. coli	100	1800	10	137	6	yes	–
DS12	1	E. coli	100	2100	10	137	7	yes	–
DS13	1	E. coli	100	24000	10	137	4	yes	mouse dentate gyrus (10X chromium)
DS14	1	E. coli	100	36000	10	137	6	yes	mouse cerebral cortex (Illumina HiSeq 2000)
DS15	1	E. coli	100	900	10	137	3	yes	–

$$P_i = \sum_{j \in R_i} p_{ij} + \sum_{(m,n) \in C_i} s_{i,(mn)} \tag{3.42}$$

where R_i is the set of all regulators of gene i, and p_{ij} is calculated using equations 3.6-3.7. Also, C_i denotes the set of all activator pairs $(x, y), x \in R_i, y \in R_i, x \neq y$ that cooperatively regulate gene i, and $s_{i,(mn)}$ denotes contributions from cooperative regulation of gene i by two of its activators, m and n, which is calculated as following:

$$s_{i,(mn)} = K_{i,(mn)} \frac{(x_m x_n)^{n_{i,(mn)}}}{(h_{i,(mn)})^{n_{i,(mn)}} + (x_m x_n)^{n_{i,(mn)}}} \tag{3.43}$$

where $K_{i,(mn)}$ denotes the maximum contribution of $m - n$ cooperative interaction to regulation of target gene i, x_m and x_n respectively represent the concentration of regulator m and n, $h_{i,(mn)}$ is the product of concentration values that produces the half maximal response and $n_{i,(mn)}$ is Hill coefficient that introduces non-linearity to the model. This cooperative regulation term uses a similar Hill function form as was used to model individual regulatory effects, p_{ij}; however, it depends on both regulators' concentration, and thus approximates an AND operation.

We used a GRN containing 1200 genes and 2713 regulatory interactions (the same GRN as in DS3-8) to simulate synthetic data sets via the cooperative regulation mode of SERGIO. To this end, for each target gene i that at least has two activators we selected one or more pairs at random from its regulators set R_i to construct C_i. As a result of this step, we incorporated 268 cooperative regulation in the GRN. For calculating p_{ij} and master regulators' concentration we used the exact same parametrization as that we used for DS3. For calculating $s_{i,(mn)}$ terms we set $h_{i,(mn)}$ to the $E[x_m]E[x_n]$, where $E[.]$ denotes average over all cells and cell types, also we set $n_{i,(mn)} = 2$ for all cooperative interactions. Moreover, we sampled the maximum cooperative effect parameter $K_{i,(mn)}$ in one set of simulations from range 1 to 5 (moderate cooperative regulation; referred to as "w/ Coop") and from range 50 to 100 in another set of simulations (large cooperative regulation; "w/ large Coop") with 15 replicates for each set (see Appendix B, Supplementary Figure B.6 and B.7).

3.3.11 A Comparison Between Running Time of SERGIO and BoolODE

We compared the running time of SERGIO with that of BoolODE [138] on the same networks with varying numbers of genes. To this end, we converted the gene regulatory networks prepared for SERGIO to a format suitable for running BoolODE. Since BoolODE requires the explicit regulatory rules for each target gene, we combined all of the activators of each target gene with "AND" operands and we did the same for all repressors of each target gene. For a fair comparison, in each experiment we set the "number of cell type" parameter in SERGIO and "number of simulation time" parameter in BoolODE to the same number. We conducted four comparisons, and results are summarized in Supplementary Table B.4 (Appendix B). SERGIO runs significantly faster than BoolODE on the same networks. Since BoolODE cannot handle more than ∼10 regulators for a target gene, in the third experiment (400 genes) we removed 63 interactions from BoolODE's network to satisfy this constraint. These interactions were not removed from SERGIO's network (see Appendix B, Supplementary Table B.4).

3.3.12 Settings of Single-Cell Analysis Tools

In this study we applied several tools to the real or synthetic data sets to mimic real-world analysis of such data and to benchmark these tools. We did not normalize the data sets prior to using these tools unless otherwise specified. We note below the specific settings used for each of the tools we tested:

SC3 [86]: We did not run SC3 to infer the number of cell types, instead we treated the number of cell types as a known quantity and required SC3 to cluster data sets into 9 cell types.

GENIE3 [11] v1.4.3: We provided the identities of true regulators to the GENIE3 tool except when analyzing the differentiation data sets where we used all the genes as potential regulators. Also, for differentiation analysis GENIE3 was run on the exact same expression matrices as used for the other GRN inference tools in this study.

MAGIC [92] v1.10.1: Prior to running MAGIC, we filtered the synthetic data for rare genes (those expressed in less than 5 cells), and performed library size normalization as well as a square root transformation. We used MAGIC with the parameter $t = 2$, $t = 7$, and default setting where t is inferred from data.

SLINGSHOT [89] v1.0.0: We used the first two PCs as a low-dimensional representation of single-cells, and provided these as input to SLINGSHOT, along with the cell type labels. We did not provide any further prior information about origin and end cell types of trajectories.

VELOCYTO [130] v0.17.17: In addition to velocity inference from clean and noisy expression matrices, we used Velocyto to pre-process and normalize data sets. In particular, we used Velocyto to filter low-quality cells and normalize the two noisy differentiation data sets, i.e. DS13 and DS14, prior to using Slingshot and GRN reconstruction tools. We performed all of the filtering and normalization steps for spliced and unspliced counts that are recommended by developers of the software. We removed all cells whose total unspliced count is smaller than the 70^{th} percentiles of unspliced counts for noisy differentiation data sets. We also performed K-nearest neighbor imputation on 20-dimensional PCA representations of single cells with $K = 5$ for clean and $K = 400$ for noisy differentiation data sets. The "constant velocity" model was employed for inferring velocity fields, and square root transformation was used for estimating transition probabilities from PCA representation of clean data sets, while log-transformation was used for noisy data sets.

SCODE [139]: For each differentiation trajectory, we used SCODE with D = 2,4,6,8, and 10 to infer regulatory relationships and report the results that produce the highest AUROC. Also, for each differentiation trajectory and setting of D parameter, we ran SCODE for 100 iterations and averaged results over 5 trials as recommended by the tool's developers. All genes were considered as potential regulators. The inferred sign of interactions (activating or repressing) was ignored in evaluation of the tool's performance: we sorted all gene-gene interactions by the absolute value of their inferred scores and assessed this ranked list for accuracy. Thus, the reported performance values are an overestimate of GRN inference accuracy in our setting.

SINCERITIES [140] v2.0: For each differentiation trajectory, we used the tool with parameters specifying Kolmogorov-Smirnov distance, Ridge regularization, and no auto-regulatory edge setting for unsigned GRN inference. As for SCODE, performance evaluation ignored signs in the true GRN.

SINGE [141] v0.4.1: For each differentiation trajectory, we executed the tool with $\lambda = 0$, 0.01, 0.1. For each setting of λ, we evaluated the tool on an ensemble of 200 hyper-parameter settings (see Appendix B, Supplementary Table B.2). For each λ, we aggregated the results over its ensemble of 200 parameter settings and reported the result that produced the best AUROC (in clean data sets, all three settings of λ showed equal AUROC, in noisy data sets $\lambda = 0$, 0.01 showed an equal but better AUROC than $\lambda = 0.1$).

Data sets for evaluating GRN inference on differentiation data: For each differentiation simulation data set, we generated sub-matrices that represent cells belonging to a single lineage. Therefore, we obtained 1, 2, and 3 sub-matrices for clean data sets DS9, DS10, and DS11 respectively, and 2 and 3 sub-matrices for noisy data sets DS13 and DS14 respectively. Assignment of cells to different lineages was performed according to the pseudotime inferred by Slingshot, and the assigned sets of cells need not be mutually exclusive (i.e., some single cells might be assigned to more than one lineage). GRN inference was performed for each of the lineages separately.

3.3.13 Technical Definitions

Here, we briefly explain some of the technical terms and quantities used throughout this study:

Total Variation (TV): It is a measure for the distance between two probability distributions. For two probability distributions P and Q over a finite countable set X, total variation is defined as:

$$TV(P,Q) = \frac{1}{2}\|P - Q\|_1 = \frac{1}{2}\sum_{x \in X} |P(x) - Q(x)| \tag{3.44}$$

where $\|.\|_1$ denotes the L1 norm. Note that total variations varies in range $[0 \quad 1]$.

Total Deviation (TD): It is a measure for evaluating the difference between two functions of the same variable. For two bounded continuous function F and G, the normalized total deviation in range $[a \quad b]$ is defined as:

$$TD(F,G) = \frac{1}{b-a}\int_a^b |F(x) - G(x)|dx \qquad (3.45)$$

Note that if F and G are lower-bounded by zero and upper-bounded by h, the normalized total deviation $TD(F,G)$ is also bounded similarly.

Coefficient of Variation (CV): Characterizes the dispersion of data around its mean and is defined as the ratio of the standard deviation to the mean:

$$CV = \frac{\sigma}{\mu} \qquad (3.46)$$

3.3.14 Data and code availability

This study used several published and publicly available data sets. We obtained the raw mouse cortex data set [132] from Gene Expression Omnibus (GEO) with accession GEO: GSE60361. The raw 10X genomics expression for mouse MGE, CGE, and cortical regions [133] obtained from GEO with accession GEO: GSE104156. The raw 10X genomics expressions for human kidney [134] and human PBMC cells [135] obtained from GEO with accession GEO:GSE102596 and GEO: GSE117988, respectively. The raw Drop-seq expression for human lung [136] obtained from GEO with accession GEO: GSE130148. The raw Smart-seq2 expression for mouse heart [96] obtained from GEO with accession GEO: GSE109774. Also, we obtained raw 10X genomics expression data for dentate gyrus of mouse hippocampus from GEO with accession GEO: GSE104323 [137]. SERGIO v1.0.0 used to generate synthetic data sets in this study is available as a python package on GitHub: `https://github.com/PayamDiba/SERGIO`.

3.4 RESULTS

We developed SERGIO to simulate how expression values of a specified number of genes vary from cell to cell under the control of a given GRN, and how such information is captured in modern single-cell RNA-seq data sets. We first simulate "clean" gene expression data based on the GRN and mathematical models of transcriptional processes, including stochasticity of such processes ("biological noise"). We then add "technical noise" to the clean data, mimicking the nature of measurement errors attributed to scRNA-seq technology [142] (Figure 3.1).

3.4.1 Simulation of “clean” data

We generate expression profiles of single cells by sampling them from the steady state of a dynamical process that involves genes expressing at rates influenced by other genes (transcription factors) (Figure 3.1, top). A select few of the genes are pre-designated as master regulators (MRs); these have no regulatory inputs in the GRN and their expression evolves over time under constant production and decay rates (see Methods). Expression of every other gene (non-MR) evolves under a production rate determined by adding contributions from its GRN-specified regulators (equation 3.5 in Methods) and a constant decay rate. Each regulator’s contribution to a gene depends on the former’s current concentration and an interaction parameter (strength of activation or repression) specific to the regulator and regulated gene. This dependence is described by a Hill function [126], thus allowing for non-linear effects.

Each gene’s time course is simulated while incorporating biological noise, using the chemical Langevin equation [120], as adopted in the GeneNetWeaver (GNW) simulator [114]. Once the system of evolving expression profiles reaches steady state, we sample profiles from randomly selected time points. Variation in expression profiles across cells of the same type is assumed to mimic variation across time points in the steady state (the “ergodic assumption” [127]), hence the temporally sampled cells are used as the collection of cells in the synthetic data.

Specifying the fixed production rates of MRs determines the average steady state expression profile of the sampled cells and is used to generate data for a single cell type. In order to synthesize a data set with multiple cell types, the above simulation is performed as many times, with each simulation using a different setting of MR production rates. The aggregate of expression profiles sampled (from steady state) across all simulations forms the “clean” synthetic data set. Due to the underlying simulation model being stochastic, distinct runs of the entire process provide distinct “replicates” of the data set.

The clean expression data resulting from the above-mentioned step form a matrix of continuous values that represent mRNA concentrations of each gene in each cell. Unlike data obtained from RNA-seq technologies, the clean data do not comprise discrete mRNA “count” values; simulating such counts to mimic experimental data involves a further sampling step described below.

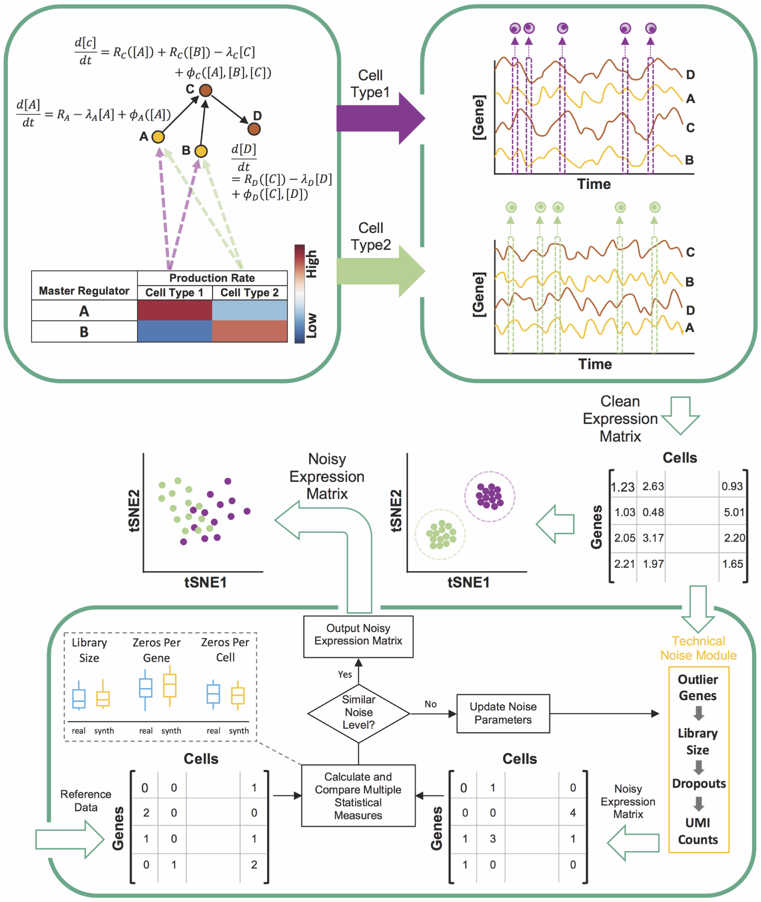

Figure 3.1: Overview of steady-state simulation pipeline. SERGIO uses stochastic differential equations (SDE) to describe the dynamics of mRNA transcripts of each gene (A,B,C,D) in a specified GRN (top left). Each gene's SDE consists of a production rate which is modeled as the sum of contributions the gene receives from its regulators (e.g., from A and B for gene C). Such a contribution is modeled as a regulatory function (R_{gene}) of the concentration of the TF except for "master regulators" (genes without regulators) for which the production rate is a constant (e.g., R_A). Also, each SDE contains a term representing the decay of mRNA transcripts (e.g. $\lambda_c[C]$) and a term representing biological noise (e.g., $\phi_c([A],[B],[C])$). A cell type is specified by the production rates of MRs, and SERGIO performs separate simulations for each cell type using these MR production rates. It initializes the concentration of genes to their estimated averaged steady-state concentrations and continues simulations in steady-state region for all genes simultaneously to generate time-course expression data (top right). Finally, it samples single-cells from the time-course uniformly at random over the steady-state region and outputs the "clean" expression matrix. A cartoon illustration of the clean data generated by SERGIO (after dimensionality reduction) shows single-cells tightly clustered by cell type. Bottom panel: Clean expression matrix is fed into the technical noise module. Parameters of this module are manually tuned so that the noise level in simulated data is similar to that in a user-selected reference ("reference") data set. Multiple statistical measures are used to compare the noise level between reference and simulated data sets. Upon adding technical noise, cells of different types become less well-separated but are still distinguishable by clustering algorithms.

3.4.2 Incorporation of technical noise

In the second phase (Figure 3.1, bottom), we use the clean data to simulate integer-valued "count" data, as are produced in current scRNA-seq technologies, by sampling from a Poisson distribution whose mean is the real-valued expression level. However, prior to this conversion, the real-valued expression data matrix (genes x cells) is operated upon by modules that incorporate three different types of technical noise – outlier genes, library size effects, and dropouts (see Methods). The statistical details of these modules are borrowed from the Splatter simulation tool ([107]) and re-implemented in SERGIO. A user-provided real single-cell data set is used as a reference for adding technical noise. In particular, parameters of the technical noise modules are iteratively tuned until a level of noise comparable to that in the real data is achieved (Figure 3.1). Comparison of noise levels between the simulated noisy data and the provided real data is performed using multiple statistical summaries of the two data sets, as explained in the next section.

It is worth noting here that several existing single-cell expression simulators employ a probabilistic model whose parameters are directly estimated from a real data set, and then sample synthetic data from the model. This approach is not feasible in SERGIO since the true GRN underlying the real data set is unknown and notoriously hard to reconstruct, and the explicit use of a GRN is a crucial distinguishing feature of SERGIO. As such, SERGIO uses a randomly generated GRN to first synthesize clean expression data, and uses the real data set only in the second phase, to determine the extent of technical noise to add to the clean data.

3.4.3 SERGIO simulates realistic data sets

We used SERGIO to generate eight synthetic data sets under three different settings of the underlying GRN (Table 3.1, Network IDs 2-4). These three settings use GRNs with 100, 400 and 1200 genes, that were sampled from real regulatory networks in E. coli or S. cerevisae (Table 3.1, also see Appendix B, Supplementary Figure B.1 for graphical representation of the extracted networks). The motivation for this sampling is not to mimic expression data from these species, but to use a realistic regulatory network for simulations. All of the eight simulations included 300 cells for each of 9 cell types, for a total of 2700 single cells. Each data set was synthesized in 15 "replicates" by re-executing SERGIO with identical parameters multiple times.

For each of the simulated data sets, we configured SERGIO to introduce technical noise to an extent that matches published real scRNA-seq data sets. Our goal was to compare data generated by SERGIO against various scRNA-seq technologies including Illumina HiSeq2000, Drop-seq, Illumina 10X chromium and Smart-seq. We matched data sets DS1-3 against a published data from mouse brain sequenced by Illumina HiSeq2000 comprising expression profiles of cells that are categorized into nine cell types with high confidence [132]. DS4 was compared against a published single-cell RNA-seq data from Medial Ganglionic Eminences (MGE), Caudal Ganglionic Eminences (CGE), and cortical regions of mouse sequenced by 10X chromium [133], while DS5 and DS6 were respectively compared against data from human kidney [134] and human Peripheral Blood Mononuclear Cells (PBMC) [135], both sequenced by 10X chromium. DS7 was matched against a single-cell RNA-seq data from human Lung [136] sequenced by Drop-seq, and DS8 was matched against single-cell expression from mouse heart obtained from the Tabula Muris Consortium [96], sequenced by Smart-seq2. These comparisons were done through manual iteration of the technical noise parameters (see Methods), and comparison of statistical properties between the synthetic and real data sets, as described next. First, we sampled from each of the real data sets the same number of genes as in the corresponding synthetic data, repeating this 50 times to obtain 50 "replicates" for each of the (sampled) real data sets, each of which was compared to the 15 replicates of the corresponding synthetic data set. All comparisons were performed using synthetic data with or without technical noise, referred to as the "noisy" and "clean" forms of the data set respectively. Note that DS3-8 share the same settings of GRN topology and parameters (Network ID 4), and therefore they all correspond to the same "clean" simulated data.

We used several commonly used summary statistics, reflecting coverage and noise levels in scRNA-seq, to compare each synthetic data set with a matching real data set (Figure 3.2). These include two cell-level statistics – "library size" and "zero count per cell" (number of genes with zero recorded expression in a cell) – and three gene-level statistics – "zero count per gene" (number of cells in which a gene has zero recorded expression), "mean count" and "variance count" (mean and variance of expression of genes). As shown in Figure 3.2A-J, there is a strong qualitative agreement between real and synthetic (noisy) data sets in terms of each of these five statistics. (The noise level used in generating the synthetic data sets shown were obtained after tuning the noise parameters.) This qualitative agreement is consistently observed across different scRNA-seq technologies. As expected, the clean form of each synthetic data set has substantially different statistical properties from real data. (For a more intuitive interpretation of the "total variation" metric used to compare distributions, see Appendix B, Supplementary Figure B.2.).

An empirical observation about scRNA-seq data reported in the literature is that there is an inverse relationship between the number of zeros in the recorded expression of a gene and its mean expression level across cells [143, 144]. This inverse relationship is clearly seen in our (noisy) synthetic data sets and their corresponding real data sets (Figure 3.2K,L), and arises not only because genes with lower expression levels are more likely to result in sampled zero counts, but also because the simulator creates "dropouts" (a form of technical noise) with higher probability for such genes. Similarly, an inverse relationship between the coefficient of variation (CV) – a common measure of expression noise – and mean expression of a gene has been extensively discussed in the literature [145, 146, 147]. Figure 3.2M shows the existence of this relationship in a representative synthetic data set as well as in a corresponding real data set. This inverse relationship is not the result of adding technical noise and is present in the clean synthetic data sets as well (Figure 3.2N). It arises naturally from the gene regulatory model implemented in SERGIO, in contrast to other single cell simulators that explicitly add such a relationship to their statistical sampling procedures [107]. In other words, the synthetic data sets generated by SERGIO not only exhibit realistic distributions of key summary statistics (Figure 3.2A-J), they also exhibit second-order relationships between pairs of variables that are characteristic of real data sets (Figure 3.2K-N).

The simulation capability of SERGIO is not limited to small GRNs sampled from E. coli or S. cerevisiae and it can be used to simulate large mammalian networks also. To illustrate this, we obtained a curated regulatory network for mouse from RegNetwork database [148], which included 15272 genes (43 are master regulators) and 76483 gene-gene interactions after preprocessing. This GRN was used in SERGIO to simulate a single-cell data set containing 3600 single-cells, resulting in a simulated data comparable in size with a real scRNA-seq data of mouse brain [132]. By adding technical noise, we were able to match key summary statistics, CV-vs-mean and zero-vs-mean relationships between the simulated data and mouse brain HiSeq2000 data [132] (Appendix B, Supplementary Figure B.3), with a similar quality of match as that seen in DS1-8.

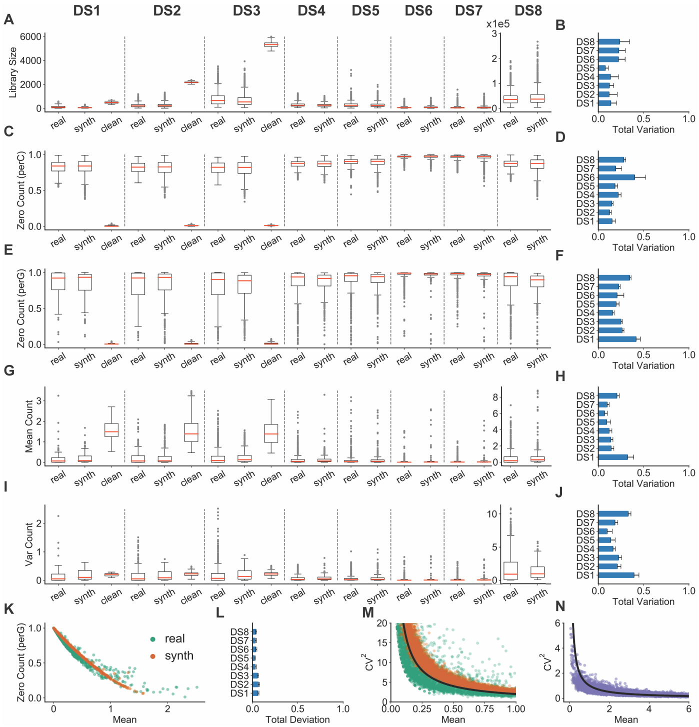

Figure 3.2: Comparisons between synthetic data generated by SERGIO and real scRNA-seq data sets. We show the distributions of per-cell quantities in (A,C), and per-gene quantities in (E, G, I), for DS1-8 separated by dashed lines. These comparisons are shown between one sample from the real data set ("real"), one replicate of clean simulated data ("clean"), and its technical noise-added version ("synth"). DS4-8 have the same underlying "clean" data as DS3 (only shown for DS3). More comprehensive comparisons – between every pair of a noisy simulated replicate and a real sample – are shown in panels to the right: the total variation (see Methods) is calculated to compare the real and synthetic distributions and the average total variation across all such pairs is shown in panels (B, D, F, H, J). (A,B) Distributions and total variation of library sizes. (C,D) Distributions and total variation of zero counts per cell (normalized by number of genes). (E,F) Distributions and total variations of zero counts per gene (normalized by total number of cells). (G,H) Distributions and total variations of genes' mean expression. (I,J) Distributions and total variations of genes' expression variances. (K) Inverse relation between normalized zero count of each gene and its mean expression. Data are shown for one of the simulated replicates of DS3 and one sample from the real data containing 2500 single cells and 1200 genes selected at random. (L) Total deviation (see Methods) is calculated between two curves derived from real and synthetic points shown in (K) repeated for every pair of a noisy simulated replicate and a real sample, and the average total deviation is shown. (M) Inverse relation between squared coefficient of variation (see Methods) and mean expression of genes over all single-cells. Data are shown for one of the simulated replicates of DS3 and one sample from the real data containing 2500 single cells and 1200 genes selected at random. The black line shows an arbitrary function of form $y \sim 1/x$ which completely matches with the observed behavior in both real and synthetic data. (N) The inverse relation of form $y \sim 1/x$ is not a result of technical noise and is also observed in clean simulated data. See also Appendix B, Supplementary Figure B.2.

3.4.4 Simulated data exhibit cell heterogeneity similar to real data

Motivated by the growing use of single cell RNA-seq data to characterize cellular heterogeneity in biological samples, we next asked if the synthetic data sets from SERGIO exhibit heterogeneity similar to real ones. We first used Principal Components Analysis (PCA) to reduce each cell's representation (without any pre-processing on data) to ten dimensions (by using the first 10 PCs). Then the popular tSNE [149] algorithm was used to reduce the 10 dimensional representation of cells into 2 dimensions for visualization. Figures 3.3A,B show such tSNE plots for a representative synthetic data set (in the DS3 setting) in their clean and noisy forms respectively. It is clear that in the absence of technical noise the nine cell types (as specified during simulation) are highly distinguishable, and that the noisy data sets smear this visual separability significantly. In addition to tSNE, we tested an alternative non-linear dimensionality reduction algorithm called Uniform Manifold Approximation and Projection (UMAP) [150] that has been recently shown to outperform tSNE in capturing local and global structures in single-cell data [151]. As above, we applied UMAP on the 10-dimensional PC space of data to obtain a two dimensional representation of cells. Figure 3.3C,D show these representations in clean and noisy versions respectively of the representative data set. A comparison between Figure 3.3B,D reveals the better ability of UMAP to resolve cellular heterogeneity.

However, cell type detection in practice does not rely only on visual separation, and specialized high-dimensional clustering algorithms are being developed for the purpose. One such algorithm is SC3 [86], which has been shown to have high accuracy for the task. It was used by Aibar et al. [85] to cluster mouse cortex cells in the "real data set" of our study [132] and the clusters were found to closely correspond to the true cell types present in the sample (Adjusted Rand Index, ARI, of ∼0.8). If our synthetic data sets exhibit similar levels of cellular heterogeneity as the real set, then we expect SC3-reported clusters to have similar levels of concordance with "true" cell types as known to the simulator. Figure 3.3E shows the composition of nine clusters found by SC3 on the (noisy) synthetic data set visualized in Figures 3.3B,D, in terms of the true cell types present in each cluster. We note that seven of the nine reported clusters predominantly comprise cells of one (distinct) type, and only two of the clusters are of mixed composition, thus suggesting a high accuracy of clustering. To make this observation more formal, we computed the Adjusted Rand Index (ARI) between SC3-reported clusters and true cell types for each of the 15 replicates of the DS3 data set, noting an average ARI of 0.78. We repeated this for each of the 50 sampled subsets of the real data set corresponding to DS3 settings (using prior knowledge of true cell types in these

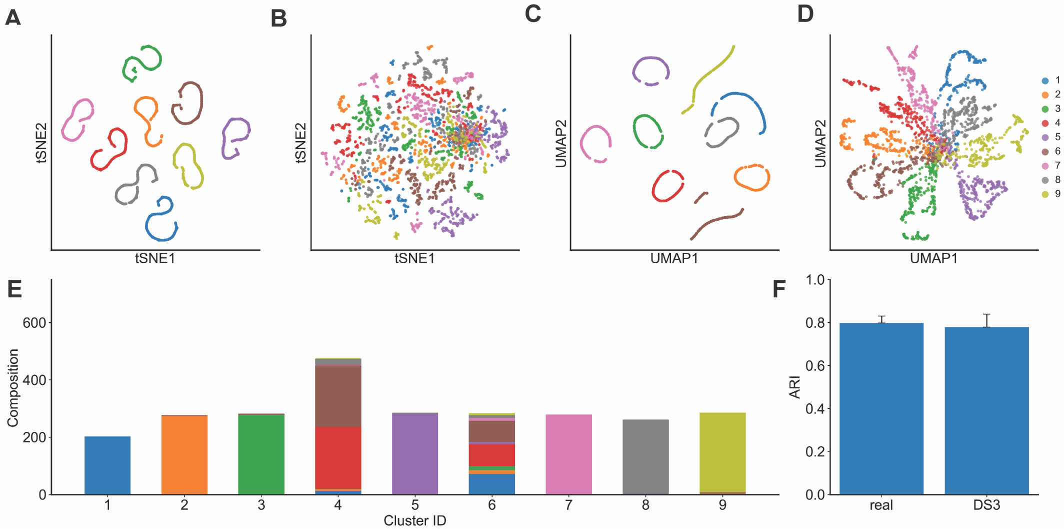

Figure 3.3: Cell heterogeneity in synthetic data generated by SERGIO. (A) tSNE plot of the single-cells in one clean simulated replicate of DS3. All cells of the same cell type are correctly clustered together. (B) tSNE plot of the same data set after adding technical noise. Cells are scattered such that two dimensions of tSNE representation are not sufficient for the human eye to distinguish different cell types. (C) UMAP plot of the clean simulated replicated of DS3 shown in (A). (D) UMAP plot of the noisy data set shown in (B). In contrast to tSNE, cell types are visually distinguishable in the UMAP representation of single-cells. (Data sets were not normalized for library sizes or filtered for rare genes prior to applying tSNE and UMAP.) (E) The noisy data set shown in (B and D) was clustered into nine groups using SC3 clustering method. Cell type compositions of all nine groups are shown, revealing that SC3 can correctly separate 7 of 9 cell types. Clusters 4 and 6 are less homogeneous and comprise a mixture of multiple cell types. (F) SC3 was applied to all 50 real samples (random subsets of the real data set), each containing 2500 cells and 1200 genes, as well as to all 15 simulated replicates of DS3. The adjusted rand index (ARI) was calculated for each clustering task, comparing the SC3 clusters to (known) true clusters defined by cell types and the average ARI is shown for each type of data (real or synthetic). ARI values obtained from simulated data are very close to those observed in real data sets.

data), and found the average ARI to be 0.80, very close to that seen in synthetic data. This exercise demonstrates that synthetic data sets generated by SERGIO exhibit realistic levels of cellular heterogeneity, and also illustrates the use of SERGIO to benchmark clustering methods.

3.4.5 Benchmarking GRN reconstruction methods

A unique aspect of the simulator is that the generated gene expression values, prior to adding technical noise, are the result of direct regulatory influence of transcription factors, and a comprehensive GRN comprising these TF-gene relationships is at the core of its simulations. We next illustrate how this unique feature makes SERGIO-simulated data sets ideal for benchmarking GRN reconstruction tools. In our first tests we worked with clean data sets generated by SERGIO, reasoning that these should provide an upper bound for performance on noisy realistic data sets. We evaluated the popular GRN inference algorithm called GENIE3 [11], which was originally developed for analyzing bulk RNA-seq data but has since been used successfully on single cell data as well. We applied GENIE3 on the (clean) data sets DS1 (100 genes) and DS3 (1200 genes) and evaluated the predicted TF-gene pairs based on the underlying GRNs in these data sets, using the common metrics Area Under Receiver Operating Characteristics (AUROC) and Area Under Precision-Recall Curve (AUPRC). Recall that these data sets were synthesized to include 300 cells for each of nine cell types. To assess the impact of data set size, we created smaller sets by sampling 200, 100 or 10 cells per cell type from the original simulated data (for each replicate of DS1 and DS3), and repeated the GRN reconstruction assessments for these. We also sought to assess the advantage of having single cell resolution in the data, and thus synthesized "bulk" expression data sets by averaging the expression of each gene in all cells of the same type, mimicking a situation where each cell type has been sorted separately and subjected to traditional expression profiling. (The resulting synthetic data sets included nine conditions with "bulk" expression values of each of 100 or 1200 genes, depending on the original data set.)

Figures 3.4A,B show the ROC and PRC respectively for a representative replicate of the DS3 data set, in its original setting (300 cells per type) as well as its sampled smaller versions and their respective "bulk" data set versions. A more comprehensive view, spanning all replicates of DS1 and DS3, is shown in Figure 3.4C-F. Several points are apparent from these figures. First, in nearly all versions of the data sets, GENIE3 performs significantly better than random, as is evident from AUROC values well above the 0.5 value expected from a random predictor. Second, we note that while performance is significantly better on larger data sets than on the smallest data set (10 cells per type), there is not a clear difference among the data sets with 100 cells per type or more. This suggests that, at least in the absence of technical noise, the benefits of greater cell count for GRN reconstruction accuracy saturate at commonly seen data set sizes. Third, the "bulk" data sets consistently yielded lower accuracy than the single-cell data sets, regardless of the numbers of cells, confirming the potential value of single-cell data for regulatory inference. Finally, we noted that although the DS1 and DS3 data sets had similar AUROC values, the AUPRC values revealed significantly worse predictions in the larger (DS3, 1200 genes) data sets. This is expected, in part because the random baseline is lower for DS3 (random AUPRC of 0.002) than for DS1 (random AUPRC of 0.026), but also suggests that high levels of gene co-expression may confound methods such as GENIE3 more for larger data sets.

We next examined the impact of cellular heterogeneity on GRN reconstruction accuracy, using our clean synthetic data sets. For this, we sampled from each replicate of DS1 and DS3 (at their original setting of 300 cells per type) smaller data sets comprising 6, 3 or 1 cell type rather than the 9 cell types simulated. As shown via AUROC and AUPRC measures in Figures 3.4I-L (with representative ROC and PRC curves in Figures 3.4G,H), we found data sets with greater heterogeneity to consistently improve GENIE3 performance, which remained clearly above the random baseline (AUROC of 0.5 and AUPRC of 0.026 and 0.002 for DS1 and DS3 respectively) for all but the "1 cell type" setting. This is expected, since the latter setting includes gene expression variation resulting only from biological noise, and even though extrinsic noise (fluctuations in TF levels reflected in target gene levels [152]) may be exploited to infer TF-gene relationships, such correlations are diluted by the presence of intrinsic gene expression noise in the simulations (see Methods). On the other hand, in settings with 3 – 9 different cell types, the dominant form of expression variation arises from differences in the steady state profiles of the cell types, making regulatory inferences more effective.

We next examined the effect of technical noise on GRN reconstruction. For this, we compared GENIE3 performance on clean and noisy versions of each replicate of DS3 (1200 genes), in the original setting of 300 cells per type as well as a sampled version thereof with 100 cells per type. The complete results are shown in Figures 3.4O,P, with representative ROC and PRC curves shown in Figures 3.4M,N. Both performance metrics (AUROC and AUPRC) deteriorate to levels expected from random prediction when analyzing noisy synthetic data, in contrast to the very high levels seen prior to introducing technical noise. Notably, increasing the number of cells (from 100 per type to 300) does not change our conclusion. Such nearly-random performance of GENIE3 on noisy single-cell expression data has been reported in previous studies conducted based on real as well as synthetic single-cell expression data sets [119, 139]. It also provides support for the need to combine expression-based inference with cis-regulatory data such as TF-ChIP during GRN reconstruction [85, 153].

In light of the above finding, we considered the possibility of using imputation tools specialized for single cell RNA-seq data as a means to improve the signal necessary for GRN reconstruction. We thus utilized the popular imputation tool called MAGIC [92] to pre-

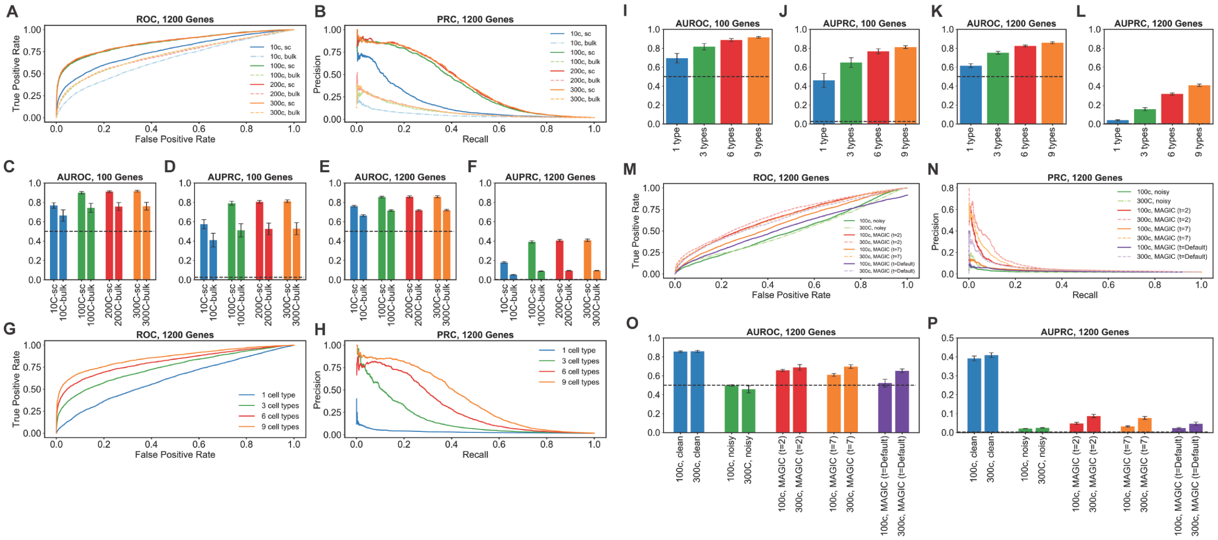

Figure 3.4: GRN inference from synthetic data generated by SERGIO. (A,B) Receiver Operating Characteristic (ROC) curves and Precision-Recall Curves (PRC) respectively of GRN inferred by GENIE3 on one replicate of clean synthetic data (DS3) and various subsets thereof consisting of varying number of cells per cell type. "300c" setting refers to the entire DS3 replicate, including 300 cells per cell type. The other settings ("200c", "100c", "10c") refer to data sets where we sampled 200, 100, or 10 cells respectively from each cell type in DS3. For each data set, we evaluated GENIE3 on single-cell data ("sc" setting), and bulk data ("bulk" setting) which are obtained by averaging expression profiles of all cells of the same type in the corresponding single-cell data set. (C-F) Area under ROC curve (AUROC) and area under PR curve (AUPRC) of the GRN inferred by GENIE3, averaged over all clean replicates of DS1 (C,D) or DS3 (E,F) and their subsets comprising varying number of cells per cell type in both single-cell and bulk settings. (G,H) Similar to (A,B), except that subsets of the clean synthetic data set DS3 were created to have varying numbers of cell types; every cell type retains all of its 300 simulated single-cells in the original DS3 replicate. (I-L) AUROC and AUPRC of GRN inference by GENIE3, averaged overall all replicates of DS1 (I,J) or DS3 (K,L), as well as their subsets comprising varying numbers of cell types; as in (C-F), evaluations were done for the single-cell data set as well as bulk data sets derived from them. (M,N) ROC and PRC of the GRN inferred by GENIE3 on one noisy replicate of DS3 ("300c") with 300 cells per type and a sub-sample of it ("100c") containing 100 cells per type. Also shown are results of GRN inference by the same method on the same data sets, but using data imputation by MAGIC in three different settings ($t = 2$, $t = 7$, $t = default$). (O,P) AUROC and AUPRC of the inferred GRN by GENIE3 on all replicates of the data sets used in (M,N) as well as all the clean replicates of DS3 ("300c, clean" and "100c, clean").

process the noisy synthetic data sets prior to analyzing them with GENIE3, and compared the performance metrics to those obtained above. Results were only modestly improved from those without imputation, with AUROC values ∼0.65 in the 300 cell/type setting and ∼0.52

in the 100 cell/type setting (Figures 3.4M-P). Closer examination revealed that the default settings of MAGIC made the data overly structured, resulting in unrealistically large gene-gene correlations (see Appendix B, Supplementary Figure B.4 and B.5), similar to previous reports [11, 154, 155]. In order to address this issue, we employed two smaller values of the "t" parameter in MAGIC ($t = 2$ or 7), in separate runs, prior to GRN reconstruction. Both of these settings resulted in improved performance over the default setting of MAGIC, and substantially better than that seen in noisy data sets without imputation (Figures 3.4M-P). For instance, AUROC values for the 300 cell/type setting were at ∼0.70 ($t = 7$), squarely in the middle of those without imputation (∼0.46) and those on clean data sets (∼0.86). AUPRC values (∼0.08) were also significantly above random expectation (∼0.002), though far from the high values ∼0.4 observed on clean data sets. Although we noted above that GRN reconstruction accuracy on clean data sets did not improve when increasing the cell counts (300 versus 100 cells per type), we do notice a significant and consistent effect of cell counts in performance on imputed data (Figures 3.4O,P). Presumably, greater cell counts are beneficial for the imputation step, which in turn results in higher performance of GENIE3. Our overall conclusion from the above tests (Figure 3.4) is that a state-of-the-art GRN reconstruction method such as GENIE3 [11] can perform accurately on single cell expression data in the hypothetical scenario where technical noise is absent, but falls to near-random performance in the face of realistic levels of technical noise. The accuracy does improve above random baseline if the data are imputed with specialized tools but remains far short from the upper bar observed in clean data, making technical noise a major factor for future GRN reconstruction methods to address.

SERGIO's simulation model relies on the assumption that the combined effect of multiple regulators is simply the sum of their individual effects. Cooperative regulation by multiple regulators is not considered, so as to reduce the number of parameters that need to be specified and to facilitate the simulation of large regulatory networks where first-order regulatory effects may be known but second-order effects (dependent on pairs of regulators) are mostly unknown. However, to investigate the impact of this simplifying assumption on benchmarks of GRN inference, we conducted a set of simulations using an in-house version of SERGIO that includes cooperative regulation by pairs of activators (see Methods). Upon addition of technical noise, the resulting synthetic data sets were comparable to a real mouse brain scRNA-seq data set [132] in terms of overall statistical characteristics, with the quality of match being similar to that seen in DS1-8 (see Appendix B, Supplementary Figure B.6). Moreover, we confirmed that the performance of GRN inference by GENIE3 is not different between synthetic data sets that do or do not include cooperative regulation. This was observed in our evaluations on clean and noisy (with technical noise) simulated data sets as well as noisy simulated data imputed by MAGIC (see Appendix B, Supplementary Figure B.7).

3.4.6 Simulating single-cell expression dynamics of early T-cell development

T-cell development and the gene regulatory processes that underlie the T lineage dynamics have been extensively studied, most recently through a comprehensive analysis involving bulk and single-cell RNA-seq profiling [156]. Zhou et al. [156] used highly sensitive, sequential single molecule fluorescent in situ hybridization (seqFISH) to profile the expression of 65 marker genes, including important regulators of T-cell development, in 4551 cells belonging to different stages ranging from "early T-cell precursor" (ETP) to committed T-cells. Clustering analysis of these data revealed nine cell clusters, eight of which were associated with one of the developmental stages according to expression of the marker genes. These seqFISH data have far less technical noise than that observed with scRNA-seq technologies, providing us with a unique opportunity to simulate them through the "clean" simulation mode of SERGIO.

To simulate T-cell development expression dynamics as captured by Zhou et al., we first used GENIE3 on the seqFISH data from that study to obtain a GRN containing 108 interactions among the 65 genes (see section 3.4.7 below). We then used this network to fit a regression model for each gene as a function of its regulators prescribed by the GRN, such that its average predicted expression in each cluster matches seqFISH data (see section 3.4.7 below). Regression parameters thus obtained, along with the GRN (henceforth called "parameterized GRN"), were used in SERGIO to simulate nine cell types, each of which was represented by a similar number of cells as in the seqFISH data. Figure 3.5A shows the PC plots of real and synthetic data sets. Note that we used the same cluster IDs and developmental stage labels as in [156]. This plot reveals a qualitative agreement between the simulated and real data in terms of ordering of cell types (stages) in development.

We speculated on the possibility of erroneous inferences in the GENIE3 network due to the small number of genes in the training data set. We therefore repeated the simulations with a literature-based GRN for early T-cell differentiation, reported by Longabaugh et al. [157]. This GRN model contains 22 of the 65 genes present in the seqFISH data, including *Tcf7*, *Bcl11b*, and *Gata3* that are known to be important regulators of T-cell differentiation. The GRN contains 32 interactions involving a total of 10 regulators, of which four are master regulators (as defined above). We used the same regression-based approach as used for GENIE3 network (above) to parameterize this GRN model. Figure 3.5B shows the PC plot of the expression of the 22 genes in seqFISH and simulated data generated by SERGIO using this literature-based GRN model. Again, a good qualitative match between real and synthetic data is observed in terms of the ordering of developmental stages, although the three cell types belonging to DN2 stages (cluster 3-5) are less well separated in the simulated data. Interestingly, we noted that the two-dimensional representation of the data set with 22 genes (Figure 3.5B) shows a correct ordering of stages at the right end of the lineage (DN3a (cluster 0) – DN3a (cluster 6) – DN3b (cluster 7)), while PC representation of the data with all 65 genes (Figure 3.5A) shows an opposite ordering of these stages at the end of the lineage. In the rest of this section, we discuss findings using the simulated data generated using the literature-based 22 gene GRN model.

Figure 3.5C shows the relationship between coefficient of variation and mean of gene expression values across cells of each cluster, in real (teal) and synthetic (orange) data. Also, Figure 3.5D directly compares the mean expression values of genes in each cluster, between real and synthetic data. Both comparisons revealed a reasonable level of agreement between real and synthetic data. Note that our only objective during the simulation was to match a gene's mean expression value between the two data sets using a very simple model of regulation. Future studies may employ more complex optimization strategies to improve the quality of match.

Next, we sought to use SERGIO simulations to evaluate the significance of different regulators in T-cell development. We performed "in silico knockout" of eight regulators including the four master-regulators, i.e. *Tcf7*, *Runx1*, *Hes1*, *Gfi1b* (Figure 3.5E). In order to visualize the effect of knockouts, we projected the single-cell trajectory of each knockout (KO) simulation onto the two-dimensional PC space of the simulated wild-type expression (Figure 3.5E, top left). The most pronounced effect on the trajectory is observed in the knockouts of *Tcf7*, *Bcl11b*, and *Runx1*. *Tcf7* and *Bcl11b* are known to be major regulators of T-cell differentiation [156, 157], and *Runx1* is also known to play a role via up-regulation of *Bcl11b* [157]. In order to accurately quantify the contribution of each regulator in T-cell differentiation, we projected the single-cell trajectory of its KO simulation onto the 10-dimensional PC space of the simulated wild-type data and measured, for each cluster (excluding cluster 8), the Euclidian distance between the cluster centers of the wild-type and KO trajectories. As shown in Figure 3.5F, *Tcf7* has the most prominent effect on the differentiation trajectory, on average, followed by *Gata3*, *Runx1*, *Bcl11b* and *Spi1*. Four of these TFs (*Tcf7*, *Gata3*, *Bcl11b* and *Spi1*) are known to have important roles in T-cell differentiation [156, 157]. Interestingly, KO of *Tcf7*, *Gata3*, and *Spi1* in the network obtained by GENIE3 show smaller effects on the differentiation trajectory compared to the network obtained from Longabaugh et al. (See Appendix B, Supplementary Figure B.8).

In addition to seqFISH profiling, Zhou et al. carried whole-transcript single-cell RNA sequencing using Smart-seq2 as well as 10X Chromium v2 [156]. In all three profiling methods, their clustering analysis consistently revealed an outlier cluster (e.g., cluster 8 in the seqFISH analysis) that was not identified as any of the established differentiation stages [156]. We sought to utilize SERGIO simulations to better understand this phenomenon, in particular the outlier status of cluster 8 in seqFISH data. This cluster lacks expression of *Tcf7* – the key regulator of T-cell development. For our simulations, we defined nine artificial stages (cell types), each matching cluster 8 of real data in its master regulator profile except that *Tcf7* that is gradually overexpressed across different stages. We simulated 300 cells per artificial stage and visualized the simulated single-cell trajectory (Figure 3.5G, upper-left) in the same two-dimensional space as that used for wild-type simulated data (Figure 3.5B, right). Although the overexpression of *Tcf7* causes cluster 8 to migrate toward committed T-cells, the overall trajectory does not resemble the differentiation trajectory of T-cells. We noticed that cluster 8 also lacks expression of *Runx1* – the other important regulator according to our KO analysis above. We performed a similar simulation of artificial stages as above, but now driven by overexpression of *Runx1*, and found (Figure 3.5G, upper-right) that expression of this TF alone is not sufficient for triggering cluster 8 to follow the wild-type differentiation trajectory. A similar result was observed when *Tcf7* and *Runx1* were both overexpressed in a completely correlated manner (Figure 3.5G, bottom left). However, when we induced an overexpression pattern for *Tcf7* and *Runx1* similar to their expressions in the eight established stages (clusters) in the seqFISH data, we observed a developmental trajectory (Figure 3.5G, bottom-right) similar to wild-type T-cell differentiation. This suggests that the down-regulation of *Tcf7* and *Runx1* contribute to the divergence of cluster 8 from the T-cell differentiation program, a hypothesis that merits future experimental investigation.

3.4.7 Details of T-cell differentiation simulations

Simulations of T-cell development involved two important steps: first obtaining a GRN model and second parameterizing the GRN and tuning other parameters of SERGIO. Here we describe these two steps in detail.

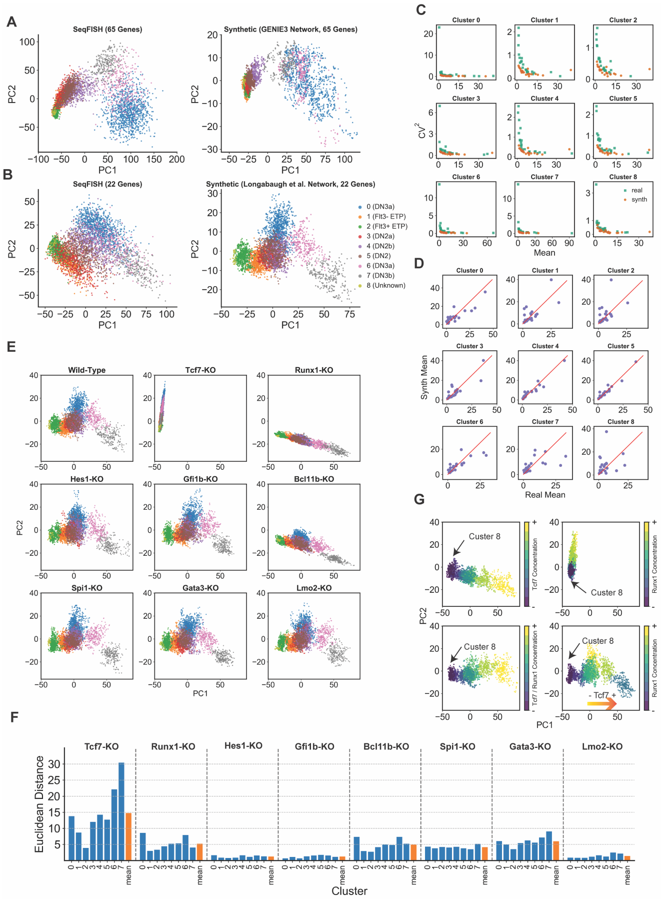

Figure 3.5: Simulations of T-cell differentiation. (A) Left: Principal Components (PC) representation of seqFISH data based on expression of all 65 genes in the data set. Right: PC representation of simulated data using the GRN inferred by GENIE3 on seqFISH data. Clusters are labeled by the IDs (stages) used in the original study by Zhou et al., and cells are colored by their cluster IDs. (B) Left: PC representation of seqFISH data based on expression of 22 genes present in the Longabaugh et al. GRN model. Right: PC representation of simulated data using the GRN obtained from Longabaugh et al. Cells are colored by the same scheme as in (A). (C) Coefficient of variation and mean expression of genes, across cells in a cluster, shown for each of the 9 clusters, for real and synthetic data (corresponding to (B)). (D) Mean expression of genes, across all cells in a cluster, from real and synthetic data (corresponding to (B)), shown for each of the 9 clusters. (E) PC representation of the simulated data generated as in (B) (Wild-type; WT), and upon in silico knockout (KO) of each TF. (The two-dimensional projection is identical in all nine plots.) (F) For each TF, Euclidean distance is computed between a cluster's center in the simulations under that TF's in silico knockout and those under wild-type conditions, in a 10-dimensional projection. This Euclidean distance is shown for each cluster, representing the impact of the TF's KO on the average profile of that cluster. The mean across all nine clusters is also shown. (G) Movement of cluster 8 upon in silico overexpression of Tcf7 (upper-left), or Runx1 (upper-right), both, in fully correlated manner (bottom-left), and with expression of both factors set to their cluster-specific levels in wild-type conditions (bottom-right). See also Appendix B, Figure B.8.

GRN model: We considered two GRN models for regulation of T-cell development. One model was obtained from [157], which provides a GRN model for pre (stages ETP to DN2a) and post commitment (DN2b to later stages) in BioTapestry format. We assembled the part of this GRN that involves genes present in the seqFISH data. GRN edges (regulatory interactions) in the pre and post commitment stages were combined to obtain a single GRN that includes 32 interactions among 22 genes, four of which are master regulators, i.e., do not have regulators in the GRN. Another GRN model was obtained from the sorted list of possible interactions inferred by GENIE3 on the seqFISH data. We considered the top 126 interactions, and since SERGIO requires GRNs to be acyclic, we opened up the cycles by removing an interaction in each cycle, making sure that the removed interaction is not present in the literature-based GRN of Longabaugh et al. This resulted in a GRN containing 108 interactions among the 65 genes in the seqFISH data, of which 23 are modeled as master regulators.

Parameterizing GRN model: Our goal was to parameterize either GRN discussed above such that the simulated data resembles the real seqFISH data. For this, we tuned the interaction parameters such that the mean gene expression values in each cluster match between the real and simulated data. Based on the structure of a GRN, we first separated the master regulators (MR) – genes without any incoming regulatory edge – from “non-MR” genes. For each MR gene i, the production rate is determined only by a basal production rate b_i which is defined per each cluster separately. According to equation 3.12, for each gene i, we computed the basal production rate b_i in each of the clusters (stage) from the real mean expression of gene i in the same cluster, which is known from the seqFISH data, and an assumed decay rate of $\lambda = 0.8$.

$$\lambda_i E[x_i] = b_i \tag{3.47}$$

where $E[.]$ denotes the expectation operator. However, for non-MR genes, the production rate is a function of its regulators' concentrations as shown in equations 3.5-3.7. By using a relatively large value for half-response parameters in all interactions, for a non-MR gene i, equations 3.5-3.7 are simplified to the following equation:

$$P_i = \sum_{j \in A_i} K_{ij} (\frac{x_j^{n_{ij}}}{h_{ij}^{n_{ij}}}) + \sum_{k \in R_i} K_{ik} (1 - \frac{x_k^{n_{ik}}}{h_{ik}^{n_{ik}}}) + b_i \tag{3.48}$$

where A_i and R_i are the set of all activators and repressors of gene i, respectively. Similar to equation 3.13, we can express mean expression values as a function of production and decay rates:

$$\lambda_i E[x_i] = E[P_i] = \sum_{j \in A_i} K_{ij} (\frac{E[x_j^{n_{ij}}]}{h_{ij}^{n_{ij}}}) + \sum_{k \in R_i} K_{ik} (1 - \frac{E[x_k^{n_{ik}}]}{h_{ik}^{n_{ik}}}) + b_i \tag{3.49}$$

Note that this equation holds for each cluster, where expectation is calculated over the cells of that cluster only. So, the expectation appearing on the left-hand side of the above equation can be calculated for every target gene i in each cluster from the seqFISH data. Also, for any fixed value of the Hill coefficient parameter n, the expectations appearing on the right-hand side can be similarly computed for each of the regulators of gene i known from the GRN model. Therefore, with an assumed decay rate (e.g., we used $\lambda_i = 0.8$ for all genes), large enough half-response parameters h, and fixed Hill coefficients n we can solve this regression problem for each target non-MR gene i over the 9 clusters to determine interaction strengths K, and the basal production rate b_i. In contrast to the other simulations in this study, we allowed non-MR genes to have a non-zero basal production rate, shared among all clusters, in order to improve the fitting accuracies.

For both GRN models, we solved these regression problems by minimizing the least square errors. For each target gene i we used the sign of the regression coefficient to determine the role (activator or repressor) of its regulators (i.e., whether the regulator belongs to set A_i or R_i). Also, various values of Hill coefficient n were tested, while requiring the same value to be used for all interactions. The smallest fitting error was obtained with $n = 2$ for the GENIE3 network and with $n = 1$ for Longabaugh et al. network. The obtained interaction parameters and basal production rates for non-MR and MR genes, together with the optimal Hill coefficients and half-response parameters and decay rates used for fitting the model, formed a "parameterized GRN model" that was used by SERGIO to perform simulations. Moreover, a noise amplitude parameter $q = 0.5$ and $q = 1$ was used for all genes in simulations of GENIE3 and Longabaugh et al. GRNs respectively. For each cluster, we simulated the same number of single-cells as in the seqFISH data. Although our simplified optimization strategy produced fitting errors sufficiently small for our down-stream analysis, future studies with SERGIO may employ more sophisticated optimization strategies. This might involve the optimization of all hyper-parameters including n,h,K and q, especially when enough data are available.

3.4.8 Benchmarking differentiation trajectory inference tools

Our analysis so far involved using SERGIO to synthesize steady-state expression profiles representing different established cell types. We were able to use steady-state simulations even for reproducing the dynamics of T-cell differentiation, by utilizing data and knowledge of established cell types located on the differentiation trajectory and simulating those cell types in steady-state. However, this approach is infeasible in the absence of real data and a well-characterized GRN tied to those data. We next sought to demonstrate how to use SERGIO for simulating a differentiation program using any given GRN. SERGIO offers the capability of synthesizing dynamic expression data on a set of genes controlled by a given regulatory network in single cells differentiating along a given trajectory. In this mode the simulator is provided with a differentiation graph whose nodes represent established cell types in a differentiation program and whose edges represent differentiation from the parent cell type to child cell type (Figure 3.6A). The simulator samples expression profiles from the steady state represented by the parent cell type, and then simulates a dynamical process (identical to that described above) that begins with one of these expression profiles and evolves into the steady state represented by the child cell type. It then samples expression profiles from the temporal duration when the cells are transitioning from the initial to final cell type (Figure 3.6B). The entire "clean" data set is synthesized by repeating this simulation process for each edge in the differentiation graph. Technical noise is then added in a manner identical to the steady state simulation mode.

An emerging approach to describe the dynamics of differentiation programs through single-cell expression profiling involves examination of spliced as well as unspliced transcript levels in the data and inferring "RNA velocity" of each cell [130]. To allow synthesizing data sets amenable to such analysis, the differentiation simulation mode uses a variation on the underlying model described above. In particular, it invokes two chemical Langevin equations (CLE) similar to equation 3.1 to generate unspliced and spliced transcript levels (see equations 3.11 and 3.12 in Methods). It reports the simulated expression values as levels of unspliced as well as spliced transcripts, whose sum may be considered the total expression of a gene.

To illustrate these features of the simulator, we generated four synthetic differentiation data sets (DS9 – DS12), each containing 100 genes controlled by the same GRN, but obeying different differentiation graphs – linear (DS9), bifurcation (DS10), trifurcation (DS11) and tree (DS12) (Figure 3.6C). Figure 3.6C also shows the two dimensional PCA plot of the clean total transcriptome (without technical noise added) for the four types of differentiation graphs. It is visually evident that these two-dimensional representations of cells based on their gene expression profiles match their corresponding graphs used in the simulations. We note that the dispersion of cells of each type (end points of each branch of a graph) as well as the width of the differentiation path from one type to another in the clean simulated data can be controlled by user-specified parameters in SERGIO (see Appendix B, Supplementary Figure B.9).

In order to obtain synthetic data comparable in sizes with data sets obtained from single-cell RNAseq technologies, we simulated larger versions of DS10 (bifurcation) and DS11 (trifurcation), containing 6000 cells per cell type. This gives us a clean bifurcation data set containing 100 genes and 24000 single cells (DS13) and a clean trifurcation data set containing 100 genes and 36000 single cells (DS14). Supplementary Figure B.10 (Appendix B) shows the PCA plots of these two data sets, which closely resemble their respective smaller versions (DS10 and DS11) since the same simulation parameters were used for the small and large versions. Next, DS13 and DS14 were used to synthesize noisy expression data for the bifurcation and trifurcation trajectories. We used a recently published 10X genomics single-cell data set representing dentate gyrus of mouse hippocampus [137] as a reference for calibrating technical noise in DS13 (bifurcation). Separately, we employed the mouse cortex data set [132] that we used in steady-state simulations above as the reference for adding technical noise to DS14 (trifurcation). In both cases, similar to our approach in steady-state simulations, we sampled 50 comparison data sets from the real data each having 100 genes randomly selected out the pool of all genes. These sampled data sets were then compared against the noisy data generated by SERGIO to calibrate the level of technical noise (Appendix B, Supplementary Figure B.11). We confirmed once again that with the appropriate parameter settings noisy data sets synthesized can match real data sets in their statistical properties.

In order to visualize the above data sets, we used the preprocessing functions of Velocyto [130] to normalize expression matrices and filter low quality cells from noisy versions of DS13 and DS14, preserving only the top 2999 and 6560 high quality cells in bifurcation and trifurcation trajectories, respectively. We used PCA to reduce the dimensionality of these data to the first 10 PCs, and then applied UMAP to obtain a two dimensional representation of single cells. Figures 3.6D,E show these data in their PCA-based (top two PCs) and UMAP-based representations. Although significant amount of technical noise has been added to the simulated data, the underlying bifurcation and trifurcation trajectories of cells are clearly evident in the noisy versions of DS13 and DS14.

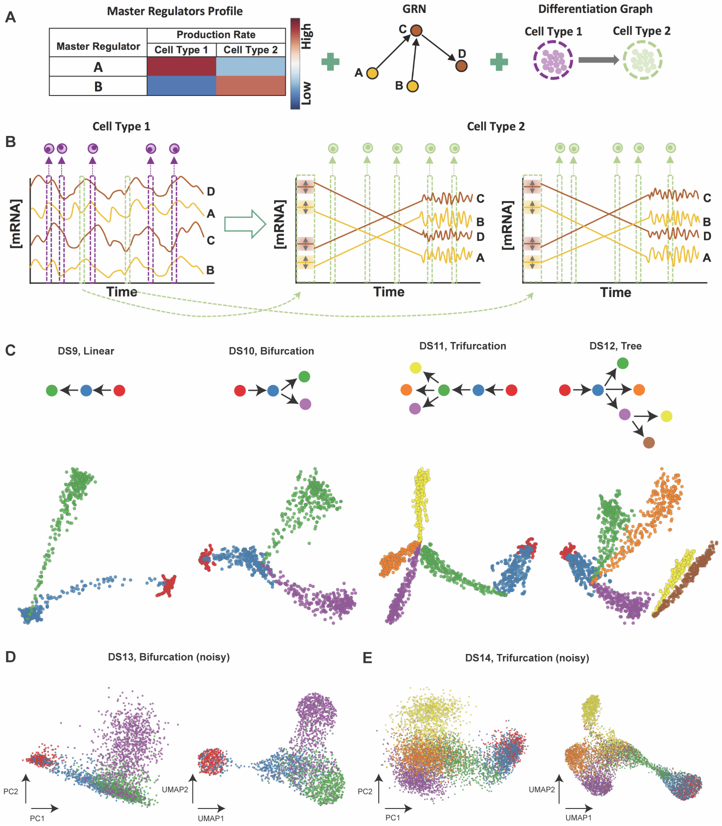

Figure 3.6: Overview of differentiation simulation pipeline. (A) Inputs required for simulation of differentiation programs: in addition to master regulators' profiles defining cell types and the GRN, this simulation mode requires a differentiation graph as an input (right). (B) Differentiation simulation is started from the origin of differentiation trajectory (cell type 1 here). Since the origin cell type is not differentiated from any other cell type, its samples (cells) are drawn from steady-state simulations (left). For each child cell type (cell type 2 in this case), SERGIO initializes transcript levels to values close to their steady-state concentrations in the parent cell type (cell type 1 here). Simulations are then performed so that transcript levels reach their steady-state concentrations in the child cell type, after which simulations continue for a user-defined number of additional steps so as to collect sufficient time-course data in steady-state. SERGIO repeats simulation of the child cell type (from initialization until steady-state) for a user-defined number of times to sample enough paths between the parent and the child cell type. (Two such repeats are shown here.) Finally, single-cells belonging to the child cell type are sampled from the aggregated pool of all time-course data from the initial to the last simulation time point. Temporal fluctuations of expression in the transient region are often negligible in amplitude compared to the overall change in expression from initial to new steady state, hence the transient region is shown with a straight line in this cartoon illustration. (C) PCA plots of single-cell data sets synthesized for differentiation graphs shown at top: DS9 (linear), DS10 (bifurcation), DS11 (trifurcation), and DS12 (tree). Cells of or differentiating into each cell type are shown by a distinct color. (D,E) PCA and UMAP representation of noisy data set DS13 synthesized for bifurcation differentiation graph (D) and noisy data set DS14 synthesized for trifurcation differentiation graph (E). See also Appendix B, Supplementary Figure B.9-B.11.

Differentiation data sets synthesized by SERGIO can be used to benchmark trajectory inference algorithms since the underlying differentiation trajectory (graph) is known for these data. To illustrate this, we applied the Slingshot [89] tool on the clean as well as noisy data sets synthesized based on bifurcation and trifurcation trajectories. Slingshot is a tool specifically developed for trajectory inference, with published reports of high accuracy. For the clean data sets DS10 (bifurcation) and DS11 (trifurcation), Slingshot infers the correct lineages (Figures 3.7A,B); however, it did not fully reconstruct the underlying trajectories for the noisy data sets DS13 and DS14, failing to separate one of the lineages in either case (data not shown). On the other hand, once we provide prior information about the undetected terminal cell types to Slingshot, it correctly infers the trajectories for the noisy data sets DS13 and DS14 (Figures 3.7A,B).

We also analyzed the above synthetic data sets with the Velocyto [130] tool, which infers an “RNA velocity” field in a low dimensional representation of single cells that indicates the direction in which each cell’s expression profile appears to be changing. The velocity field also provides an intuitive visualization of differentiation trajectories. Figures 3.7C,D depict the inferred velocity fields for clean as well as noisy data sets with bifurcation or trifurcation trajectories, demonstrating how Velocyto correctly captures these differentiation trajectories. We found that for the more complex differentiation trajectory of DS12 (tree) Slingshot is unable to recover correct lineages, while Veloycto infers a velocity filed that is indicative of the correct underlying trajectory (Figure 3.7E). Thus, we find that use of an additional layer of information – separation of spliced and unspliced mRNA counts – can improve trajectory inference from single cell transcriptomic data. This is not limited to data sets with complex underlying trajectories – Figure 3.7F shows an example data set (“DS15”) generated using a simple bifurcation graph for which Slingshot infers a linear trajectory while Velocyto reports a velocity field clearly indicative of the true bifurcation trajectory. It is worth noting that here we did not provide any prior information regarding terminal cell types to Slingshot, which may resolve the errors noted above. To summarize, synthetic data sets generated by SERGIO show that, at least in the absence of prior information on established cell types, RNA velocity-based approaches may have an advantage in terms of trajectory inference on single cell data.

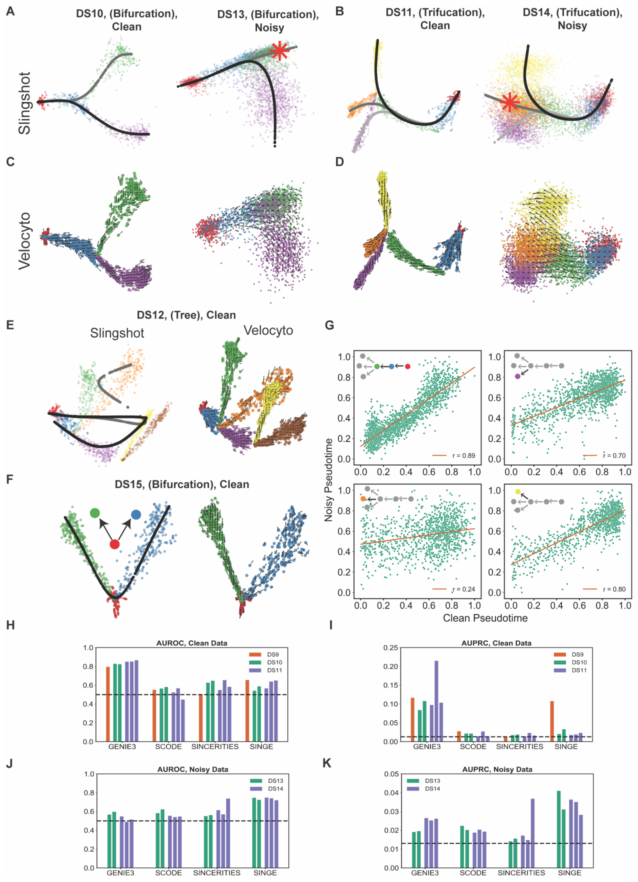

Figure 3.7: Evaluation of differentiation data sets generated by SERGIO. (A,B) Differentiation trajectories inferred by Slingshot on clean and noisy simulated data for bifurcation (A) and trifurcation (B) trajectories of differentiation programs. Each line with a slightly different grayscale color denotes a distinct inferred path. Slingshot infers correct trajectory without any prior knowledge for clean data, and with knowledge of one of the terminal cell types (green cell type in DS13, and orange cell type in DS14 marked by asterisks) for noisy data. (C,D) Velocity fields inferred by Velocyto on clean and noisy simulated data for bifurcation (C) and trifurcation (D) differentiation trajectories. In all four cases, the inferred velocity field is consistent with the underlying differentiation trajectory. (E) Differentiation trajectory inferred by Slingshot (left) and the velocity field inferred by Velocyto (right) on clean simulated expression data set DS12. (F) Differentiation trajectory inferred by Slingshot (left) and the velocity field inferred by Velocyto (right) on a simple bifurcation data set (DS15) synthesize by SERGIO. (G) Pseudotime inferred by Slingshot from noisy versus clean simulated data of DS14 (see Appendix B, Supplementary Figure B.10 for clean version of DS14) on four separate segments of the underlying differentiation trajectory. (H,I) AUROC and AUPRC respectively of the GRN inferred by various methods on the pseudotime-ordered single cells in clean data sets DS9, DS10, and DS11. GRN inference was performed on each differentiation branch separately and AUROC and AUPRC is calculated and shown for each branch of DS10 and DS11. (J,K) AUROC and AUPRC respectively of the GRN inferred by various methods on the pseudotime-ordered single cells in noisy data sets DS13 and DS14. GRN inference was performed on each differentiation branch separately and AUROC and AUPRC is calculated and shown for each branch.

Trajectory inference in differentiation data sets allows researchers to assign a (partial) ordering among single cells along the differentiation trajectories, resulting in assignment of a so-called "pseudotime" value to the cell [158]. We noted above (Figures 3.7A,B) that differentiation trajectories can be inferred with reasonable accuracy using Slingshot, but also observed that the reduced dimension representation places individual cells more diffusely along the trajectory when analyzing noisy data sets than for clean data sets. This suggests that the technical noise present in single cell data may affect the inferred temporal ordering (pseudotime) of cells. To quantify this effect, we simulated clean and noisy data for a trifurcation trajectory similar to DS9 and used Slingshot to assign pseudotime labels to cells in the two data sets (These two versions have synthetic expression data for the same cells, differing only in the presence of technical noise.) Figure 3.7G depicts the correlation between these two pseudotime labels, separately for four segments of the underlying differentiation trajectory. We noted that for three of the four segments the pseudotime inference is relatively robust to presence of technical noise (correlation coefficient r being 0.89, 0.70 and 0.80), but for one of the segments – the lineage leading specifically to the cell type shown in orange in Figure 3.7D – the pseudotime inferences on clean and noisy data sets were poorly correlated ($r = 0.24$). We noted that the dropout rate was higher for cells in this lineage (see Appendix B, Supplementary Figure B.12) compared to the three other lineages, providing a plausible explanation for the above observation and suggesting that the pseudotime inference on cells with high dropout rate may need to be interpreted with greater caution.

3.4.9 Benchmarking GRN reconstruction on differentiation data

Single-cell transcriptomic profiles of differentiation processes offer unique opportunities for GRN reconstruction, since pseudotime labels can be exploited to infer causal relationships between TFs and target genes. Several methods have been recently proposed that specifically channel this opportunity, including SCODE [139], SINCERITIES [140] and SINGE [141]. We used the differentiation data simulated by SERGIO to benchmark these specialized GRN-reconstruction algorithms, using Slingshot for pseudotime inference. In particular, we used one simulated replicate of clean data sets DS9 (linear), DS10 (bifurcation) and DS11 (trifurcation) and two noisy data sets DS13 (bifurcation) and DS14 (trifurcation) for which we verified above that Slingshot infers correct trajectories. For each data set, we evaluated and compared the three above-mentioned GRN reconstruction methods on single cells associated with a single branch of the inferred differentiation trajectory (see Methods). We also used GENIE3 as a baseline method to infer TF-gene relationships without utilizing pseudotime information. Interestingly, in the absence of technical noise, GENIE3 clearly outperforms the three specialized algorithms in five out of six evaluations (Figure 3.7H, I). However, for DS9 (linear) SINGE outperforms SCODE and SINCERITIES, and performs as well as GENIE3 in terms of AUPRC. In general, the use of temporal ordering of single cells does not seem to help GRN reconstruction in the absence of technical noise. This result is consistent with findings of Pratapa et al. [138], where GENIE3 was placed among the top three GRN inference methods evaluated, above SCODE, SINCERITIES, and SINGE, when applied on synthetic data sets based on curated Boolean networks and without technical noise. Interestingly, the authors found that GENIE3 is among the best performing GRN inference methods even when evaluated on real scRNA-seq expression data sets.

On the other hand, for noisy data sets DS13 and DS14, performance of GENIE3 (in terms of AUROC) falls down to random levels (Figure 3.7J, K) similar to what we observed for steady-state data sets. Here, SINGE clearly outperforms the other methods, including GENIE3, in four out of five evaluations, and in the fifth evaluation both SINCERTIES and SINGE show equally strong performance. Interestingly, the performance of SINGE here is significantly above random and is even better than its performance on the clean data sets DS9-11, at least in terms of AUROC. This suggests that SINGE is robust to technical noise present in the single-cell RNA-seq technologies. Next to SINGE, GENIE3 has the best overall performance in terms of AUPRC, followed by SCODE and SINCERITIES. The same overall performance order among the last three methods was reported by Pratapa et al. [138] in evaluations on real scRNA-seq data sets (SINGE was excluded from these evaluations in [138]). In four out of our five evaluations, performance of SINCERITIES in terms of AUPRC is worse than the other methods and is close to random. This is also consistent with evaluations of this tool on real scRNA-seq data sets by Pratapa et al. [138].

3.5 DISCUSSION

The main distinguishing quality of SERGIO is its ability to simulate single-cell expression data based on a specified GRN. Its implementation strikes a balance between a biologically realistic model of transcriptional processes and simplifying assumptions that facilitate fast simulation, capable of scaling to thousands of genes and regulatory interactions. To mimic cellular heterogeneity commonly seen in single-cell data, SERGIO employs an intuitive definition of cell types as steady states of GRN dynamics (Huang et al., 2005). The steady-state assumption is admittedly a simplification, and in reality, some genes in a cell type might have out-of-equilibrium expression states. However, this simplification allows for a more robust benchmarking of single-cell tools that do not examine cell state transitions and differentiation information. On the other hand, SERGIO can also simulate collections of cells differentiating from one cell type to another, an important feature not available in GNW (Schaffter et al., 2011) even after modifications to simulate single-cell data.

We showed that with a reliable and properly parameterized GRN, SERGIO can reproduce the expression dynamics of early T-cell development, capturing the significant regulatory effects of key regulators of T-cells, such as *Tcf7*, *Gata3*, and *Bcl11b*, and suggesting an important role for *Runx1*. These observations were made using a well-studied GRN model we attained from the literature (Longabaugh et al., 2017), but could not be fully reproduced when using a GRN reconstructed with GENIE3 (see Appendix B, Supplementary Figure B.8). This suggests that SERGIO can be utilized to examine alternative networks of transcription regulation in light of single-cell expression data and prior knowledge, and to rank or exclude possible interactions. In addition, we showed how SERGIO can be used to predict the broader effects of a perturbation in the GRN model.

Furthermore, we demonstrated that SERGIO is a powerful tool for benchmarking a wide variety of single-cell analysis tools. For instance, our assessment of a leading GRN inference tool found that it is rendered largely inaccurate (close to random performance) due to technical noise typical of contemporary data sets, even though it is capable of far greater accuracy in the absence of measurement errors. We also evaluated GRN inference methods designed specifically for time-ordered single-cell expression data (Deshpande et al., 2019; Matsumoto et al., 2017; Papili Gao et al., 2018), and found that in the absence of technical noise, a more general-purpose method – GENIE3 (Huynh-Thu et al., 2010) – outperforms these specialized methods; however, SINGE (Deshpande et al., 2019) shows the best performance when technical noise is present. Future studies can use SERGIO to study the effect of technical noise on GRN inference. Moreover, the performance of these specialized tools depends on the type of differentiation trajectories, number of single-cells and other factors. For example, SCODE has a hyper-parameter named D, whose appropriate value is not known *a priori* and might vary from one data set to another. Similarly, SINGE uses a hyper-parameter named *lambda* to control the sparsity of the inferred network. It is common for GRN inference tools to resort to user-defined hyper-parameters, and future studies on GRN inference can utilize SERGIO to examine such hyper-parameters as a function of data set properties.

An important work related to ours is the BoolODE simulator developed by Pratapa et al. (Pratapa et al., 2020), which adapts the model of GeneNetWeaver (GNW), but allows the user to provide a Boolean function to describe combinatorial influence of multiple regulators on each gene, thereby making the GNW model more configurable. SERGIO, on the other hand, simplifies the GNW model in its treatment of combinatorial regulation, modeling the influence of multiple regulators as the sum of their independent contributions. This difference has the following practical consequences: (1) The time complexity of simulating a target gene in SERGIO is linear in the number of regulators while that in BoolODE is exponential in this number. We found SERGIO to run significantly faster than BoolODE for the same networks (see Methods and Appendix B, Supplementary Table B.4); (2) When simulating data sets from a random network, the rules of combinatorial regulation are much simpler to specify in SERGIO than in BoolODE (influence of multiple TFs is additive in SERGIO, while BoolODE requires combination rules to be explicitly specified for each target gene). The SERGIO model also ignores protein translation and degradation, which are featured in BoolODE, thus marking another difference between the simulators in their tradeoffs between model simplicity and realism. We believe the simplifications made in SERGIO to be a practical advantage since large GRNs are rarely characterized in the necessary detail. Secondly, stochastic expression in BoolODE arises only from the biological noise term in the GNW model, and a dropout rate. SERGIO on the other hand incorporates multiple sources/types of noise beyond the biological noise, viz., outlier expressions, library size effects and dropout through appropriate statistical distributions discussed in the literature (Zappia et al., 2017). Thirdly, the dynamic mode of SERGIO enables simulations of spliced and unspliced mRNA counts for user-defined differentiation trajectories (a feature not included in BoolODE currently), which allows the benchmarking of RNA velocity and trajectory inference algorithms.

Recent work has also examined the related but distinct task of learning a generative model from a given scRNA-seq data set, to be then used for simulations. Marouf et al. (Marouf et al., 2020) employed a neural network to automatically learn the underlying distributions of gene expression from a real single-cell data set and used the learnt model to generate synthetic expression profiles (cells) that are indistinguishable from real profiles. Interestingly, they showed that this machine learning approach captures gene-gene dependencies in its latent space, therefore implicitly including regulatory relationships in the model. Our work differs fundamentally from (Marouf et al., 2020) in that we seek to simulate data with an explicit GRN as input (a forward simulation goal), rather than attempt to estimate it from data (a reverse engineering goal). This key difference allows SERGIO to be useful for benchmarking of GRN inference tools.

It should be noted that the GRN benchmarking in this study considered methods based on expression only, while better accuracy can result from existing tools that use additional information such as TF-DNA binding data (Aibar et al., 2017). Future work can combine SERGIO simulations of single-cell expression with existing ideas on benchmarking GRN inference from bulk data and prior information (Siahpirani and Roy, 2017). Expression data from TF knockout experiments can also be exploited by GRN inference algorithms (Bonneau et al., 2006), and knockout of regulators can be easily simulated in SERGIO to assess such algorithms.

In conclusion, we believe that SERGIO will prove useful to a number of researchers developing tools for the rapidly developing field of single-cell transcriptomics. It will be especially useful for testing GRN reconstruction methods, which according to our assessments is the analytical task most in need of future improvements. But its usefulness will extend to future tools for other popular tasks as well, since synthetic data sets that capture real data more closely naturally provide more reliable assessments of those tools. Moreover, the “clean” simulated data sets (without technical noise) generated by SERGIO should be useful in their own right, since they also capture realistic expression variation due to biological noise and can provide upper bounds on accuracy in the idealized scenario where measurement noise has been eliminated.

CHAPTER 4: INFERENCE OF DIFFERENTIAL CAUSAL GENE NETWORKS

This chapter is largely based on a manuscript that at the time of writing this thesis is in preparation for submission to a journal. This manuscript is co-authored by Saurabh Sinha.

4.1 ABSTRACT

The discovery of causal relationships from high-dimensional data is a major open problem in bioinformatics. Machine learning and feature attribution models have shown great promise in this context but lack causal interpretation. Here, we show that a popular feature attribution model estimates a causal quantity reflecting the influence of one variable on another, under certain assumptions. We leverage this insight to implement a new tool, CIMLA, for discovering condition-dependent changes in causal relationships. We then use CIMLA to identify differences in gene regulatory networks between biological conditions, a problem that has received great attention in recent years. Using extensive benchmarking on simulated data sets, we show that CIMLA is more robust to confounding variables and is more accurate than leading methods. Finally, we employ CIMLA to analyze a previously published single-cell RNA-seq data set collected from subjects with and without Alzheimer's disease (AD), discovering several potential regulators of AD.

4.2 INTRODUCTION

A key challenge in bioinformatics today is to extract causal relationships from omics data. A common approach involves calculating statistical associations between pairs of variables, such as between expression levels of two genes [159], between alleles and phenotype [160] or gene expression [161], etc. Pairwise associations are error-prone due to the presence of confounding variables [162]. This has led to multivariable regression and machine learning (ML) models that learn a mapping between the target variable (e.g., gene expression) and multiple potentially causal variables or "features" (e.g., expression levels of transcription factors) [163, 164, 165, 166]. After model training, various approaches are used to extract the importance of each feature to the model [164, 166, 167], a process informally known as "interpretation" of the model. Model interpretation is straight-forward in case of linear models, where the coefficient of each feature serves as its importance score, but is challenging for non-linear ML models (e.g., Random Forests and Convolutional Neural Networks) with intense on-going efforts towards improvement of “local” [168, 169, 170, 171] (for each sample) and “global” [172, 173, 174, 175] (across all samples) feature attribution models.

The ultimate goal of model interpretation in bioinformatics is to infer “causal” relationships that can provide biological hypotheses testable via experimental perturbation. Indeed, the search for causal relationships is a hallmark of much of natural and social sciences, and successful paradigms include interventions in randomized control trials (RCT) [176] and temporal data modeling [177]. However, causality is not explicitly addressed in the above-mentioned approaches to ML model interpretation. ML models themselves are already recognized as tools for counterfactual reasoning [178, 179], but the feature attribution models used with them lack a formal causal interpretation, focusing instead on quantifying the importance of a variable to the ML model’s output [169, 180, 181]. We believe this is a significant conceptual gap in the modern “interpretable AI” movement and has led to confusion about the relative strengths and weaknesses of different feature attribution models [182]. Our work is an attempt to bridge this gap and use the resulting insights to address an important bioinformatics goal.

In recent times, a popular formalism for causal relationships and counterfactual inference with observational data has emerged in the form of Pearl’s do-calculus [183]. It provides a mathematical framework to clearly describe causal relationships and, when possible, infer their strengths from data. This is referred to as calculating the value of a “causal estimand” from data. On the other hand, a recent breakthrough in model interpretation is the “SHapley Additive exPlanations (SHAP)” feature attribution model [180], which adopts a game theoretic approach to provide a measure of local feature importance in an ML model. SHAP unifies and improves on an important group of existing feature attribution models. Our main theoretical contribution is to show that the SHAP score estimates a causal quantity defined in Pearl’s causal inference framework. We define this causal estimand and present the conditions under which it is estimated by the SHAP score.

Building on the newfound intuition about the SHAP score as a measure of causal influence, we propose a framework to quantify how much a feature’s influence on a target variable differs between two data sets. The causal interpretation of SHAP scores is critical here, as it enables us to directly compare these scores between models trained on different data sets. We present a tool called CIMLA (**C**ounterfactual **I**nference by **M**achine **L**earning and **A**ttribution Models) that implements this framework for inferring differential associations. CIMLA can be a powerful tool for a variety of bioinformatics challenges where one seeks to identify biological relationships that have changed between conditions (such as disease and healthy, treatment and control, etc.). To demonstrate this, we use the CIMLA score to recover regulatory relationships that differ between transcriptomics data sets from two conditions, a problem also known as differential gene regulatory network ("dGRN") reconstruction [184].

The common approach to dGRN reconstruction is based on pairwise co-expression analysis to identify regulatory relationships in each data set, followed by tests of significance of the change in their regulatory strength between the data sets (reviewed by Bhuva et al. [184]). To address the well-known limitations of pairwise association analysis, multivariable linear regression models are sometimes used [185, 186], with comparison of corresponding regression coefficients to infer differential regulation. Other authors have explored joint modeling of two data sets in search of shared as well as exclusive regulatory associations [187, 188]. All of the above strategies have relied on linear modeling of gene expression as a function of other genes' or transcription factors' expression. Yet, it is known that GRN reconstruction (on a single data set) benefits from incorporating non-linear dependencies, and leading methods rely on ML approaches such as Regression Random Forests (GENIE3 [11]), or Gradient Boosting Regression (GRNBoost2 [189], ENNET [190]) to achieve cutting-edge performance on standard benchmarks. Thus, it is natural to ask if dGRN reconstruction can be improved by training an ML model on either data set and comparing feature importance scores between the two models. This is exactly the task that CIMLA is designed to perform, making it an ideal test case for the new measure of differential feature importance.

To summarize, we make the following main contributions in this work. First, we define a quantity that measures the causal influence of a variable on an outcome, in the presence of other causal covariates and confounders, and show that under certain assumptions this quantity is estimated by the SHAP score of Lundberg et al. [168, 180]. Second, we present the CIMLA score to measure changes in feature importance between two observational data sets. It is based on the SHAP score and thus inherits its desirable feature attribution properties, while also having an intuitive causal interpretation for changes in influence. Third, we systematically assess CIMLA for the task of dGRN inference, using synthetic benchmarks based on state-of-the-art expression simulators, and demonstrate its advantages over existing methods, especially in scenarios where conditional differences in GRN are confounded by other changes. Finally, we employ CIMLA to study the dGRN underlying Alzheimer's disease (AD), using one of the largest published single-nucleus RNA-seq (snRNA-seq) datasets for AD, and find two known regulators of AD – CREB3 and NEUROD6 – to undergo substantial changes in their respective regulons.

4.3 MATERIALS AND METHODS

4.3.1 A causal estimand for detection of differential regulatory relationships

We present a formulation of GRN reconstruction in the language of causal inference, with the ulterior goal of defining a differential regulatory edge as a causal influence that changes between two groups of samples. For simplicity, the GRN will be reconstructed one target gene at a time, i.e., we will attempt to identify the TFs regulating a given gene; repeating the process for each gene furnishes the full complement of TF-gene regulatory edges in the GRN.

Suppose there are m candidate TFs denoted by indices $M = \{1..m\}$, and the variables $\{X_t\}_{t\in M}$ represent the expression levels of these TFs. Let Y represent the expression of the target gene. Informally, the causal inference problem is to quantify the causal influence of each covariate X_t on Y. The common approach to this is to employ the "average treatment effect (ATE)":

$$ATE_t = E[Y|do(X_t = 1)] - E[Y|do(X_t = 0)] \tag{4.1}$$

where $do(.)$ represents Pearl's "*do-operator*"[191]. In the common case, $do(X_t = 1)$ denotes a binary "treatment" (intervention) such as administering a drug to an individual, while $do(X_t = 0)$ denotes a "control" condition such as administering a placebo. The two expectations are taken over the same population of individuals. This formula also bears resemblance to how a TF's influence on a gene is assessed by a biologist: e.g., the expression of the gene upon knockout of the TF is compared to the wild-type expression, which may loosely be considered as the "$do(X_t = 0)$" and "$do(X_t = 1)$" conditions, respectively. The average effect over biological "samples" (e.g., different cells or different biospecimens) is interpreted as the strength of the TF's regulatory (causal) influence on the gene. We note the similarity of this approach to that of Xing and ven der Laan[192].

The above formulation presents an immediate challenge in our context: knocking out a TF t will typically lead to other TFs' levels being affected as well, and an effect may be seen on the target gene due to one of these other TFs regulating the gene, i.e., an indirect effect. However, a TF-gene edge in a GRN is expected to represent direct causal influence, as implemented via the TF binding to enhancers associated with the gene, and indirect effects should not be included. The use of TF binding sites as evidence of direct regulation is the ideal solution to this problem, but not considered here since our goal is to reconstruct GRNs from expression data alone, without access to cis-regulatory information. Thus, we reformulate the above formal definition of the TF's influence on a gene as follows. We first define a "local" version of the ATE that quantifies the effect of a covariate X_t on Y for an arbitrary biological sample $x = \{x_j\}_{j \in M}$. At this point, x does not necessarily refer to a sample in any given observational data set. This "Local Treatment Effect" or "LTE" is given by:

$$LTE_t(x) = \\ E[Y|do(X_t = 1), do(X_{j \in M \setminus \{t\}} = x_{j \in M \setminus \{t\}})] - E[Y|do(X_t = 0), do(X_{j \in M \setminus \{t\}} = x_{j \in M \setminus \{t\}})] \tag{4.2}$$

where the intervention "$do(X_t)$" is now accompanied by additional interventions "$do(X_{j \in M \setminus \{t\}})$". In other words, to estimate the causal effect of TF t on the target gene in a particular biological sample characterized by TF levels $X = x$, the above definition demands that all TFs other than t be fixed at their levels seen prior to the intervention on X_t. We note that such a precise intervention is not likely to be straight-forward or even feasible in practice, but the re-definition of equation 4.2 will help clarify the causal inference problem mathematically.

Another challenge with the initial formulation of equation 4.1 is the use of a binary "treatment" variable to represent the regulatory action of a TF, whose effect on a gene depends, possibly non-linearly, on its concentration in the cellular context. To address this, we redefine LTE (equation 4.2) to be parameterized by $\hat{x}_t$, a baseline level of TF t, as follows:

$$LTE_t(x, \hat{x}_t) = \\ E[Y|do(X_t = x_t), do(X_{j \in M \setminus \{t\}} = x_{j \in M \setminus \{t\}})] - E[Y|do(X_t = \hat{x}_t), do(X_{j \in M \setminus \{t\}} = x_{j \in M \setminus \{t\}})] \tag{4.3}$$

where the effect on the gene in a sample is quantified by the difference between setting X_t to the level observed in the sample (x_t) versus setting X_t to the baseline level ($\hat{x}_t$). Note that the baseline level may be greater than or less than x_t, corresponding to experimental knockdown or overexpression, respectively, of a TF. Since a single reference value $\hat{x}_t$ may be hard to specify or justify *a priori*, we marginalize it out over a suitable probability distribution $P(\hat{X}_t)$ that may, for instance, be learnt from data. The final definition of the local treatment effect thus becomes:

$$LTE_t(x) = E_{\hat{x}_t \sim P(\hat{X}_t)}[LTE_t(x, \hat{x}_t)] \tag{4.4}$$

We show in the next section how $LTE_t(x)$ can be estimated from a given observational dataset. The procedure will allow LTE estimation for an arbitrary x, not only those present in the dataset.

Our goal is to define and calculate the change of a TF's causal effect on a gene between two conditions (e.g. case and control), represented by a binary variable C. To this end, we first define the change in LTE, i.e. "differential LTE", for a single biological sample x as follows:

$$\Delta_t(x) = LTE_t(x|C = 1) - LTE_t(x|C = 0) \tag{4.5}$$

where $LTE_t(x|C = c)$ is the estimated LTE in condition c. Finally, we define the change in a TF's causal influence on a gene for a given population of biological samples x by aggregating over differential LTE for samples in that population:

$$\Lambda_t = \sqrt{E[\Delta_t(x)^2]} \tag{4.6}$$

i.e., root mean square of changes ($\Delta_t(x)$) in all samples of the population. This is the causal estimand we define as the measure of change in a TF's regulatory effect on a gene, between two conditions.

4.3.2 A procedure for estimating Λ_t from observational data

We assume we are given observational data on the target gene expression Y and the TF expression levels $\{X_i\}_{i \in M}$ in a collection of biological samples (e.g., individual cells or biospecimens), for conditions $C = 1$ and $C = 0$. Let D_c denote the data for condition $C = c$. We present here a procedure that calculates $LTE_t(x|C = c)$ for a given $t \in M$, based on the observational data D_c.

First, we use dataset D_c to train a machine learning model f_c capable of predicting Y from $X = \{X_i\}_{i \in M}$, i.e., $f_c(X) = E_c[Y|X]$, where $E_c[.]$ denotes expectation over the distribution underlying dataset D_c. We then calculate the SHAP value [168, 180] $\phi_t(f_c, x)$ that quantifies the contribution of covariate X_t to the model's output at $X = x$. We show (see below) that under certain assumptions, $\phi_t(f_c, x)$ provides an estimate of $E_{\hat{x}_t \sim P_c(\hat{X}_t)}[LTE_t(x, \hat{x}_t)]$, i.e., the SHAP score estimates $LTE_t(x|C = c)$ for the dataset D_c.

We use the above procedure to estimate $LTE_t(x|C = c)$ for $c = 0$ and $c = 1$ separately (using separate datasets and respective machine learning models f_0 and f_1), and thus obtain $\Delta_t(x)$ of equation 4.5. We finally estimate Λ_t as the root-mean-squared value of $\Delta_t(x)$ over samples x of a dataset. Throughout this study, we use only the samples $x \in D_c = 1$, to estimate the expectation of squared differential LTEs in equation 4.6, but future studies might consider using data from both groups. Although it is common to aggregate local SHAP values using a "mean-absolute" function [168], we argue that "root-mean-square" aggregation is more useful in our context (see below).

4.3.3 SHAP score estimates a causal quantity

As above, assume there exists an observational dataset of $m + 1$ variables consisting of a set of covariates (i.e., features) $X_M = \{X_i; i \in M = \{1..m\}\}$ and an outcome variable Y. We also assume that there exists an unobserved confounder variable Z', that is causally associated with the covariate set X_M (Figure 4.1), but that there are no other unobserved confounders that are associated with the observed covariates X_M, but there are no other unobserved confounders that are associated with the observed covariates X_M and outcome Y (Appendix C, Supplementary Figure C.2). Moreover, we allow for causal associations between covariates, with unknown directionality. For each $i \in M = \{1..m\}$, we will assume there exists a causal association between X_i and Y and attempt to quantify it (Appendix C, Supplementary Figure C.2, solid arrow), while the associations between $X_{M\setminus\{i\}}$ and Y are not assumed known (Appendix C, Supplementary Figure C.2, dashed arrows). With these assumptions, we consider the estimand:

$$\alpha_i(x, \hat{x}_i) \equiv LTE_i(x, \hat{x}_i) = \\ E[Y|do(X_i = x_i, X_{M\setminus\{i\}} = x_{M\setminus\{i\}})] - E[Y|do(X_i = \hat{x}_i, X_{M\setminus\{i\}} = x_{M\setminus\{i\}})] \quad (4.7)$$

where $\hat{x}_i$ is an arbitrary baseline value of X_i. Estimation of $\alpha_i(x, \hat{x}_i)$ is challenging as the underlying causal diagram is not fully resolved. To address this, we borrow ideas from causal discovery [193, 194], enumerating all possible causal structures that are consistent with the underlying causal diagram (Appendix C, Supplementary Figure C.2), computing $\alpha_i(x, \hat{x}_i)$

for each structure ψ and taking an average. Thus, the estimand of equation 4.7 is modified to the following:

$$\alpha_i(x, \hat{x}_i) \equiv LTE_i(x, \hat{x}_i) \equiv E_\psi[LTE_i(x, \hat{x}, \psi)] \tag{4.8}$$

where the expectation $E_\psi[.]$ is taken over a suitable distribution over structures ψ and the notation $LTE_i(x, \hat{x}_i, \psi)$ is introduced to refer to LTE as defined in equation 4.3 but under a particular causal structure ψ:

$$\begin{aligned} &LTE_i(x, \hat{x}, \psi) = \\ &\qquad E[Y|do(X_i = x_i, X_{M\setminus\{i\}} = x_{M\setminus\{i\}}), \psi] - E[Y|do(X_i = \hat{x}_i, X_{M\setminus\{i\}} = x_{M\setminus\{i\}}), \psi] \end{aligned} \tag{4.9}$$

There are 2^{m-1} distinct causal structures, in all of which feature X_i is causally associated with the outcome Y and in each of which a subset of features $S \subseteq M\setminus\{i\} \quad (0 \leq |S| \leq m-1)$ is causally associated with Y. Let $A(\psi)$ denotes the subset S of features (excluding feature X_i) associated with Y in causal structure ψ and $NA(\psi)$ denotes the subset $M \setminus (\{i\} \cup S)$ of features that are not associated with Y. To simplify equation 4.9, we use two lemmas proved in Appendix C (Supplementary Notes C.1.2 and C.1.3). Using Lemma 4.1 we can simplify equation 4.9 as follows:

$$\begin{aligned} &LTE_i(x, \hat{x}_i, \psi) = \\ &\qquad E[Y|do(X_i = x_i, X_{A(\psi)} = x_{A(\psi)}), \psi] - E[Y|do(X_i = \hat{x}_i, X_{A(\psi)} = x_{A(\psi)}), \psi] \end{aligned} \tag{4.10}$$

Using Lemma 4.2 we can further simplify the causal estimand in equation 4.9 into a statistical quantity:

$$LTE_i(x, \hat{x}, \psi) = E[Y|X_i = x_i, X_{A(\psi)} = x_{A(\psi)}] - E[Y|X_i = \hat{x}_i, X_{A(\psi)} = x_{A(\psi)}] \tag{4.11}$$

Using law of total expectation, the above equation can be written as:

$$\begin{aligned} LTE_i(x, \hat{x}, \psi) = E_{X_{NA(\psi)}}&[E[Y|X_i = x_i, X_{A(\psi)} = x_{A(\psi)}, X_{NA(\psi)}]]- \\ &E_{X_{NA(\psi)}}[E[Y|X_i = \hat{x}_i, X_{A(\psi)} = x_{A(\psi)}, X_{NA(\psi)}]] \end{aligned} \tag{4.12}$$

Given an ML model $f(x)$ that was trained on the data to estimate $f(x) = E[Y|X = x]$, we can further simplify this equation as follows:

$$LTE_i(x, \hat{x}, \psi) = E_{X_{NA(\psi)}}[f(X_i = x_i, X_{A(\psi)} = x_{A(\psi)}, X_{NA(\psi)})] - E_{X_{NA(\psi)}}[f(X_i = \hat{x}_i, X_{A(\psi)} = x_{A(\psi)}, X_{NA(\psi)})] \quad (4.13)$$

Seperately, as in equation 4.4, we marginalize the reference value $\hat{x}_i$ in $\alpha_i(x, \hat{x}_i)$ defined in equation (4.8), over a distribution $P(\hat{X}_i)$ to define the estimand:

$$\alpha_i(x) \equiv E_{\hat{x}_i \sim P(\hat{X}_i)} \alpha_i(x, \hat{x}_i) = E_{\hat{x}_i \sim P(\hat{X}_i)} E_\psi LTE_i(x, \hat{x}_i, \psi) = E_\psi E_{\hat{x}_i \sim P(\hat{X}_i)} LTE_i(x, \hat{x}_i, \psi) \quad (4.14)$$

where the last equality is obtained by reordering the expectations. If we further assume that a suitable distribution of reference value $\hat{x}_i$ depends on the causal structure ψ, and in particular, that $P(\hat{X}_i) = P(X_i | X_{NA(\psi)})$, equation 4.14 simplifies to:

$$\begin{aligned} \alpha_i(x) = E_\psi E_{\hat{x}_i \sim P(X_i | X_{NA(\psi)})} LTE_i(x, \hat{x}_i, \psi) = \\ E_\psi[E_{\hat{x}_i \sim P(X_i | X_{NA(\psi)})}[E_{X_{NA(\psi)}}[f(X_i = x_i, X_{A(\psi)} = x_{A(\psi)}, X_{NA(\psi)})] \\ - E_{X_{NA(\psi)}}[f(X_i = \hat{x}_i, X_{A(\psi)} = x_{A(\psi)}, X_{NA(\psi)})]]] \\ = E_\psi[E_{X_{NA(\psi)}}[f(X_i = x_i, X_{A(\psi)} = x_{A(\psi)}, X_{NA(\psi)})] \\ - E_{X_i | X_{NA(\psi)}} E_{X_{NA(\psi)}}[f(X_i, X_{A(\psi)} = x_{A(\psi)}, X_{NA(\psi)})]] \end{aligned} \quad (4.15)$$

By combining the latter two expectations $E_{X_i | X_{NA(\psi)}} E_{X_{NA(\psi)}}$ into an expectation under the joint probability we obtain:

$$\alpha_i(x) = E_\psi[E_{X_{NA(\psi)}}[f(X_i = x_i, X_{A(\psi)} = x_{A(\psi)}, X_{NA(\psi)})] - E_{X_{\{i\} \cup NA(\psi)}}[f(X_{\{i\} \cup NA(\psi)}, X_{A(\psi)} = x_{A(\psi)})]] \quad (4.16)$$

An obvious choice for the probability distribution over ψ is the uniform distribution, i.e., $P(\psi) = \frac{1}{2^{m-1}}$. However, we use $P(\psi) = \frac{1}{m\binom{m-1}{|A(\psi)|}}$, i.e., all causal structures where the same number of covariates $k = |A(\psi)|$ in $X_{M \setminus \{i\}}$ are causally associated with Y together receive a probability of $1/m$ and each such causal structure receives equal probability. With this choice of $P(\psi)$, we obtain:

$$\alpha_i(x) = \sum_{A(\psi)\subseteq M\setminus\{i\}} \frac{1}{m\binom{m-1}{|A(\psi)|}} (E_{X_{NA(\psi)}}[f(X_i = x_i, X_{A(\psi)} = x_{A(\psi)}, X_{NA(\psi)})] -$$
$$E_{X_{\{i\}\cup NA(\psi)}}[f(X_{\{i\}\cup NA(\psi)}, X_{A(\psi)} = x_{A(\psi)})]) \quad (4.17)$$

which is equivalent to the local SHAP value of the i^{th} feature in model $f(X)$ at $X = x$ under the adjustment proposed by Janzing et al. [168, 182] (See Appendix C, Supplementary Notes C.1.4 and C.1.5), i.e., $\alpha_i(x) = \phi_i(f, x)$. Thus,

$$\phi_i(f, x) = \alpha_i(x) = E_{\hat{x}_i\sim P(\hat{X}_i)}[\alpha_i(x, \hat{x}_i)] = E_{\hat{x}_i\sim P(\hat{X}_i)}[LTE_i(x, \hat{x}_i)] \quad (4.18)$$

assuming the modified definition of $LTE_i(x, \hat{x}_i) \equiv E_\psi[LTE_i(x, \hat{x}_i, \psi)]$ (equation 4.8) as an average over causal structures ψ. Therefore, SHAP[168] provides an estimate for LTE defined in equation 4.4 under the following assumptions:

1. A machine learning model, f, can reliably predict $f(x) = E[Y|X = x]$.
2. Averaging over the 2^{m-1} defined causal structures ψ with $P(\psi) = \frac{1}{m\binom{m-1}{|A(\psi)|}}$ reliably estimates LTE, i.e., $LTE_i(x, \hat{x}_i) = E_\psi[LTE_i(x, \hat{x}_i, \psi)]$.
3. $P(X_i|X_{NA(\psi)})$ is a reasonable distribution for the reference value $\hat{x}_i$.

Additionally, *ignorability* (or *unconfoundedness*, i.e., there exist no important unmeasured confounders) and *positivity* (i.e., under any settings of X_i, the distribution of the remaining covariates has a similar support), which are the two fundamental assumptions of causal inference [195], are implicitly assumed.

4.3.4 Aggregating local SHAP values into a global score

The average of absolute local SHAP values, i.e. $E_x[|\phi_i(x, f)|]$, is commonly used to obtain global SHAP scores [168]. However, here we adopted a root-mean-square aggregation, i.e. $\sqrt{E_x\phi_i(x, f)^2}$ – a second moment of SHAP values. We motivate this through a discussion of linear models: when f is a linear model, under the feature independence assumption, we have [180]

$$\phi_i(x, f) = w_i(x_i - E[X_i]) \quad (4.19)$$

where w_i is the coefficient of the i^{th} covariate in the trained model. The root-mean-square of these SHAP values, assuming all the covariates are normalized to have a unit variance, is:

$$\sqrt{E_x \phi_i(x,f)^2} = \sqrt{w_i^2 var(X_i)} = |w_i| \tag{4.20}$$

which is commonly used as the global feature importance in linear models (assuming normalized features). By extension from this specific case, we aggregate local CIMLA scores $\Delta_t(x)$ (equation 4.5) into a global measure using root-mean-square. We train ML models using normalized features with unit variance and employ equation 4.6 to obtain $\Lambda_t = \sqrt{E[\Delta_t(x)^2]}$.

4.3.5 Synthetic data generation with SERGIO

We used our previously published single-cell RNA-seq data simulator, SERGIO [76], to generate all synthetic datasets in this study. We first obtained a published GRN from yeast through GeneNetWeaver [114, 116], comprising 400 genes of which 20 are master regulators (MR, genes with no upstream regulators) and 17 are non-MR regulators (a total of 37 regulators) and 1155 regulatory interactions. We removed a subset of regulatory interactions in this GRN at random to obtain two sub-GRNs $\mathcal{G}_{C=1}$ and $\mathcal{G}_{C=0}$, corresponding to case and control, sharing n% of their edges. We repeated this process to obtain 15 GRN pairs with extent of shared edges ranging from $n = 43\%$ to $n = 94\%$. All other parameters necessary for simulation of expression data from these GRNs were selected according to our previously published instructions associated with SERGIO [76]. Each GRN pair was simulated in two settings: low-confounding and high-confounding, as follows. SERGIO simulations require MR profiles as input. An MR profile reflects the expected steady-state expression of MRs in a "cell type". To impose the low-confounding setting, we assumed that simulated cells belong to only one cell type that has the same statistical characteristics (MR profile) in case and control simulations. To achieve this, we generate both case and control profiles, $MR_{C=1}^{low-conf}$ and $MR_{C=0}^{low-conf}$, in one cell-type by sampling all MR levels from the same distribution (specifically, a uniform distribution over the same predefined range for both case and control). On the other hand, to impose a high-confounding setting, we assumed that single cells belong to 10 different cell types that are distinct in case and control simulations. Therefore, we devised two profiles, $MR_{C=1}^{high-conf}$ and $MR_{C=0}^{high-conf}$, in 10 cell-types by sampling MR levels from two distinct distributions in case and control (specifically, uniform distributions over distinct ranges in case and control). Each of the 15 GRN pairs were separately simulated using the MR profiles in the two confounding settings to obtain 15 pairs of single-cell expression profiles in low-confounding and 15 pairs of profiles in high-confounding settings with each profile containing 3000 simulated cells.

To generate noisy datasets, we selected one of the GRN pairs (sharing 94% of their edges) and their corresponding simulated profiles in the high-confounding setting and added various levels (10%, 20%, ..., 70%) of dropout using the technical noise module of SERGIO [76]. For each dropout level, we generated five simulated replicates by repeatedly using the SERGIO's noise module.

4.3.6 Comparing global CIMLA scores of different target genes

Global CIMLA scores, Λ_t, for different genes are derived from separate pairs of ML models trained on each gene. This complicates the comparisons between Λ_t's of different genes, which are required when prioritizing top differential regulatory edges accross all target genes. To facilitate such comparisons across different genes, we standardize target gene expressions to have zero mean and unit variance prior to training ML models. Considering the "local accuracy" property of SHAP values [180], i.e. $f(x) - E[f(X)] = \sum_{i=1}^{m} \phi_i(x, f)$, and the standardization that enforces $E[f(X)] \approx E[Y] = 0$, local SHAP values sum to $f(x)$, which is distributed with a zero mean and unit variance for all genes. This makes the local SHAP values, and hence their difference between case and control groups, as well as their root-mean-square aggregate roughly comparable between different genes. Although this does not provide a mathematical guarantee for comparability of global CIMLA scores of different genes, our benchmarking results on clean simulated datasets suggest reasonable comparability in practice. However, when data are noisy, especially when the noise characteristics are not identical for different genes (for instance, dropout in single-cell RNA-seq has a stronger impact on lowly expressed genes compared to highly expressed ones) the comparability assumption can be weakened. To address this, we also employed a per-gene analysis in our synthetic data evaluations where we assessed the differential regulatory relationships of each gene separately from the others. Similarly, for analysis of the AD dataset, we employed a per-gene approach by relying on a background distribution of global CIMLA scores for each gene (see below) to extract the differential regulatory edges of that gene separately from the other target genes.

4.3.7 Pre-processing of AD snRNA-seq data

The snRNA-seq data we used in this study was obtained from [196] and profiles 48 individuals who were assigned a score between 1-6 representing the final clinical diagnosis of cognitive status at time of death ("cogdx" score). We assigned individuals with a cogdx score of 1 or 2 (no to mild cognitive impairment) to the control group and individuals with a cogdx score of 4 or 5 (Alzheimer's disease as the primary cause of dementia) to the AD group. We excluded the remaining individuals with a cogdx score of 3 or 6 from this study since we could not confidently assign them to control or AD groups. Finally, the AD group consists of 30853 single-cells from 22 individuals and the control group consists of 34852 single-cells from 22 individuals. Single-cell expression profiles in each group were separately imputed using MAGIC (with $t = 2$)[92] after a library-size normalization and a "square-root" transformation (as recommended by MAGIC [92]). Genes with low variance in imputed expression (in the bottom 5% in terms of variance over cells of both conditions) were excluded from the target gene sets in the downstream analysis.

4.3.8 Background distribution of CIMLA scores

In our analysis of AD snRNA-seq data we relied on a background distribution of CIMLA scores reflecting a scenario where we expect no differential regulations. For this, we randomly shuffled the cells between AD and control groups to reconstruct two groups with the same size as the original groups. For each gene and each of the ML model types (RF and NN) we applied CIMLA on this randomly shuffled dataset to obtain a background distribution of global CIMLA scores over all TFs. The maximum background score, representing the TF-gene edge with the highest CIMLA score, for a gene and ML type was used as a threshold to filter the differential regulatory edges for that gene and ML type obtained from the original data. For each target gene, a differential regulatory edge whose CIMLA score survives this threshold was assigned a "dGRN score" of $-log(r/|TFs|)$ where r denotes the rank of that edge in the sorted list of edges for the gene, and $|TFs|$ denotes the total number of TFs which is the same for all genes and ML types.

4.3.9 DEGs in snRNA-seq data

The original analysis of snRNA-seq data [196] published DEGs in each of the six identified cell-types by comparing cells in AD and control. We used the union of the published DEGs in each cell type as a comprehensive DEG set in this study.

4.3.10 Constructing PsychGRN

We used “GRN1” that was published by PsychENCODE project [197] and is available from PsychENCODE resource website (`http://resource.psychencode.org/`). This GRN includes TF-gene edges such that the TF has direct evidence of binding site on the *cis*-regulatory elements of the target gene. TF-gene edges are weighted by the coefficient of the elastic-net regressions trained to predict the target genes expression using the expression of their identified regulators [197]. We constructed PsychGRN by extracting the top 20% of TF-gene edges with the highest absolute edge weights (i.e., regression coefficient). Finally, edge weights in PsychGRN were converted to percentile for better presentation in Figure 4.5F.

4.3.11 Details of CIMLA runs

We used two machine learning models in the “ML module” of CIMLA, random forests (RF) and fully connected neural networks (NN). To train RF, CIMLA first performs 3-fold cross-validation (on training data) to select the best hyper-parameter values. These hyper-parameters included the number and maximum depth of decision trees in evaluations on simulated data; and additionally, the “maximum feature” in the case of noisy simulated data. In our analysis of real snRNA-seq data the RF hyper-parameters tuned included the number of decision trees and their maximum depth, minimum samples in leaf nodes and maximum number of leaves. The NN model in CIMLA is a fully connected multi-layer perceptron with 2 (in real data analysis) or 3 (in simulated data analysis) hidden layers and ReLU activation. Additionally, in real data analysis, a dropout layer with $P = 0.5$ was used following the input layer. We used mini-batch training (of size 128) and ADAM optimizer[50] to train the NN models. All ML models were trained in the regression setting using the Mean Square Error loss. For both simulated and real datasets we randomly selected 80% of cells in each group for training ML models, except for training RF on clean and noisy datasets simulated in the high-confounding setting where we used 90% of the data for training. Expressions of the target genes and TFs are normalized to have a zero mean and unit variance prior to training ML models and other downstream analyses. Moreover, when the target gene is also present in the TF list, CIMLA randomly shuffles its corresponding feature column (over cells) to decorrelate the target variable and its corresponding TF column.

CIMLA uses scikit-learn [198] for cross-validation and training RF, and TensorFlow [199] for training NN models. Also, for explaining RF and NN models CIMLA uses TreeSHAP

[168] and DeepSHAP [180], respectively. In the case of simulated datasets, we computed and aggregated local CIMLA scores over all training data in one of the groups. For real data analysis we randomly sampled 75% of the training data in AD group for computing and aggregating local CIMLA score.

4.3.12 Details of dGRN tools compared to CIMLA

Co-expression-based methods: All co-expression methods we used in this study [200, 201, 202, 203, 204] were tested using their corresponding implementation in dcanr package [184]. Scores for TF-gene pairs were extracted from the gene-by-gene score matrix outputted by these methods. We used the absolute value of TF-gene scores to produce rankings over TF-gene pairs for calculations of performance metrics. The alternative option for producing rankings is to use $-log(.)$ of adjusted p-values generated by dcanr package which we did not pursue as it results in poorer dGRN inference performances for nearly all co-expression methods and simulated datasets tested (data not shown). EBcoexpress [205] is among the top-performing methods in Bhuva et al. study [184] but was excluded from our analysis as they reported that dGRNs recovered by EBcoexpress and z-score methods are highly comparable.

BoostDiff: For every simulated dataset, BoostDiff [206] was tested using case and control expression data with a hyper-parameter setting of 100 estimators, 10 features, and 500 subsamples. The true TF list was used as input for regulators in all runs of the tool. Current version of BoostDiff (v0.0.1) outputs two dGRNs corresponding to two different settings of targeting case and control data. For every TF-gene pair, we extracted their maximum score from the two dGRNs and used it to build a ranking over all TF-gene pairs for calculations of performance metrics.

GENIE3: This method [11] was used to infer separate GRNs in case and control conditions. For every simulated dataset, GENIE3 was separately run on each condition using gene expression data for that condition. For a fair comparison with other methods, the true TF list was used as an input to this method. Weights of TF-gene pairs outputted by GENIE3 in case and control conditions were used for dGRN inference (GENIE3-diff). A TF-gene pair "*tf,g*" with GENIE3 weights of $w_{C=1}$ and $w_{C=0}$ in case and control conditions, receives a differential regulation score as follows:

$$GENIE3 - diff = |w_{C=1} - w_{C=0}| \tag{4.21}$$

GENIE3-diff scores provide a ranking over TF-gene pairs which reflects their importance in the final dGRN and is used to calculate performance metrics.

4.3.13 Data and code availability

CIMLA is freely available as a Python package at: `https://github.com/PayamDiba/CIMLA`. The single-cell transcriptomic data for Alzheimer's disease used in this study was provided by The Rush Alzheimer's Disease Center (RADC) and was obtained from Synapse (`https://www.synapse.org/#!Synapse:syn18485175`) under the doi 10.7303/syn18485175. All simulated data used in this study (prior to imputation by MAGIC) are available at: `https://github.com/PayamDiba/CIMLA_data`

4.4 RESULTS

4.4.1 Overview of CIMLA

Our goal is to identify features (variables) whose association with a measurable trait changes between two populations, using observational data. The two populations may differ in terms of some condition such as disease treatment status, and we seek to quantify the impact of this varying condition on the feature-trait associations. For instance, in the context of dGRN inference the features are expressions of a set of transcription factors (Figure 4.1A, $X = \{X_i\}_{i \in \{1..m\}}$ and the trait of interest is a target gene's expression (Figure 4.1A, Y). The observational data comprise expression profiles ($X = \{X_i\}_{i \in \{1..m\}}, Y$) of multiple samples in each condition, and the task is to detect if an association between say X_t and Y changes between the two populations. We call such an association a "differential association". There are likely to be unobserved confounders (Figure 4.1A, Z') that may lead to spurious detection of differential associations. CIMLA approaches differential association detection from a causal inference perspective, for enhanced robustness toward confounders. We present here a procedural overview of CIMLA, with accompanying details included in Methods. The terminology here is tailored to the dGRN inference problem but the procedure applies broadly to other applications of differential association detection.

Step 1 ("ML module"): Given transcriptomic data from two populations (Figure 4.1B), CIMLA first trains an ML model to predict a target gene's expression (i.e., outcome variable) as a function of TFs' expression (covariates), separately for each population (Figure 4.1C). Formally, let D_c denote the data for the population under condition $C = c$, $c \in \{0, 1\}$.

This step trains, for each D_c, an ML model f_c capable of predicting Y from X. The current CIMLA implementation relies on random forest (RF) [207] and Neural Networks (NN) with dropout [208] to better tackle multi-collinearity among covariates. (Gradient Boosting is also supported but it was not tested here.)

Step 2 ("Interpretation module"): Next, trained ML models and the data are passed to an interpretation module to assess the contribution of each covariate to the predicted outcome under the two ML models, locally at each sample (Figure 4.1D). Formally, it calculates, for either model f_c ($c \in \{0, 1\}$), the SHAP value [168, 180] $\phi_t(f_c, x)$ that quantifies the contribution of covariate X_t to the model's output at $X = x$. This step relies on TreeSHAP [168] and DeepSHAP [180] to estimate local contributions for RF and NN models respectively. This step is where CIMLA adopts a causal inference framework, as we show that the SHAP value $\phi_t(f_c, x)$ approximates a well-defined causal quantity reflecting the influence of covariate X_t on outcome Y for sample x, under a reasonable set of assumptions (see Methods).

Step 3 ("Aggregation module"): Finally, the difference in local contribution scores for the two models is computed for each sample and aggregated across all samples in $D_{c=1}$ using a "root mean square" function (Figure 4.1E). Specifically, for each sample x in $D_{c=1}$, we calculate $\Delta_t(x) = \phi_t(f_{c=1}, x) - \phi_t(f_{c=0}, x)$, and then compute the aggregate $\Lambda_t = \sqrt{E_x \Delta_t(x)^2}$. This aggregate ($\Lambda_t$) is called the "CIMLA score" of covariate X_t, and represents the extent to which a causal association (if any) between X_t and outcome Y has changed between the two populations.

The CIMLA implementation has separate modules for the three steps outlined above, allowing for future developers to modify or improve each step independently.

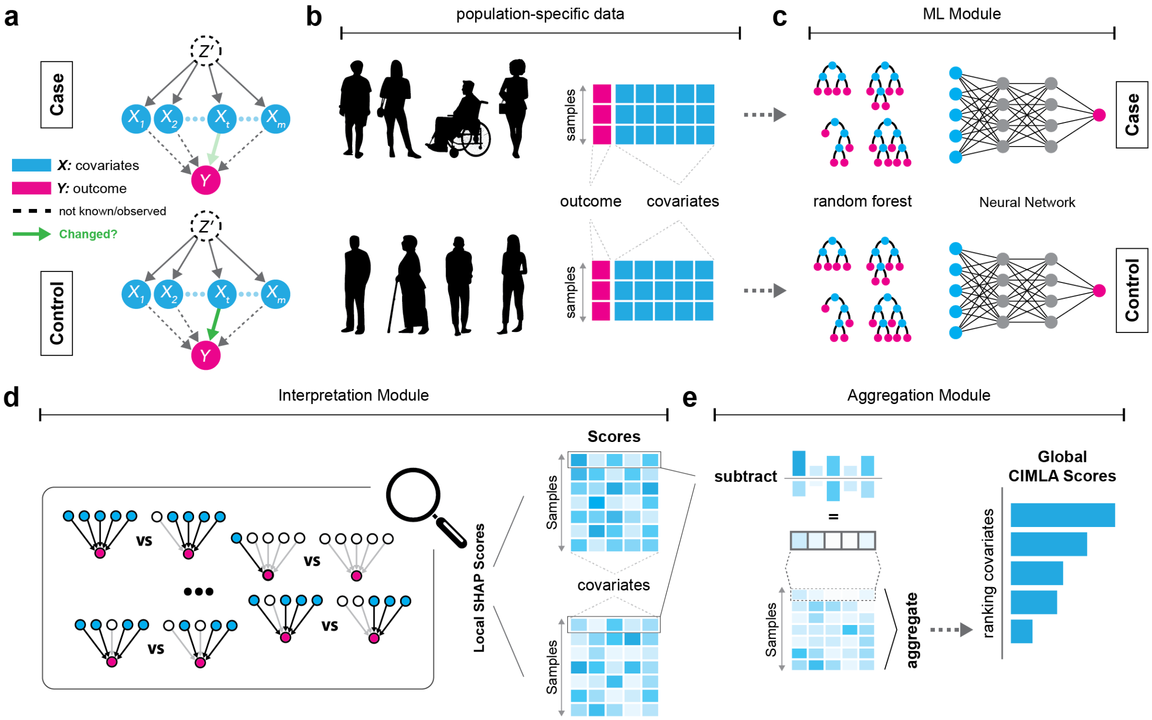

Figure 4.1: Schematic of CIMLA pipeline. (A) Given m observed covariates ($\{X_1, X_2, \ldots, X_m\}$) and an outcome of interest (Y) in two conditions, case and control, CIMLA's goal is to identify whether the causal association (if any) between a covariate of interest (e.g., X_i) and Y has changed between the two conditions. The causal associations between other covariates and Y are generally not known (dashed arrows), and an unobserved confounder (Z') may be involved. (B) Observational data for the outcome of interest (Y) and important covariates (X) are available for two populations that are different in terms of some condition (e.g., disease status). (C) In each population, the outcome variable is separately modeled as a (non-linear) function of covariates using random forests or neural networks. (D) CIMLA relies on SHAP to approximate a causal measure of association between each covariate and the outcome, in each population. Given a sample from the data and any trained ML model, SHAP searches over possible covariate coalitions to compute the contribution of each covariate to the output of the model locally around the provided sample. This "local SHAP score" is obtained for all provided samples, resulting in matrices of local SHAP scores. (E) The population-specific local SHAP scores of covariates, obtained in the previous step, are compared between the two populations and aggregated over samples into a global CIMLA score for each covariate, representing the strength of differential association between that covariate and the outcome variable. The global CIMLA scores are used to obtain a ranking over covariates by their differential association with outcome.

4.4.2 Benchmarking on simulated transcriptomics data

As there are no gold standard datasets for evaluating dGRN inference methods, we relied on synthetic data to benchmark CIMLA and compare it with existing methods. We used SERGIO [76] to simulate single-cell expression datasets representing two distinct conditions, arbitrarily termed "case" ($C = 1$) and "control" ($C = 0$), with similar but non-identical GRNs underlying the two datasets. SERGIO is a single-cell expression simulator that synthesizes transcriptomics profiles according to a provided GRN, modeling transcriptional regulation through stochastic differential equations similar to those employed in GeneNetWeaver [114, 116]. We used a "reference GRN" previously reported for yeast (see Methods) to construct the case and control GRNs, each containing a subset of the edges in the reference GRN such that the extent of difference between the two GRNs is controlled. The two GRNs were then used by SERGIO for generating case and control expression datasets respectively, with pre-defined profiles of master regulators ("MRs" – TFs that are not targeted by other regulators in the GRN). We repeated this process 15 times to obtain as many GRN pairs, each pair sharing $43 - 94\%$ of edges (Figure 4.2A), and corresponding pairs of case and control datasets, each dataset comprising a matrix of expression values with rows representing genes and columns representing cells. The two expression datasets can be provided to a dGRN inference method and since the differences between GRNs are known, it is possible to precisely evaluate its predictions.

For each of the 15 tests, we used CIMLA and other methods to infer dGRN edges. In the results presented in this section, we used RF as the underlying ML model and the TreeSHAP [168] method for interpretation. (We later show the performance of CIMLA with NN models.) For comparison, we tested six co-expression-based methods for dGRN inference (henceforth called "co-expression methods") that were found by Bhuva et al. [184] to be the leaders among all evaluated methods in recovering differential associations. These methods quantify pairwise associations between genes and test for statistical significance of the changes in association metrics between the two datasets in the test. Four of the selected methods rely on correlation metrics (z-score-Pearson, z-score-Spearman [200], MAGIC [201], and DICER [202]), one method uses entropy (entropy method [203]) and the other employs F-statistics (ECF method [204]) to measure pairwise associations. Each method, including CIMLA, is capable of reporting a sorted list of dGRN edges that can be evaluated using standard metrics of classifier performance given a ground truth list of dGRN edges.

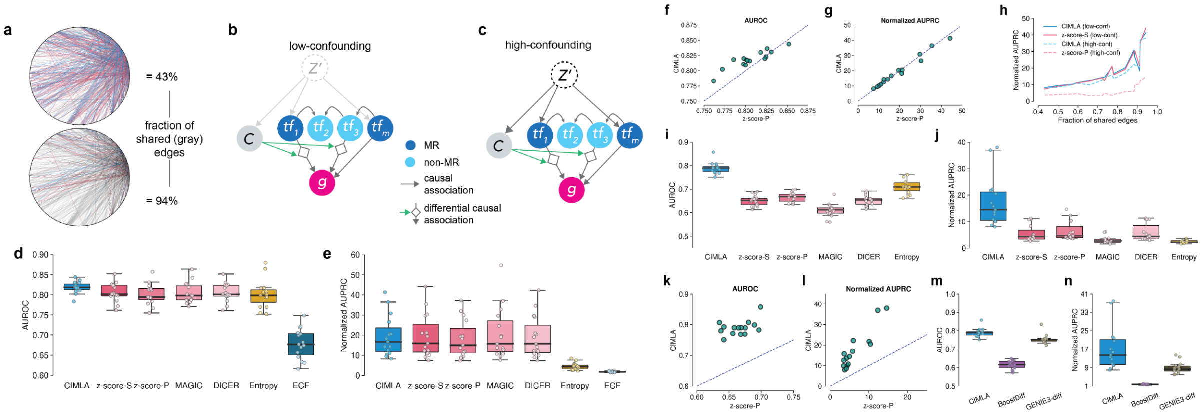

Figure 4.2: Benchmarking on clean simulated data. (A) Two of the 15 dGRNs used for simulations of gene expression data and benchmarking of GRN reconstruction in this study. The dGRNs shows are those with the least (43%) and most (94%) fraction of shared regulatory edges between conditions. Gray and colored edges represent shared and condition-specific edges, respectively. (B) Underlying causal diagram for the low-confounding simulation setting including master regulators (MR), non-MR TFs and target gene of interest. Node g represents the target gene and nodes $tf_{i\in\{1..m\}}$ represent transcription factors, some of which are master regulators (dark blue nodes). Node C represents disease status, which can impact the associations between a subset of TFs and the target gene (marked with diamonds hit by green arrows). We used MR profiles drawn from the same distribution for simulations of the two conditions, to rule out extraneous associations between C and g through backdoor paths. (C) Underlying causal diagram for the high-confounding simulation setting. We used different MR profiles (in 10 cell-types) for the simulations of the two conditions to mimic an unobserved confounder, Z', that controls both C and MR expressions. Here, in contrast to (B), there exist backdoor paths of association between C and g (which pass through Z'), thus creating a confounding effect for detection of direct causal associations between C and TF-target gene associations. (D) Area under the receiver operating characteristic curve (AUROC) for CIMLA and co-expression methods on low-confounding simulated data. (E) Area under the precision-recall curve (AUPRC), normalized by the expected AUPRC of a random classifier, for CIMLA and co-expression methods. In (D) and (E), each box plot shows performance measures for the 15 data sets. (F,G) Performance of CIMLA versus z-score-P on each of the low-confounding data sets, in terms of (F) AUROC, and (G) normalized AUPRC. (H) Performance of CIMLA and z-score method for varying levels of similarity (shared edges) between the case and control GRNs. (I,J) Performance of CIMLA and co-expression methods on high-confounding simulated data sets, in terms of (I) AUROC, and (J) normalized AUPRC. (K,L) Performance of CIMLA versus z-score-S on each of the high-confounding simulated data sets, in terms of (K) AUROC, and (L) normalized AUPRC. (M,N) Comparison of CIMLA with non-linear, multivariable modeling methods on the high-confounding simulated datasets, in terms of (M) AUROC, and (N) normalized AUPRC.

We performed our evaluations in two settings – "low-confounding" and "high-confounding" (Figures 4.2B,C). In the low-confounding setting, we used comparable Master Regulator (MR) expression profiles in generating transcriptomic data from the case and control GRNs; thus, the only difference between the groups was due to differences in their GRNs. On the other hand, for the high-confounding setting (Figure 4.2C), we set the MR expression profiles to be different in the two groups, mimicking the presence of an unobserved confounder that creates backdoor paths of associations (via Z' in Figure 4.2C) between population-specific condition (C) and target genes (g). Figure 4.2D,E compare the performance of the evaluated methods in terms of AUROC and normalized AUPRC (AUPRC divided by expected AUPRC of random, class size-aware prediction) in the low-confounding setting. We find CIMLA to be competitive with the other methods. (The ECF method performed poorly and was excluded from further analysis.) This is consistent with our expectation that in the absence of confounders, causal inference can be made from associative quantities such as correlation metrics. The z-score-Spearman (z-score-S) method, which has the second-best median normalized AUPRC after CIMLA, is highly competitive with CIMLA on each of the 15 tests (Figure 4.2F,G). Also, the performance of both methods generally improves as the extent of difference between case and control GRNs decreases (Figure 4.2H).

Next, we repeated the above evaluations in the "high-confounding" setting, where the inter-condition difference in GRNs is confounded by differences in the MR profiles that serve as "inputs" to the GRN in generating expression data (see Methods). In this case, as is evident from Figure 4.2I,J, the performance (AUROC and normalized AUPRC) of all competing methods dropped compared to the low-confounding setting (Figure 4.2D,E), leaving a large gap between these methods and CIMLA. Even the second-best method (based on normalized AUPRC) – z-score-Pearson (z-score-P) – is greatly outperformed by CIMLA on every individual dataset tested (Figure 4.2K,L). Notably, CIMLA's performance remains robust to the confounder effect (Figure 4.2H). This robustness to confounders was one of the main motivations behind the causal framework of CIMLA. As the high-confounding setting represents a more realistic scenario, our benchmarking results indicate the importance of causal approaches for dGRN inference.

While co-expression methods explore linear associations, CIMLA can capture non-linear relationships due to its underlying ML models. This motivated us to next compare CIMLA with methods – GENIE3 [11] and BoostDiff [206] – that consider the non-linearities of regulatory relationships. BoostDiff [206] employs differential boosted trees using a novel criterion for growing trees based on the difference of the predictive power of regulators in two conditions. GENIE3 [11], a leading GRN (not dGRN) inference tool, uses random forests to model each gene's expression in terms of TFs' expression profiles, and uses importance of TFs in the trained model, assessed via an information theoretic tree-based feature importance metric, to score TF-gene pairs. We adapted GENIE3 for dGRN inference by applying it separately on case and control data and comparing the scores of corresponding edges in the two inferred GRNs to rank dGRN edges ("GENIE3-diff"). As shown in Figures 4.2M,N, CIMLA convincingly outperforms both BoostDiff and GENIE3-diff in terms of both AUROC and normalized AUPRC.

In summary, benchmarking on synthetic data sets demonstrates the advantage of CIMLA over pairwise association as well as multivariable regression-based, linear as well as non-linear modeling approaches to dGRN inference.

4.4.3 CIMLA prioritizes causation among correlations

Our results so far suggest that confounders can significantly hinder current methods for dGRN inference and CIMLA is relatively robust to their effect. Our next analyses probed deeper into the source of CIMLA's empirical advantage. First, we defined "delta-correlation" for a TF-gene pair "tf, g", as the absolute difference of their pairwise correlations in case and control groups; this is a simple measure of the "correlation signal" that points to differential regulation. Figure 4.3A compares these delta-correlation values for all TF-gene pairs that belong to the true dGRNs ("differential pairs") and those that do not ("non-differential pairs"). We noted that in the low-confounding setting the distribution of delta-correlation values is clearly different between the two groups of TF-gene pairs; this enables easy detection of differential pairs. In the high-confounding setting, on the other hand, the two distributions are much more similar. Notably, non-differential TF-gene pairs show higher delta-correlations in the high-confounding setting compared to the low-confounding one. This implies that confounders introduce many non-causal group-specific associations that can cause correlation-based methods to err.

Next, we examined (Figures 4.3B,C) the scores assigned by CIMLA and a correlation-based method, z-score-S, (both converted to percentile) to the differential TF-gene pairs, along with the delta-correlations of those pairs. We noticed, in the low-confounding setting, that z-score-S is largely driven by delta-correlation, as expected (Figure 4.3B). On the other hand, CIMLA scores are less tightly determined by delta-correlation, and we noted high scores being assigned to many of the differential pairs with small delta-correlations. In high-

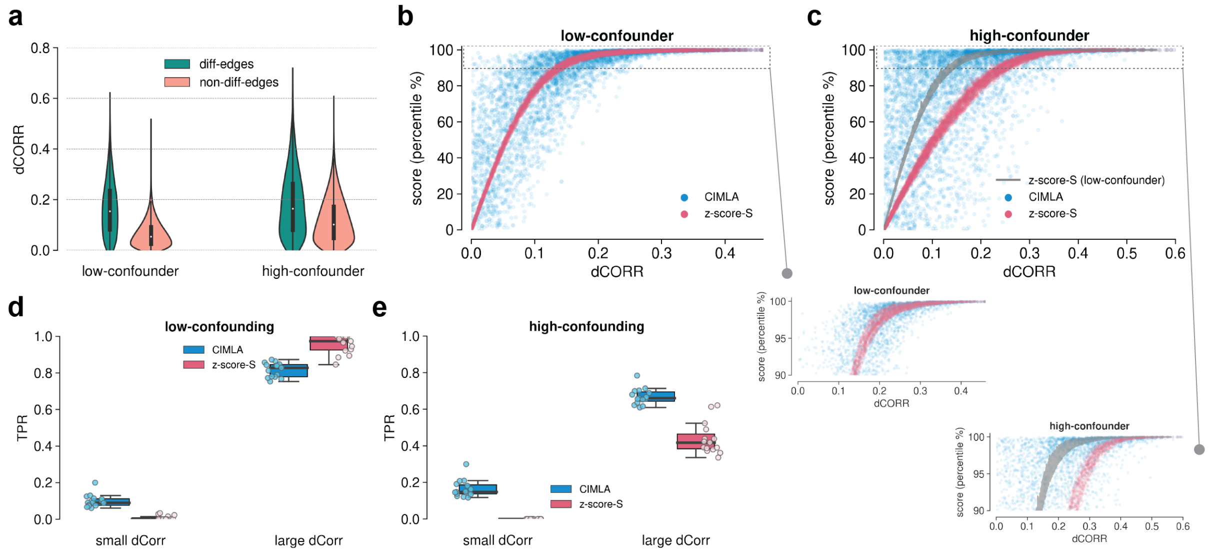

Figure 4.3: (A) Distributions of delta-correlations (dCORR, difference of TF-gene correlation coefficient between case and control populations), for true differential and non-differential edges, across all 15 simulated data sets, in low- and high-confounding settings. The non-differential edges consist of the TF-gene edges that are present in (shared by) both case and control GRNs, as well as those that are absent in both GRNs. (B,C) Relationships between delta-correlation (dCORR) and scores (converted to percentile) assigned to the true differential edges by CIMLA (blue) and z-score-S (burgundy) methods in (B) low-confounding, and (C) high-confounding settings. Panel (C) additionally shows z-score-S assigned scores from the low-confounding setting for comparison. Insets show zoomed-in views of the top 10 percentile scores suggesting that in high-confounding setting, CIMLA assigns high scores to differential pairs with even smaller delta-correlations compared to low-confounding setting, while z-score-S demands relatively larger delta-correlations to assign high scores. (D,E) True Positive Rate (TPR) for the top 5% predictions of CIMLA and z-score-S for the task of discriminating differential edges from non-differential edges. Evaluations are done separately for all TF-gene pairs with small delta-correlations (dCorr≤0.16) and those with large delta-correlations (dCorr>0.16), shown as two groups in (D) low-confounding, and (E) high-confounding settings. The 0.16 cutoff was used based on the median of delta-correlations over the union of true differential edges in all low- and high-confounding simulated datasets.

confounding settings, CIMLA is even less dictated by the delta correlation signal (compared to the low-confounding setting), resulting in a larger gap between the two methods in their ability to identify differential pairs with small delta-correlations (Figure 4.3C). This enlarged gap is also apparent when we examine the True Positive Rates (TPR) of either method in the high-confounding setting (Figures 4.3D,E), and is likely the reason behind the greater precision and recall exhibited by CIMLA compared to z-score-S in this setting (Appendix C, Supplementary Figure C.3A,B). Interestingly, when we compare the false positive rate (FPR) of the two methods (Appendix C, Supplementary Figure C.3C,D), z-score-S shows significantly larger FPRs compared to CIMLA for detecting the large delta-correlation pairs, suggesting that even in low-confounding settings, co-expression methods may be prone to reporting non-causal correlations (Appendix C, Supplementary Figure C.3C).

4.4.4 Evaluations on noisy simulated data

The evaluations reported above were performed with “clean” simulated datasets, where SERGIO does not introduce any technical noise to the generated data. However, real single-cell RNA-seq data suffer from significant technical noise, especially “dropout”, which incorrectly introduces significant numbers of zero values to the expression matrix. To explore the impact of technical noise we repeated parts of the above evaluations on synthetic single-cell expression datasets with dropout. Specifically, starting with one of the 15 tests from above (with fraction of shared GRN edges = 94%), in the high-confounding setting, we added increasing levels of dropout (10%, 20%, . . . , 70%), with 5 “replicates” per dropout level. Following recommendations from previous work [76], we imputed missing values (zeros) in the expression matrix using MAGIC [92] (with t=2) prior to applying dGRN methods. Moreover, in addition to RF, we also tested a fully connected neural network (NN) as the underlying ML model and DeepSHAP [180] for SHAP score calculations in CIMLA.

We compared the performance of CIMLA with the co-expression methods as well as Boost-Diff and GENIE3-diff at varying dropout levels. As expected, dGRN inference deteriorates as the level of noise increases (Appendix C, Supplementary Figure C.4A,B). In terms of both AUROC and AUPRC, the two versions of CIMLA – CIMLA-RF and CIMLA-NN – outperform co-expression methods at almost all levels of dropout; however, GENIE3-diff shows competitive performances in terms of AUPRC. These results were obtained for the task of predicting all differential regulators of all genes, i.e., the entire dGRN. In an alternative evaluation, we calculated the AUROC and AUPRC for each target gene separately and examined the per-gene performance metrics over all genes. (Genes with no differential regulators in the ground truth were excluded.) Figures 4.4A,B compare all methods by the median per-gene AUROC and (normalized) AUPRC respectively, revealing an even clearer advantage for the two CIMLA versions (also see Appendix C, Supplementary Figure C.4C-E). We noted that at the highest level of dropout CIMLA-NN achieves better performances compared to CIMLA-RF which suggests greater tolerance of CIMLA-NN to technical noise in the data.

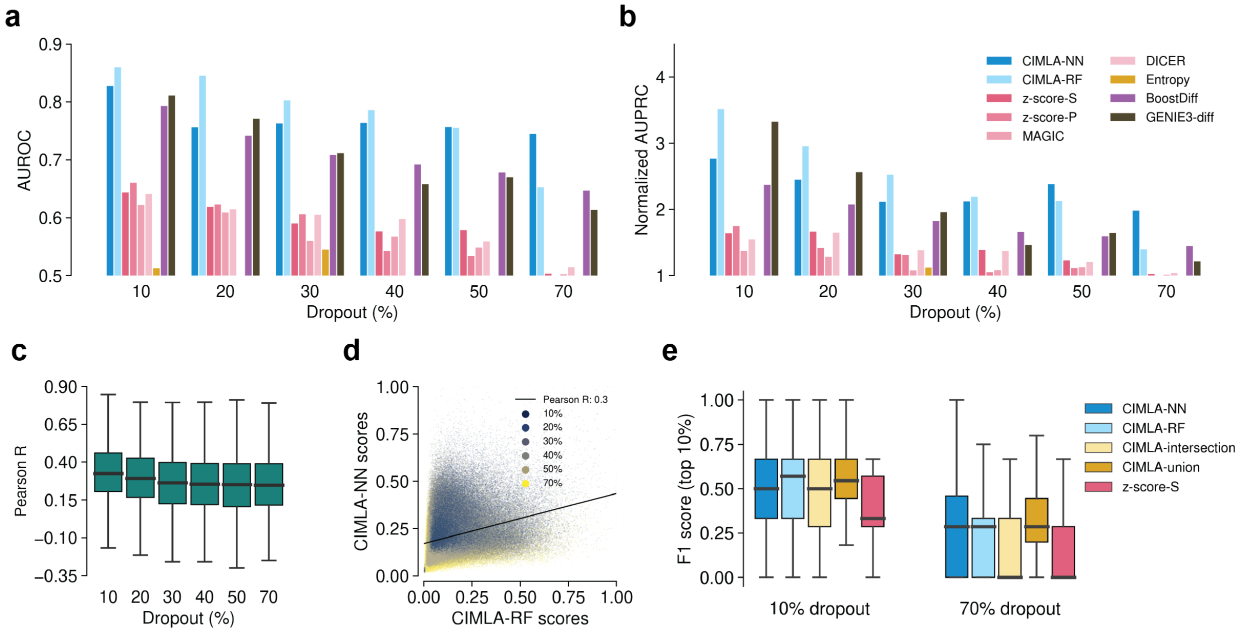

Figure 4.4: Benchmarking on noisy simulated data. (A,B) Performance of CIMLA and other methods in terms of (A) AUROC, and (B) Normalized AUPRC, at varying dropout levels. Performance is assessed for each gene separately and each bar represents the median over all differentially regulated genes (every gene with at least one true differential regulation was included) and five simulated replicates. (C) Pearson's correlation between CIMLA-RF and CIMLA-NN scores of all candidate transcription factors of each target gene, at different dropout levels. At each dropout level, distribution of correlation values is shown over all target genes and simulated replicates. (D) Relationship between CIMLA-NN and CIMLA-RF scores across all TF-gene pairs and all (noisy) simulated data sets. Scores in each simulated dataset were max-normalized (with respect to the maximum score in that dataset) for both methods separately. (G) Performance (F1 score) of dGRN inference derived from the intersection or union of the top 10% scoring TF-gene pairs predicted by CIMLA-RF and CIMLA-NN is compared with the performance of the individual methods and z-score-S method at their top 10% scoring TF-gene pairs.

We observed that the CIMLA-RF and CIMLA-NN scores of candidate TFs for a gene show relatively small correlation with each other for many of the target genes (Figure 4.4C, e.g., median Pearson's correlation of 0.25 at 70% dropout) and also at a more global level (Figure 4.4D , Pearson's correlation of 0.30). This suggests some complementarity between the two methods, which may allow a combination of the two methods to yield superior performance compared to either alone. We thus defined two related methods – CIMLA-intersection and CIMLA-union – whose dGRN output is the intersection and union respectively of the top 10% scoring TF-gene pairs of CIMLA-RF and CIMLA-NN. We compared their F1 scores with those of the individual methods and z-score-S (also restricted to top 10% predictions) and found that CIMLA-union outperforms the predictions made by other methods, especially at the higher dropout level that is a more realistic scenario for single cell data (Figure 4.4E).

4.4.5 Differential GRN of Alzheimer's disease: a case study

We used CIMLA for the inference of regulatory changes underlying Alzheimer's disease (AD), analyzing a previously published single-nucleus RNA-seq data obtained from the prefrontal cortex of individuals with or without AD pathology [196]. We used the expression profiles of 44 individuals (see Methods), spanning 16,004 genes in 30,853 cells from 22 individuals in the AD group and 34,852 cells from 22 individuals in the control group. Missing values in the expression matrices of AD and control groups were separately imputed using MAGIC (with $t = 2$) [92]. GRN reconstruction was focused on a set of 2652 target genes associated with AD according to the DisGeNET database [209] and an unbiased list of 1289 transcription factors annotated in the AnimalTFDB database [210]. We employed both CIMLA-RF and CIMLA-NN (separately) to identify differential TF-gene pairs between the two groups.

The first step of CIMLA is to train an ML model (Random Forest or Neural Network) to predict each target gene's expression using all TFs as covariates. We assessed the accuracy of these ML models with train-test splits (Figures 4.5A,B) and noted 1803 (resp. 2079) of the 2652 genes to be reliably modeled by RF (resp. NN) in both AD and control groups, as suggested by their R-squared, $R^2 \geq 0.5$, on both train and test data. The remaining genes and low variance genes were not analyzed further. We assessed the statistical significance of predicted differential regulators, using a background distribution of CIMLA scores obtained on randomized data where group labels (AD/control) of cells had been shuffled (see Methods), and selecting the highest CIMLA score seen for a gene as the significance threshold. Figure 4.5C shows the distribution of the number of differential regulators of the reliably modeled genes, as predicted by CIMLA-RF and CIMLA-NN, showing that the former is more conservative in its predictions. We limited the resulting dGRNs to target genes that are differentially expressed between AD and control (see Methods) and identified the hub TFs and highly targeted genes (TFs and genes respectively included in greatest number of differential pairs) that may play important roles in AD-related dysregulation (Figures 4.5D,E and Supplementary Figures C.5A,B in Appendix C). A survey of the literature revealed evidence in favor of eight of the top 10 hub TFs and nine of the top 10 highly targeted genes in CIMLA-NN's dGRN to be associated with AD and other neurodegenerative diseases (Appendix C, Supplementary Tables C.1,2).

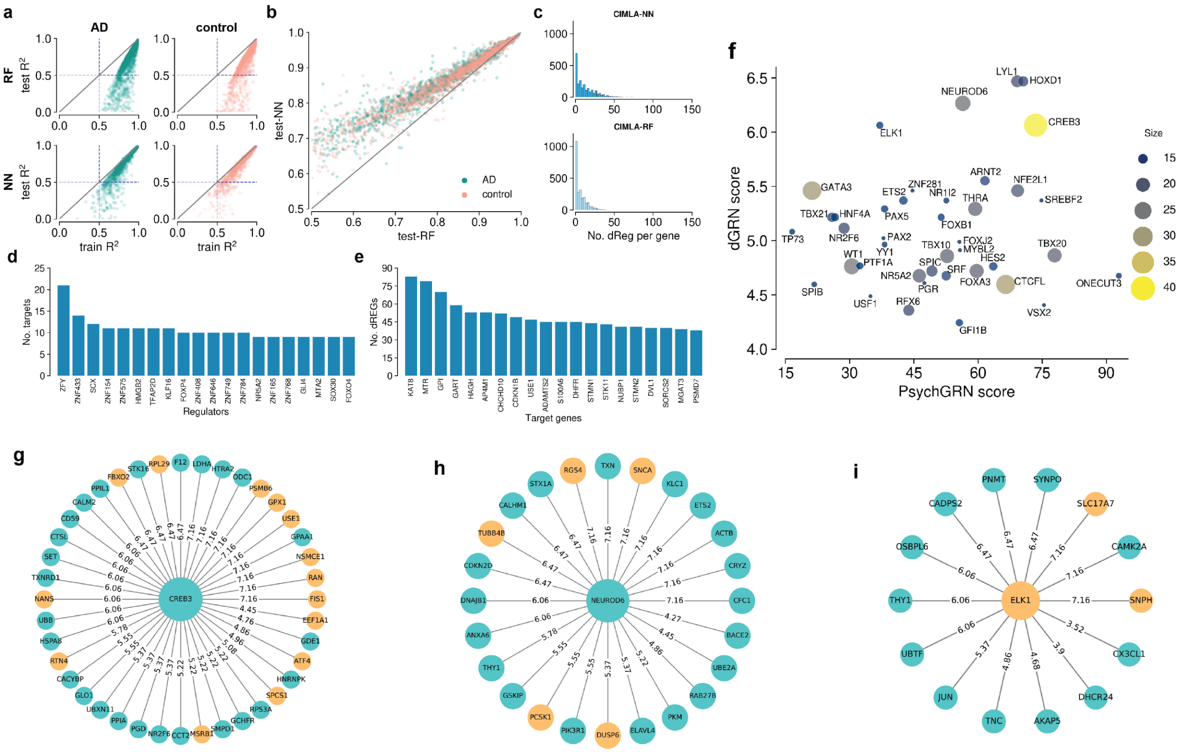

Figure 4.5: Case study of Alzheimer's disease. (A) Training and test data performance (R-squared, R^2) of the RF and NN models trained in step 1 of CIMLA for each of the target genes, in the AD and control groups. (Genes with negative test R-squared are not shown. Note that R-squared for nonlinear regression can have negative values indicating significant overfitting.) (B) Comparing the test performance (R-squared, R^2) of NN with that of RF for genes that are reliably modeled ($r^2 \geq 0.5$ on both train and test data in both AD and control groups) by both models. NN yields more accurate predictions than RF in general, suggesting that it is less prone to overfitting. (C) Distributions of the number of differential regulatory edges found for each target gene (after thresholding by maximum background score) by CIMLA-NN and CIMLA-RF. (D) Top 20 "hub" TFs (those targeting greatest number of DEGs) found by CIMLA-NN. (E) Top 20 differentially regulated DEGs, i.e., those with greatest number of differential regulators (dREGs) found by CIMLA-NN. (F) Summary of the top 40 largest "regulons" (targets of individual TFs) in the dGRN of AD obtained by intersecting the predictions of CIMLA-union with PsychGRN. For each regulon, its dGRN score (average of dGRN scores of the TF-gene pairs in the regulon), PsychGRN score (average of PsychGRN percentile scores of the TF-gene pairs in the regulon) and its size (number of target genes in the regulon) are shown. (G-I) Predicted regulons for (G) CREB3, (H) NEUROD6, and (I) ELK1 transcription factors. Predicted target genes that are differentially expressed between AD and control groups (i.e. DEGs) are shown by orange-colored nodes. Edges are labeled by their CIMLA dGRN score.

4.4.6 CIMLA reveals CREB3 and NEUROD6 as potential key regulators of AD

The TF-gene pairs identified by CIMLA above are likely to include many indirect regulatory relationships, as mechanistic information such as TF-DNA binding was not included in the reconstruction. Thus, we integrated our results with a published GRN for the human brain (from PsychENCODE project [197]), in order to enrich the dGRN for direct regulatory relationships. This GRN (henceforth called "PsychGRN") is based on multi-omics (DNaseI hypersensitivity, Hi-C, TF motifs, RNA-seq) data on multiple psychiatric disorders. We focused on the 2055 TF-gene edges shared between PsychGRN and the dGRN derived using the CIMLA-union method, involving 401 TFs and 1114 genes which we refer to collectively as the "high-confidence dGRN".

For each TF, we extracted its "regulon" (all predicted targets) in the high-confidence dGRN, and assigned it a dGRN and a PsychGRN score based on the average dGRN scores and average PsychGRN weights (see Methods) of the included TF-gene relationships, respectively. Figure 4.5F shows the top 40 largest regulons in the high-confidence dGRN. We find the TF CREB3 to have the largest regulon, targeting 40 genes (Figure 4.5G), with a dGRN score that is among the top five. CREB3 (cAMP response element binding protein 3) is involved in regulation of Golgi homeostasis and significantly contributes to Central Nervous System (CNS) function and development [211, 212, 213]. It is involved in unfolded protein response to endoplasmic reticulum (ER) stress [214], a response activated in AD [215]. CREB3 is also known to regulate GLUT3, a neuronal glucose transporter that is related to AD [216].

NEUROD6 (neuronal differentiation 6) has the third top regulon in terms of dGRN score (Figure 4.5F) targeting 24 genes (Figure 4.5H). It is involved in nervous system development and differentiation [217]. Its downregulation is a biomarker of AD [218] and a significant predictor of cognitive decline [219]. SNPs in its locus are associated with AD in a sex-specific manner [220, 221]. ELK1, a member of the TCF subfamily of ETS-domain transcription factors, is another TF whose regulon is in the top five in terms of dGRN score (Figure 4.5F), comprising 14 genes (Figure 4.5I). ELK1 is implicated in neuronal differentiation [222] and has been associated with neuronal death and Alzheimer's disease [223]. Elk1 inhibits presenilin 1 (PS1), which is important for making variants of beta amyloid, a trigger for Alzheimer's Disease [224]. Signaling involving Elk1, Ras1 and CentA1 has been reported to connect beta amyloids and synaptic dysfunction, a hallmark of AD [225].

It is also interesting to examine TFs whose predicted regulons have a high dGRN score but a relatively low PsychGRN score, possibly pointing to regulators for which mechanistic evidence is under-represented in PsychENCODE. One such TF is GATA3 (Figure 4.5F), a pioneer TF [226] involved in signaling pathways associated with neuronal development [227] and control of immune T-cell fate [226]. Donepezil, a drug used for AD, modulates immune response in part by inducing GATA3 [228]. GWAS SNPs associated with late onset AD show allele-specific binding of GATA3 [229]. Moreover, a GWAS for resilience to cognitive consequences of AD revealed a female-specific locus that interacts with GATA3 and suggested GATA3 as a candidate gene [230].

In summary, differential regulation detected by CIMLA, combined with evidence from the PsychENCODE project, points to regulatory programs that have been reported to play a role in AD and also reveals novel potential regulatory pathologies of AD.

4.5 DISCUSSION

Inference of the differential associations in comparisons of two or multiple groups is of paramount importance in systems biology, with potential applications ranging from tissue-, sex- or population-specific genetic association analysis [231, 232, 233] to contrasting the regulatory programs of different populations [184, 206]. In recent years, association analysis in genetics and genomics has increasingly relied on complex statistical and ML models [163, 164, 165, 166, 167]. However, inference of causal (and differential causal) associations from observational data is fundamentally hindered by confounding variables that are introduced by limitations and biases in data collection or are intrinsic to the problem at hand. This motivated us to approach the inference of differential associations from a causal perspective and develop CIMLA, employing non-linear, multivariable models and model interpretation based on SHAP values [168, 180] to approximate a causal estimand of association and changes thereof.

We demonstrated the application of CIMLA for differential gene regulatory network (dGRN) inference from gene expression data. On realistic synthetic data sets CIMLA outperforms existing methods that are based on linear as well as non-linear models, especially in simulations including strong confounders, which are a realistic reflection of biological systems. Our results suggest that CIMLA, in contrast to co-expression methods, can even identify differential regulatory relationships that show relatively small differences in TF-gene correlation between the two populations. Finally, we used CIMLA to infer the differential regulatory program underly Alzheimer's Disease (AD). The resulting dGRN points to CREB3 and NEUROD6 as two important regulators of AD which is in concordance with previous reports for the important role of these TFs in AD [212, 218], and also suggests novel potential regulators such as LYL1 and HOXD1 (Figure 4.5F).

CIMLA employs ML models to impute counterfactuals, i.e., to predict the hypothetical outcome if a sample had belonged to the alternative condition or intervention of interest. This relies on the strong assumption of transferability of the ML models to data distributions that have not been seen during training. While this transferability assumption is consistent with the positivity assumption [195], which is crucial for causal inference, the tradeoff between positivity and conditional ignorability assumptions [234] raises concerns about the transferability of ML models in practice. This defines a future direction for improving CIMLA by employing ML models that can simultaneously learn from the two populations via training jointly on their observational data. (See Shalit et al. [235] and Baur et al. [236].) These advancements in CIMLA may better separate the regulators that are common between conditions from differential associations and also facilitate the extension of this tool to dGRN inference from more than two conditions.

Although we used data from two conditions to train ML models, throughout this study we have used samples of only one of the conditions to quantify local differential scores (equation (4.5) in Methods) before aggregating them into a global score (equation (4.6) in Methods). However, a rigorous incorporation of samples from both conditions in the local interpretation step can further enhance the performance of dGRN inference and can be a potential direction for improving CIMLA in future studies.

A broader implication of this work lies in the causal interpretation we provide for SHAP values [168, 180]. Causal notations were first introduced to SHAP by Janzing et al. [182] and have been later adopted by other studies [168, 237]. However, a rigorous mathematical interpretation of the SHAP feature attribution score as the solution to a causal inference problem has been lacking. We bridged this gap by formulating feature attribution as a causal problem and positing a precise set of assumptions under which it simplifies to a statistical quantity that can be approximated from observational data by SHAP. Interestingly, our causal formulation, under three clearly stated assumptions (see Methods), is resolved into a format that is in concordance with the viewpoint of Janzing et al. [182]. But we note that using machine learning and feature attribution methods to draw conclusions about causal relationships needs extreme care. Any of the three stated assumptions can be impacted by problem characteristics and observational data. For example, significant multi-collinearity in data can negatively impact the accuracy of ML models (assumption 1), though this impact can be relieved by regularization techniques during the training of ML models. Data characteristics, including multi-collinearity, might also adversely impact the distributions assumed over causal structures and reference values (assumptions 2 and 3). For example, although the particular distribution assumed over different causal structures guarantees the "local accuracy" property of SHAP [180], recently Kwon and Zou [238] showed that it can lead to suboptimal feature attributions. We hope that the causal interpretation we provided for SHAP and its three underlying assumptions help future studies to further guide feature attribution models toward capturing genuine causal effects.

CHAPTER 5: CONCLUSIONS

Many intra-cellular relationships are inherently non-linear, motivating corresponding non-linearities in the mathematical functions used for modeling them. These non-linearities for example include non-linear regulatory effects of TFs' on their target genes' transcription, and non-linearities in interaction energies between proteins and cis-regulatory elements resulting in significant variabilities in TF binding specificities. Computational models that attempt to learn various aspects of transcriptional regulation form data have been provably benefited from including non-linearities in their underlying models [13, 16, 23, 49, 239]. But as the underlying model deviates from linearity, its interpretation becomes more ambiguous. This imposes an immediate challenge on computational studies of transcriptional regulation that on the one hand attempt to capture higher-order relationships via non-linear models, and on the other hand rely on some degree of interpretability of the learnt models. In this work, we adopted existing ideas in interpretable machine learning to study transcriptional regulation through non-linear models whose proper interpretation can provide mechanistic and causal insights in regulatory genomics.

In chapter 2, we first studied a group of existing quantitative models of gene expression, namely thermodynamics models, through an extensive benchmark on the various settings of such models. These models provide some level of interpretability for transcriptional regulatory mechanisms while employing a non-linear underlying model that is learned from omics data. Interpretability in these models is achieved by relying on hardcoded mathematical formalisms that are biophysics inspired representation of regulatory mechanisms. Next, we expanded on this mechanistic perspective by developing an interpretable neural network model of gene expression – CoNSEPT [12], whose greater flexibility compared to thermodynamics models enables the learning of regulatory mechanisms in a free-form manner. We focused on learning distance-dependent regulatory mechanisms by employing a simple attention mechanism using convolutional kernels that explicitly enforces CoNSEPT to learn meaningful higher-order representations for pairs of TF bindings sites (TFBS) on cis-regulatory elements. Using a previously published dataset for Drosophila melanogaster, we showed that the interpretations of properly trained CoNSEPT models reveal meaningful non-linear patterns of interactions between TFBSs, activating and repressing regulatory effects, and a few characteristics of regulatory functions such as effective ranges of pairwise interactions. While CoNSEPT learns such biological insights solely from data, earlier thermodynamics models commonly offer simpler insights that are confined by user-defined regulatory relationships and rules.

In chapter 3, we proposed a model of transcriptional regulatory networks for simulations of collective expressions of genes comprising a regulatory system. To achieve this, first we leveraged an existing model for stochastic gene expression, which explicitly encodes the causal regulatory effects between genes to develop a probabilistic model for gene expressions under regulatory networks. Next, using statistical assumption of ergodicity we extended this framework to single-cell level. Finally, through biophysical assumptions and numerical methods we adopted this framework for scalable simulations of gene expressions under GRNs at the cell resolution. Due to the lack of gold-standard annotations for true regulatory associations in most of the mammalian systems, such forward modeling of transcriptional regulation is particularly useful for benchmarking the GRN inference algorithms that attempt to learn causal regulatory interactions from gene expression datasets. This publicly available simulator-SERGIO [76], due to its underlying biophysical model of transcriptional regulation, also provides a useful framework for benchmarking other single-cell analysis tools such as cell clustering, trajectory, and velocity inference algorithms. The interpretable transcriptional model of SERGIO extends its applications beyond benchmarking and make it useful for *in silico* analysis of regulatory dynamics as we showed with our analysis of T-cell differentiation. Moreover, in practice, SERGIO is useful for benchmarking general causal inference algorithms (from observational data) in various domains [240, 241].

In chapter 4, we approached the inference of regulatory associations from a causal perspective. We pursued this direction by defining an estimand that measures the causal associations between genes and proposed a procedure for estimating it through non-linear machine learning models and proper interpretations of the learned models. We showed that under clearly stated assumptions, this procedure immediately connects to feature attribution in machine learning using a specific class of models, SHAP [168, 180]. Therefore, by relying on SHAP as a tool to approximate our defined causal attributions, we extended this procedure toward the inference of differential regulatory associations between two conditions. Consequently, we developed CIMLA that approximates causal differential GRNs (dGRN) using gene expression data in two different conditions of case and control. CIMLA relies on non-linear machine learning models such as neural networks and random forests to model genes' expression as functions of regulators' expression profiles and SHAP [168, 180] to approximate differential association by a proper interpretation of the trained models. We showed that CIMLA is more robust to the confounding effects compared to simpler models that rely on univariate co-expression measures for dGRN inference. Additionally, we used CIMLA to study the differential regulatory program of Alzheimer's disease (AD) and identified transcription factors with potential roles in AD. An additional implication of this work is the causal intuition it provides for SHAP values [168, 180] that supports one of the previous interpretations of SHAP [182].

In summary, this thesis provides a data-driven study of transcriptional regulations with an emphasis on model interpretability. We showed that interpretability can provide mechanistic insights about cellular interactions and can improve the inference of causal regulatory associations.

APPENDIX A: SUPPLEMENTARY MATERIALS FOR CHAPTER 2

A.1 SUPPLEMENTARY NOTES

A.1.1 Analysis of binding affinities and overlapping binding sites in CoNSEPT

We examined the sensitivity of a trained CoNSEPT model to TFs' binding affinities by using it to predict the output of an enhancer containing a single binding site for the TF DL that was learnt as an activator. We repeated this with varying strength (affinity) of the singleton binding site, using the site's likelihood score per the TF's known PWM as a measure of site strength. We tested 35 binding sites with site likelihoods ranging from weak (close to zero) to strong (optimal site). Moreover, in order to rule out any bias arising from the position of the singleton site along the enhancer, for each tested binding site we generated 19 enhancers with different placements of the site and averaged the predictions over the 19 enhancers. In all tests, we used a fixed concentration for the TF and we evaluated the trained "CoNSEPT 1" model discussed in Figure 2.8F,G. As expected, the predicted expression driven by one DL site increases as the likelihood of the site increases (Supplementary Figure A3A). Specifically, we observed a non-linear saturating relationship between expression and site's likelihood score. The precise relationship, e.g., point of saturation, will depend on the TF concentration also, which is not shown.

The same trend is observed for the TF TWI, which was also correctly learnt as an activator. The expression increases non-linearly with site strength for sites with likelihood ratios in the range 0.07 to 1 (Supplementary Figure A3B). However, we see an anomaly in the predictions for enhancers with extremely weak TWI sites (likelihood ratio < 0.07). One explanation for this anomaly is that the training enhancers have at most two sites for TWI, one with a likelihood ratio of 1 (i.e. an optimal TWI site) and one with a likelihood ratio of 0.23. This may have prevented CoNSEPT from learning the proper use of binding affinities for extremely weak TWI sites.

Next, we performed a similar analysis for the TF SNA. Since SNA is (correctly) learnt to act as a repressor, we expect no predicted expression from enhancers with only one SNA binding site. Therefore, we placed a SNA binding site at a 20 bp distance from an optimal DL (activator) site and repeated this for 29 different SNA sites whose likelihood scores range from weak to strong according to the SNA PWM. Similar to the above analysis, for each

SNA site we generated 18 synthetic enhancers that differ in their placement of DL-SNA pair along the sequence and averaged the predicted expression over the 18 enhancers. In all tests we used fixed concentrations for DL and SNA. As expected, the predicted expression decreases as the strength of the SNA site (likelihood ratio to the optimal site) increases (Supplementary Figure A3C). Also, similar to the above analysis, we see that CoNSEPT learnt a non-linear saturating relationship between expression and the strength of SNA site where sites with a likelihood ratio of 0.4 or stronger have the same repressive power (this point of saturation will depend on the TF concentrations used).

Finally, to interpret the predictions of CoNSEPT regarding overlapping binding sites, we assessed the expression predicted by a trained CoNSEPT model (CoNSEPT 1 model shown in Figure 2.8F,G, and used in analyses above) for an overlapping pair of TWI and SNA sites and compared it with the expression driven by enhancers with the same TWI and SNA sites at various inter-site spacings. To guarantee an overlap we used and optimal TWI site (on the negative strand) and a sub-optimal SNA site (on the positive strand). Also, similar to the above analysis, for each inter-site spacing as well as for the overlapping site we generated multiple synthetic enhancers with different placements of the sites pair in the enhancer and averaged the expression over these enhancers. Supplementary Figure A3D illustrates the predicted expression where the left most point; i.e., "distance = -1", corresponds to the enhancer with the overlapping sites. Predicted expression at distances > 0 are essentially analogous to Figure 2.8G (third column) in the manuscript except that here we used a sub-optimal SNA site (which explains the overall higher predicted expression here). We noted that the expression driven by the overlapping site is no different from the expression driven by a single TWI site (red dashed line). This suggests that SNA site that overlaps with a TWI site is not able to repress the activation at all, possibly because occupancy is dominated by TWI at the tested settings of TF concentrations and site strengths. However, this finding might not be necessary reliable because the training enhancers harbor at most a single overlapping site pair (for TWI and SNA). Having a richer dataset containing many enhancers with multiple overlapping sites can strengthen the accuracy of our findings. Therefore, by no means we intend to imply here that our finding is "the correct" mechanism that underlies the overlapping sites, especially that a significant amount of research has already aimed at addressing this question and suggested various underlying mechanisms for the effects of overlapping sites.

A.1.2 Enhancer length constraints and data augmentation

CoNSEPT model requires each enhancer to be of the same length. The common length of all enhancers in the Sayal et al. training set is 332 bp [26]. Enhancers in the Sayal et al. test data set however have varying lengths and some are longer than 332 bp [26]. We excluded two such enhancers significantly longer than 332 bp (855 bp and 893 bp long). After this pruning, the longest enhancer in the remaining of test data set has a length of 635 bp which is nearly two times longer than training set enhancers. In order to make all the enhancers have the same length, we expanded all of the enhancers shorter than 635 bp by adding "dummy" bases at both ends. During one-hot encoding of the enhancer sequence, this dummy base is encoded into a vector of zeros, $[0, 0, 0, 0]$, to guarantee the detection of no binding site in these regions. Moreover, in order to decrease any bias toward the placement of these dummy bases during training, for each enhancer we repeat expansion for ∼10 times each with a different random number of dummy bases at the two ends, while keeping the total length at 635 bp. Therefore, after expansion we obtain an augmented training data set with a total size of up to 7107 (38 enhancers in 17 trans contexts with up to 11 different expansions).

A.2 SUPPLEMENTARY FIGURES

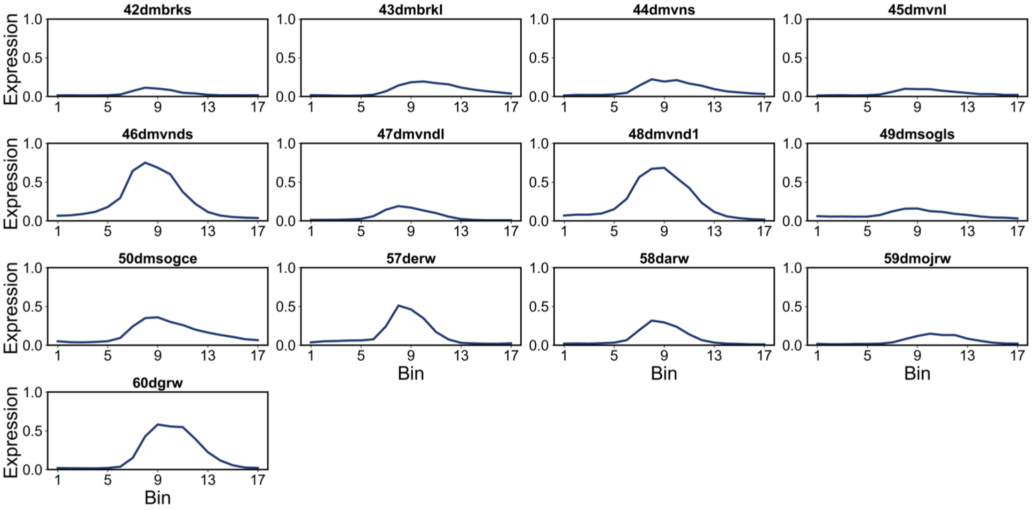

Figure A.1: Expression driven by 13 test enhancers along 17 V-D bins, as per data from [26].

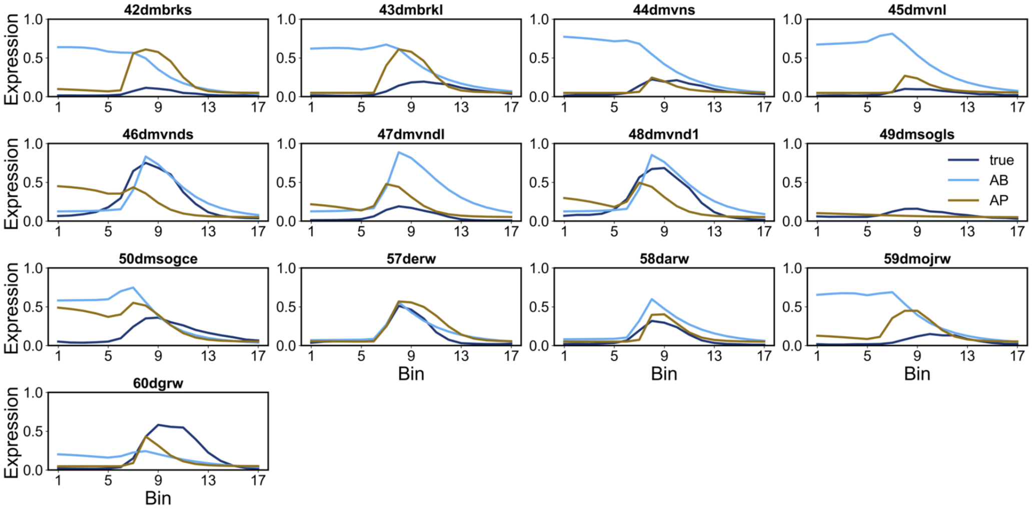

Figure A.2: Predictions of AB and AP models for test enhancers.

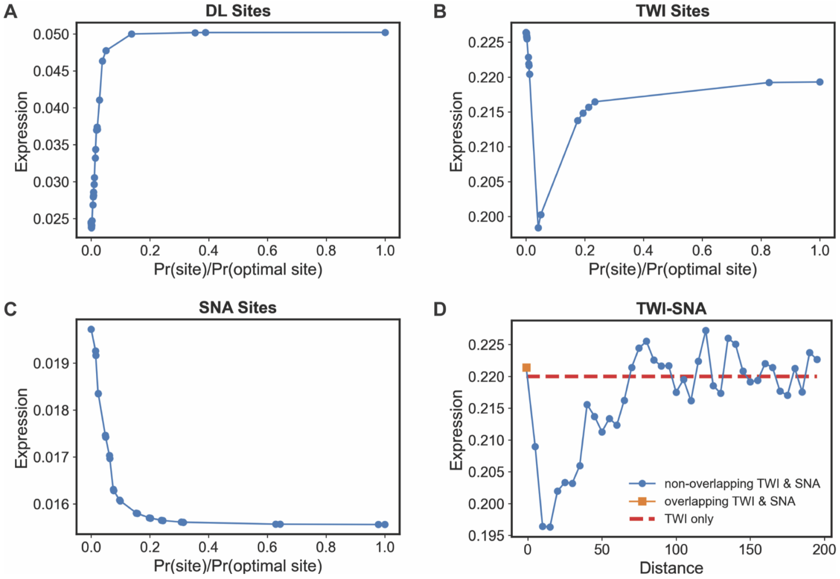

Figure A.3: (Also see Appendix A.1.1) Effect of binding affinities and overlapping binding sites on expression predicted by a trained CoNSEPT model. (A,B) Predicted expression for enhancers containing a single DL site (A) or a single TWI site (B) with varying strengths. (C) Predicted expression for enhancers containing a single SNA site with varying strengths placed at a distance of 20 bp from an optimal DL site. (D) Predicted expression for enhancers with an optimal TWI site and a sub-optimal SNA site (Pr(site)/Pr(optimal site) = 0.01) at various inter-site spacings (Blue), enhancer with an overlapping TWI and SNA site (orange point), and enhancers with a TWI and a SNA site but with SNA at zero concentration (red dashed line).

A.3 SUPPLEMENTARY TABLES

Table A.1: Mapping between names used for enhancers in this study and Sayal et al. [26] study.

Construct's name in [26]	**Name in current study**		**Construct's name in [26]**	**Name in current study**
dmrw	Wild-Type		20dmrt2d3	D4-S5-T2
02dmrd1	D2		21dmrt2d4	D5-S5-T2
03dmrd2	D3		22dmrd3d4	D4-D5
04dmrt1	T1		23dmr4sm	S1-S3-S4-S5
05dmrt2	S5-T2		24dmrs1	S3-S4-S5
06dmrd3	D4		25dmrs2	S1-S4-S5
07dmrd4	D5		26dmrs3	S1-S3-S5
08dmrd1d2	D2-D3		27dmrs4	S1-S3-S4
09dmrd1t1	D2-T1		28dmrs1s2	S4-S5
10dmrd1t2	D2-S5-T2		29dmrs1s3	S3-S5
11dmrd1d3	D2-D4		30dmrs1s4	S3-S4
12dmrd1d4	D2-D5		31dmrs2s3	S1-S5
13dmrd2t1	D3-T1		32dmrs2s4	S1-S4
14dmrd2t2	D3-S5-T2		33dmrs3s4	S1-S3
15dmrd2d3	D3-D4		34dmrb	S2
16dmrd2d4	D3-D5		35dmrd1b	D2-S2
17dmrt1t2	S5-T1-T2		36dmrbt2	S2-S5-T2
18dmrt1d3	D4-T1		37dmrbd3	D4-S2
19dmrt1d4	D5-T1		38dmrbd4	D5-S2

Table A.2: Trained parameters for LM, GLM and GLMQ models. (For GLM and GLMQ the model with the smallest training error is shown).

Model	W_{DL}	W_{TWI}	W_{SNA}	W_{DL-DL}	$W_{TWI-TWI}$	$W_{SNA-SNA}$	W_{DL-TWI}	W_b
LM	8.40e-1	7.20e-2	-3.40e-1	-	-	-	-	1.44e-1
GLM	1e-5	1e-5	-4.35e1	-	-	-	-	-6.21e1
GLMQ	8.11e1	1e-5	-8.32e2	-2.68e2	7.82e2	2.33	-1.01e2	-6.09

Table A.3: Parameters used in different GEMSTAT models in this study.

Model	Repression Parameters	Activator Cooperativity	Common Parameters
AP	DL-SNA & TWI-SNA short-range interactions	DL-DL, TWI-TWI, DL-TWI	q_{BTM}; DNA-binding parameters for DL, TWI, SNA; Activator potency parameters for DL, TWI; SNA-SNA cooperativity
Q	DL-SNA & TWI-SNA short-range interactions		
NR	β_r for SNA		
DIR	SNA-BTM direct interaction		
COOP	(same as NR) β_r for SNA		
NO-COOP	(same as NR) β_r for SNA	none	

Table A.4: Hyper-parameter values used for constructing and training an ensemble of 2016 CoNSEPT models.

Parameter	Values
–cAct (cooperativity activation function)	Tanh , ReLU
–csc (size of cooperativity kernels)	(4,2) , (5,2) , (8,2) , (10,2)
–dr (dropout rate)	0 , 0.25 , 0.5
–nChans (number of channels in the additional convolutional layers)	(36,6) , (24,6) , (24,4) , (12,3) , (6,3), 0
–oAct (output activation function)	Tanh , sigmoid
–psb (pool size for extracting strongest binding sites)	(4,2) , (5,2) , (6,2) , (7,2) , (8,2) , (9,2) , (10,2)

Table A.5: Hyper-parameter setting of the CoNSEPT model with the smallest validation error.

Model Parameters	**Value**
–bs (batch size)	20
–nEpoch (number of training epochs)	1000
–cAct (cooperativity activation function)	relu
–csc (size of cooperativity kernels)	4,2
–dr (dropout rate)	0.5
–nChans (number of channels in the additional convolutional layers)	36,6
–oAct (output activation function)	sigmoid
–psb (pool size for extracting strongest binding sites)	5,2
–sc (stride of cooperativity kernels)	4,2

Table A.6: P-values of Spearman's correlation between the predicted and true expression driven by each of the train, validation and test enhancers for the best-trained GLM, GEMSTAT and CoNSEPT models discussed in Figure 2.8B. Number of significant p-values (No. significant) were computed using a significance threshold of 0.05. The aggregated p-values were computed for each of the train, validation and test groups using Fisher's method.

Model	Train (38 enhancers)		Validation (3 enhancers)		Test (11 enhancers)	
	No. Significant	Aggregated p-value	No. Significant	Aggregated p-value	No. Significant	Aggregated p-value
GLM	34	1.3e-50	3	2.5e-9	10	7.1e-32
GEMSTAT	34	7.2e-72	2	1.1e-6	8	2.0e-20
CoNSEPT	37	9.2e-262	3	1.0e-12	10	4.0e-36

Table A.7: A comparison between the performance of different models used in this study. Asterisks (*) indicate performance on a test set with 11 enhancers that is a subset of the default test set with 13 enhancers. The smaller test set was used since two of the default test enhancers could not be subjected to the CoNSEPT model due to their lengths.

Model	**Train RMSE**	**Validation RMSE**	**Test RMSE**	**Train Correlation**	**Validation Correlation**	**Test Correlation**
LM	0.17	-	0.20	0.61	-	0.66
GLM	0.15	-	0.15	0.63	-	0.72
GLMQ	0.13	-	0.18	0.73	-	0.54
AB	0.10	-	0.32	0.84	-	0.20
AP	0.09	-	0.16	0.72	-	0.59
NR	0.09	-	0.15 0.16*	0.68	-	0.62 0.58*
Q	0.09	-	0.16	0.72	-	0.59
DIR	0.10	-	0.15	0.76	-	0.65
COOP (same as NR)	0.09	-	0.15 0.16*	0.68	-	0.62 0.58*
NO-COOP	0.11	-	0.16	0.63	-	0.65
CoNSEPT	0.08	0.15	0.15*	0.91	0.78	0.75*

APPENDIX B: SUPPLEMENTARY MATERIALS FOR CHAPTER 3

B.1 SUPPLEMENTARY FIGURES

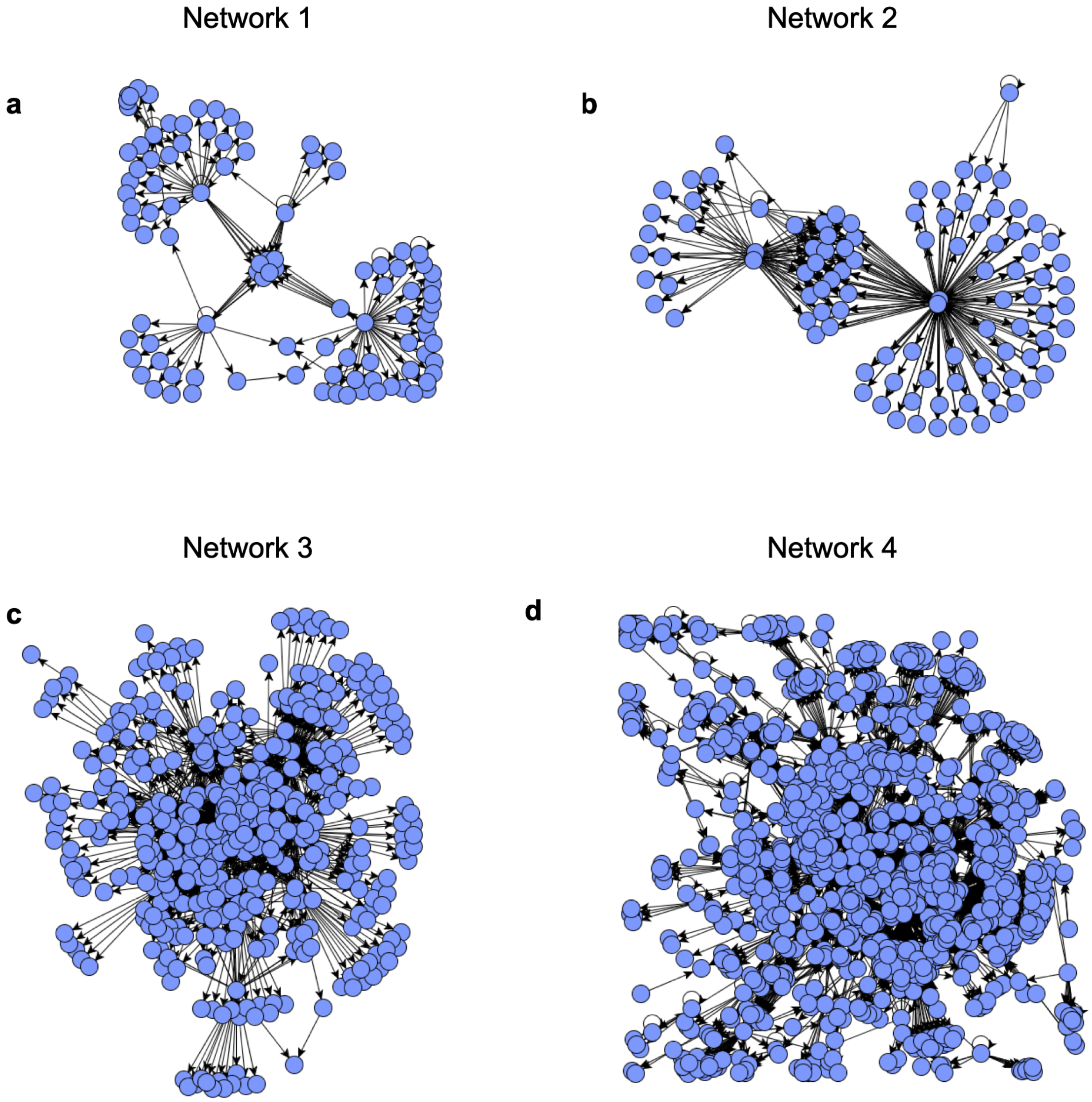

Figure B.1: The structure of four gene regulatory networks used in this study titled by their network ID. These figures were generated using GNW package [114]. Note that all the auto-regulatory edges as well as cycles were removed prior to feeding networks to SERGIO although they are present in this figure. (a) Shows network 1, sampled from E.coli, containing 100 genes and 137 regulatory edges. (b) Shows network ID 2, sampled from E.coli, containing 100 genes and 258 regulatory edges. (c) Shows network ID 3, sampled from S. cerevisiae, containing 400 genes and 1155 regulatory edges. (d) Shows network ID 4, sampled from E. coli, containing 1200 genes and 2713 regulatory edges.

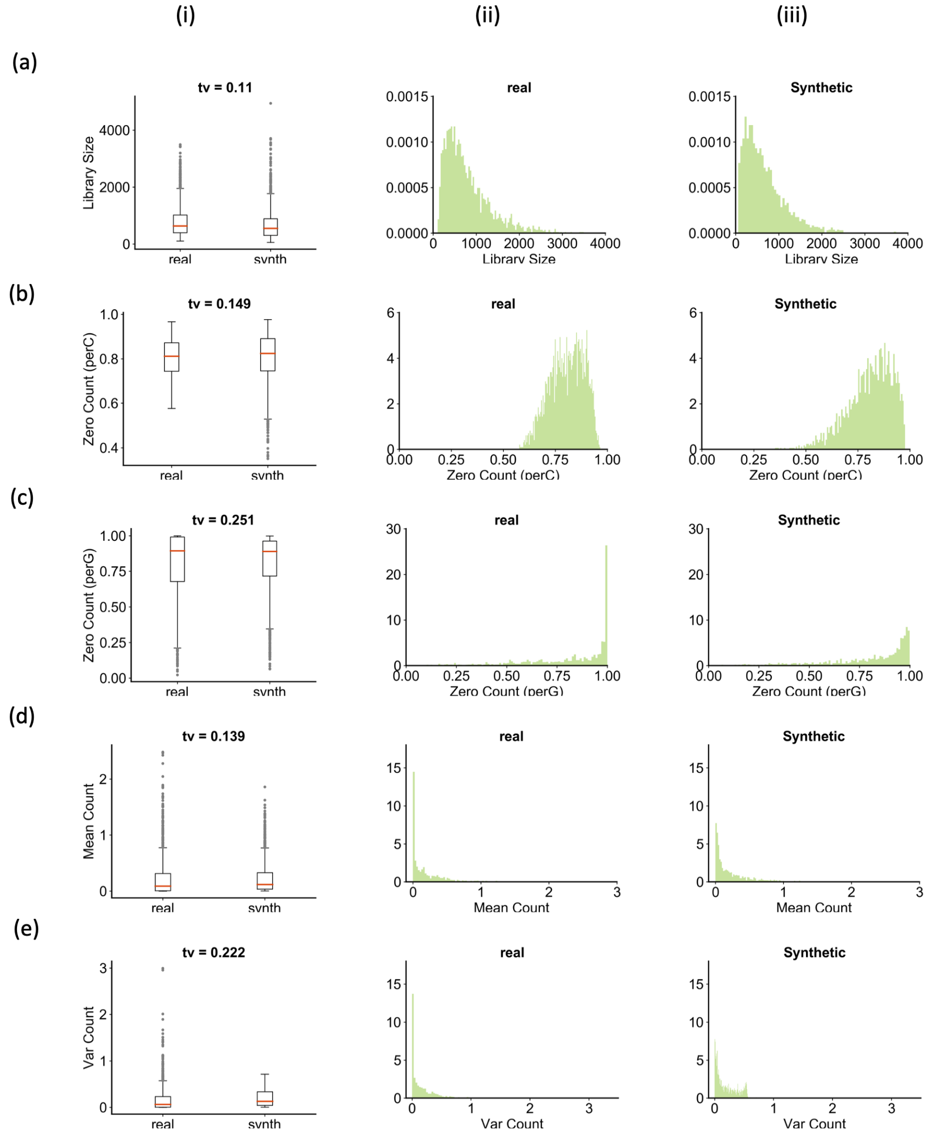

Figure B.2: To interpret the total variation values and assess the quality of match between the real and synthetic data, for each statistic we looked at a pair of simulated replicate of DS3 and a real sample which their total variation is close to the median of the total variations of the corresponding statistic. This figure gives a qualitative understanding of the total variation score, which is a number between 0 and 1 reflecting how well two distributions match. Continued on next page. Each row represents one of the quantities studied in Figure 3.2, and shows the distribution of that quantity in one of the simulated replicates and one of the real data sets; the two data sets selected for display here have a total variation ("tv") that is typical for that quantity. (i) This column compares the distribution of synthetic against the real data as a box plot. (ii) This column shows an alternative visualization of the distribution of the quantity of interest in real data. (iii) This column shows an alternative visualization of the distribution of quantity of interest in the synthetic data. The quantities examined include (a) library sizes (b) zero counts per cell (c) zero counts per gene (d) mean mRNA counts and (e) variance of mRNA counts.

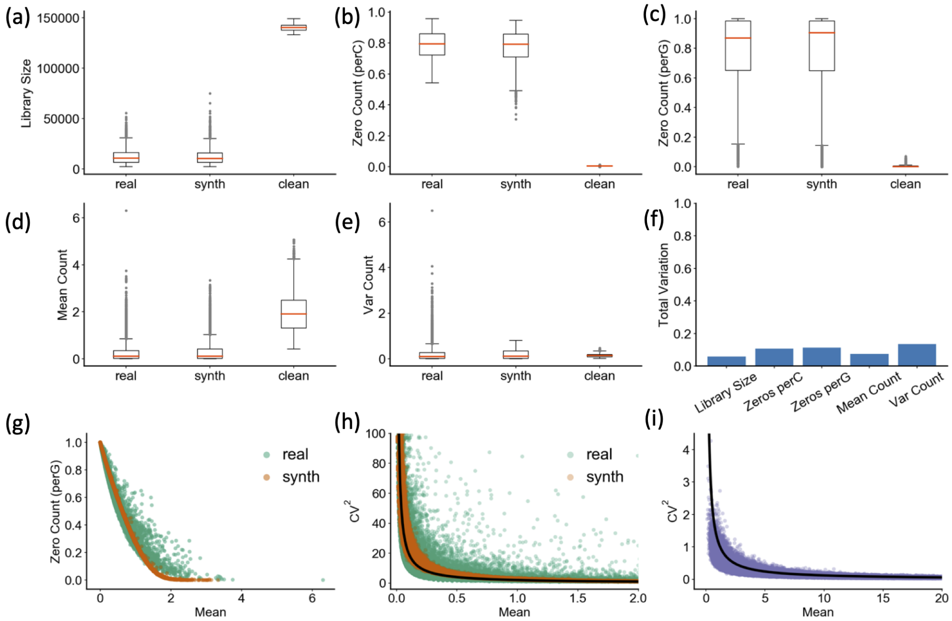

Figure B.3: A curated gene regulatory network for mouse was obtained from RegNetwork database [148]. After preprocessing and excluding the genes that are not present in the mouse brain scRNA-seq data [132] we obtained a gene regulatory network (GRN) containing 15272 genes and 76483 gene-gene interactions. Due to the absence of prior knowledge about the regulatory role of the majority of these interactions, for each interaction we randomly assigned either an activation (probability of 75%) or a repression role (probability of 25%). The interaction strengths were uniformly sampled from the range 1 to 5 (similar to DS1-15) and master regulators' production rates were sampled using the same settings used for DS2-8 to represent nine established cell types. We simulated this GRN using SERGIO to obtain one synthetic expression data containing 15272 genes and 3600 single-cells. Subsequently, we added technical noise by comparing this data against the mouse brain scRNA-seq data set [132] which contains the expression of the same 15272 genes (genes that are not present in the RegNetwork's GRN were excluded) in 3005 single-cells. The quantities examined for adding technical noise include (a) library sizes (b) zero counts per cell (c) zero counts per gene (d) mean mRNA counts and (e) variance of mRNA counts. (f) Total variation between the real and simulated data after adding technical noise (synth) are small (<0.2) and are as good as total variations for the same statistics in DS1-8. (g) The inverse relation between genes' mean expression and zero counts (per gene) present in the real data was reproduced after adding technical noise. (h) Inverse relation between squared coefficient of variation and mean expression of genes over all single-cells is matched between real and simulated data after adding technical noise. The black line shows an arbitrary function of form $y \sim 1/x$ which matches with the observed behavior in both real and synthetic data. As is evident from this plot, highly variable genes in the real data with mean expression >0.15 were not captured in the simulated data. Addition tuning of parameters of SERGIO might help tune the variance of genes and improve the quality of match between the real and synthetic data. (i) The inverse relation of form $y \sim 1/x$ is not a result of technical noise and is also observed in clean simulated data.

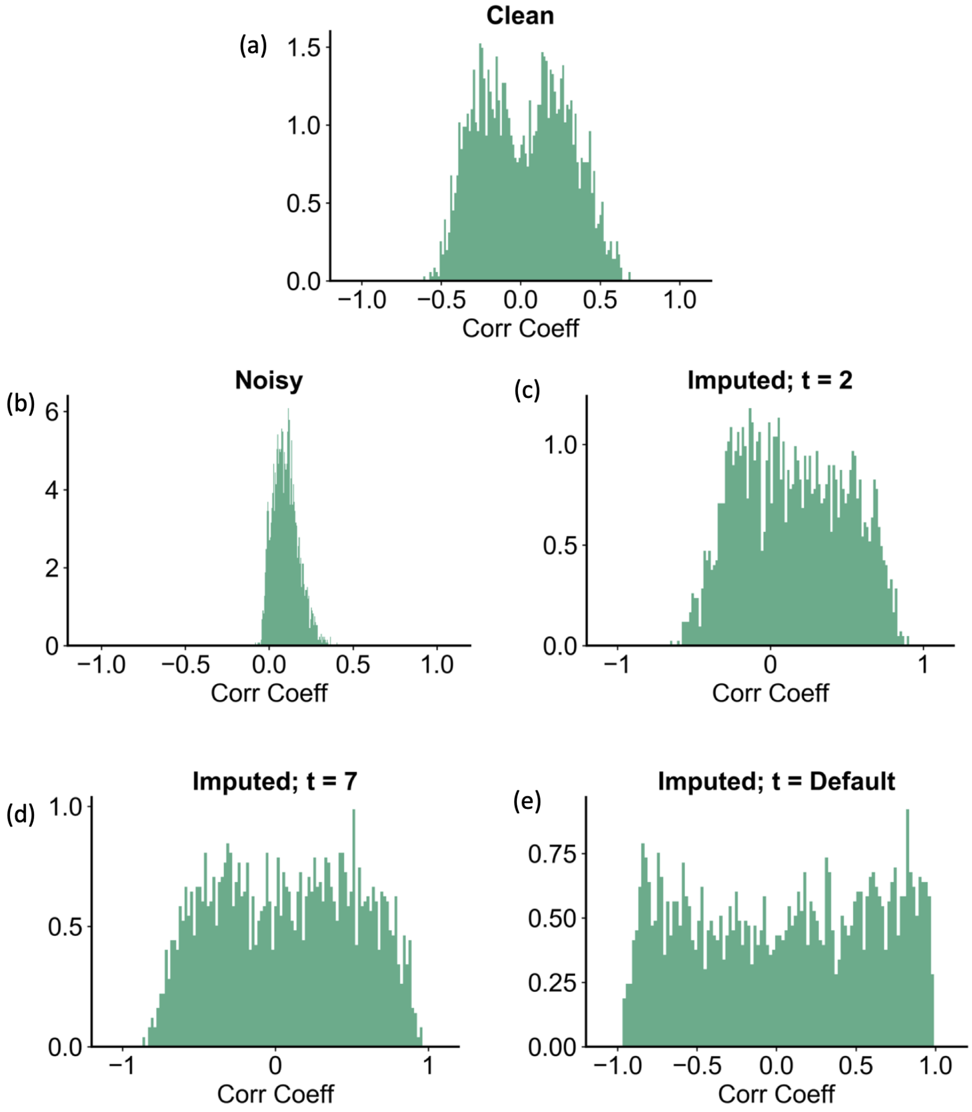

Figure B.4: Shows the distribution of correlation coefficients between all pairs of interacting genes (regulator – target pairs present in the "ground truth" GRN that was used for simulations) in clean and noisy data of one simulated replicate of DS3, as well as in data imputed by MAGIC [92]. (a) Represents the distribution of TF-gene expression correlation coefficients in the clean simulated data. (b) Represents correlation coefficients in the noisy data. After adding technical noise, the co-expression signal in the data (panel a) is severely distorted. (c) Distribution of correlation coefficients in the data underlying panel b, after imputation with MAGIC [92] using parameter setting $t = 2$. Even upon setting t to such a small value, several spurious co-expression signals (right tail of distribution as compared to panel a) emerged in the data, compared to the ground truth shown in panel a. (d) Distribution of correlation coefficients after imputation with MAGIC using $t = 7$. This introduces even more false co-expression signals compared to panel c. (e) MAGIC imputed data with default t setting. We observe almost a uniform distribution over the whole range of correlation coefficients, showing a large number of false positives of co-expressed TF-gene pairs.

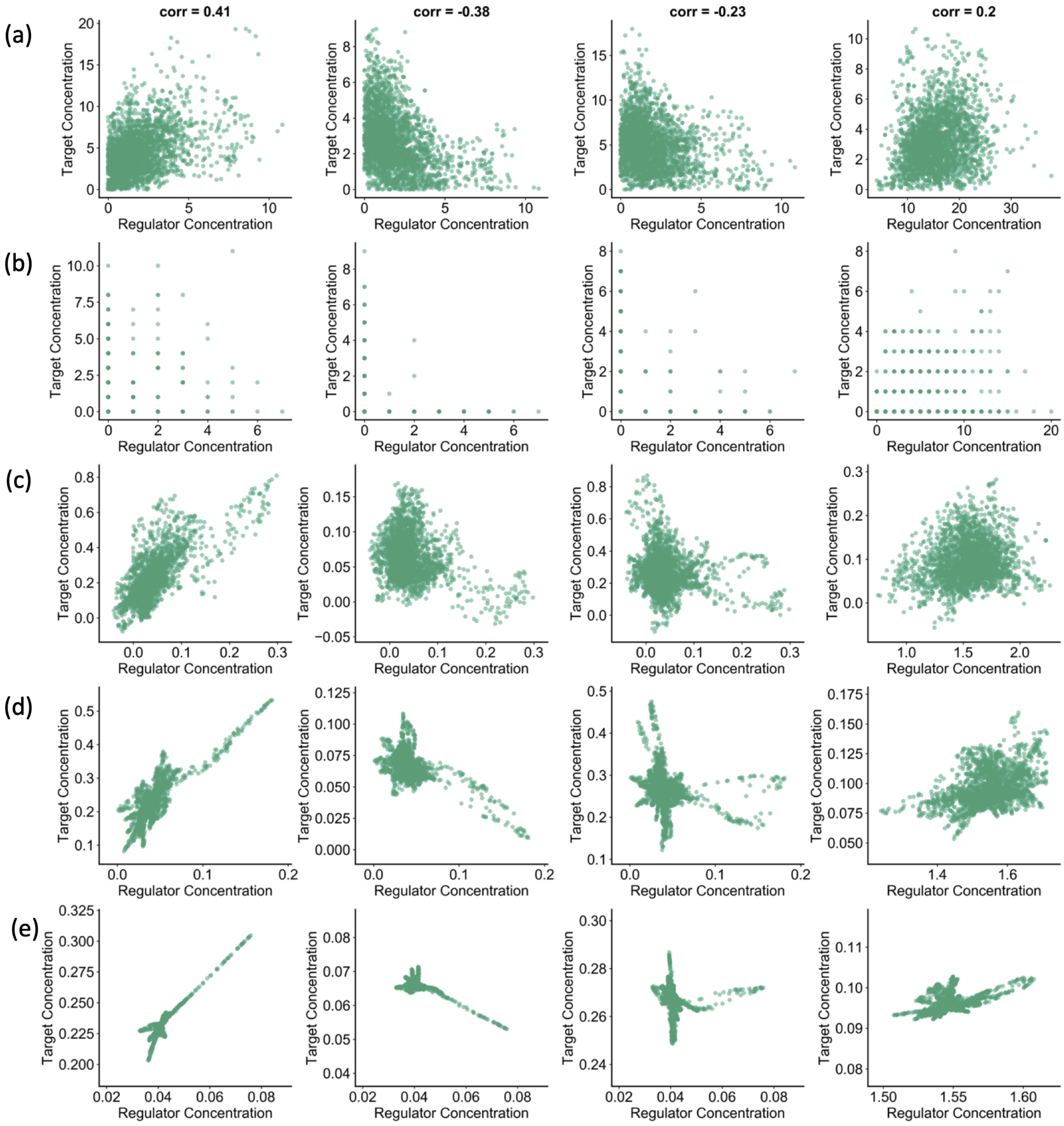

Figure B.5: Correlation structures in clean and noisy simulated data sets as well as imputed versions of the latter. Columns correspond to four arbitrarily selected regulatory interactions (TF-gene pairs) in DS3 (network 4). (a) Clean simulated data. Each panel shows the expressions of the chosen regulator and target pair, in single cells, and the Pearson correlation coefficient between these two observables is noted in caption at the top. (b) TF and target gene expression values for the same TF-gene pairs as in (a), after technical noise has been added. The simulated UMI counts are shown. (c) TF and target gene expression values for the same TF-gene pairs as in (b), after imputed using MAGIC with $t = 2$. Note that level of co-expression appears greater than that in clean data ("ground truth"). (d-e) Same as (c), but with MAGIC run using $t = 7$ and $t = default$ respectively.

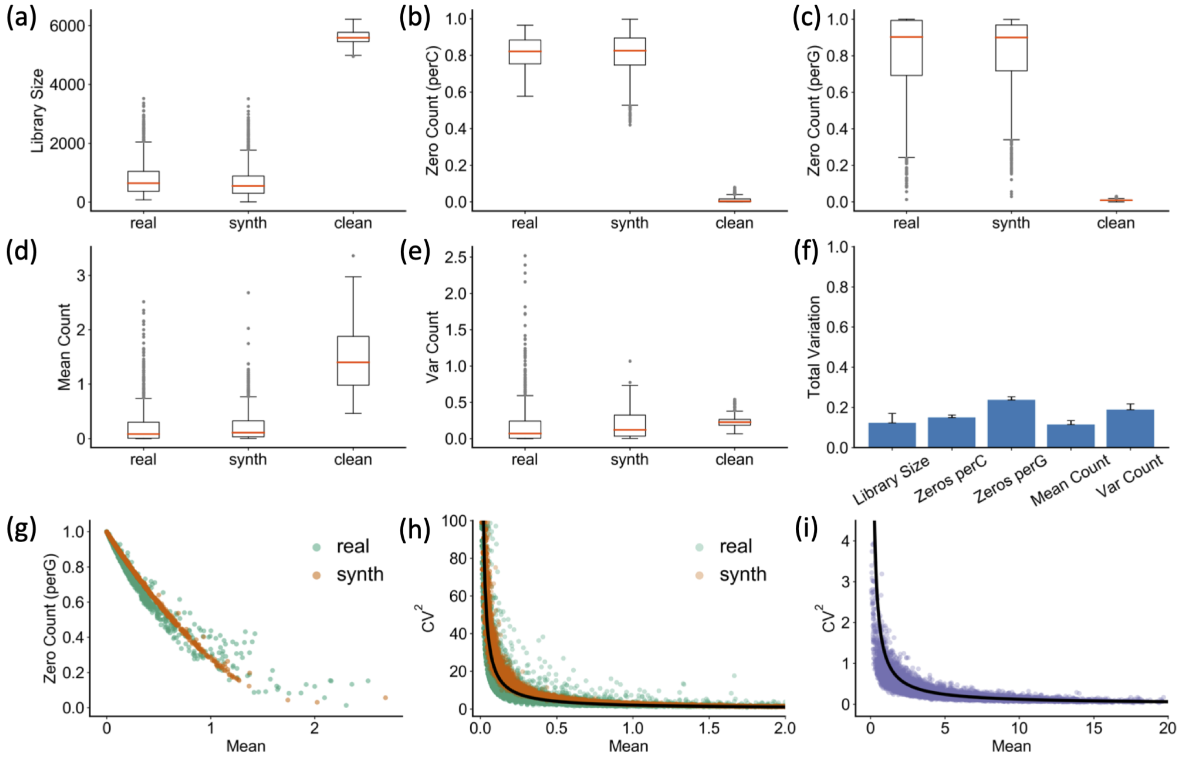

Figure B.6: We used the GRN containing 1200 genes (the same network as that used for DS3-8) to simulate data sets with 15 replicates using the mode of SERGIO that includes activator-activator cooperative regulation. Subsequently, we added technical noise by comparing this data set against 50 samples obtained from the mouse brain scRNA-seq data set [132] (the same samples as those used for adding noise to DS3). The quantities examined for adding technical noise include (a) library sizes, (b) zero counts per cell, (c) zero counts per gene, (d) mean mRNA counts, and (e) variance of mRNA counts. (f) Total variation between each sample and simulated replicate after adding technical noise. (g) The inverse relation between genes' mean expression and zero counts (per gene) present in the real data was reproduced after adding technical noise. (h) Inverse relation between squared coefficient of variation and mean expression of genes over all single-cells is matched between real and simulated data after adding technical noise. The black line shows an arbitrary function of form $y \sim 1/x$ which completely matches with the observed behavior in both real and synthetic data. (i) The inverse relation of form $y \sim 1/x$ is not a result of technical noise and is also observed in clean simulated data.

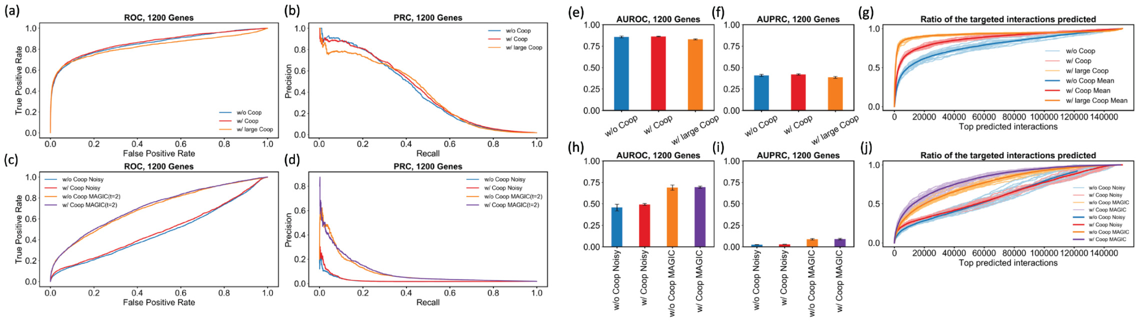

Figure B.7: Comparing the performance of GENIE3 on three clean simulated data sets, namely a data set without cooperative regulation (DS3; called "w/o Coop" here), a data set with moderate cooperative regulation ("w/ Coop"), and a data set with large cooperative regulation ("w/ large Coop") dominating non-cooperative effects. All three data sets were simulated with the same underlying GRN (Network ID 4) and in 15 replicates. (a) ROC and (b) PRC of GRN prediction by GENIE3 (on one simulated replicate from each data set) shows that inclusion of cooperative regulation does not impact GRN inference. We next added technical noise to the data set with moderate cooperativity (w/ Coop), in a way that matches the noise in a mouse brain scRNA-seq data set [132] (Figure B6). GENIE3 was applied on this noisy data set before and after imputation by MAGIC ($t = 2$). (c) ROC and (d) PRC of GRN prediction by GENIE3 (on one simulated replicate from each data set) confirms that without imputation, GRN inference from noisy data is not impacted by inclusion of cooperative regulation (compare "w/o Coop noisy" to "w/ Coop noisy"). Moreover, similar to our observations on data without cooperative regulation, using MAGIC ($t = 2$) to impute noisy data increases signal for GRN inference by GENIE3, even when the data was generated by a GRN with cooperative regulation (compare "w/ Coop noisy" to "w/ Coop MAGIC(t=2)")). Mean AUROC (e) and AUPRC (f) over 15 replicates of each of the three clean simulated data sets and mean AUROC (h) and AUPRC (i) over 15 replicates of each of the four noisy simulated data sets (two before imputation and two after imputation by MAGIC) show the same trend discussed in (a-d). We next evaluated the enrichment of regulator-target interactions that are affected by cooperativity (e.g., interactions B-A and C-A are said to be affected by cooperativity if B and C cooperatively regulate A) among the top-k predictions of GENIE3 obtained from data sets simulated with cooperative regulation. Also, to assess the impact of cooperativity on this enrichment, we collected the interactions affected by cooperativity (in cooperativity simulations) and evaluated their enrichments among GENIE3 predictions obtained from simulated data in the absence of cooperativity. Note that this is feasible because we used the same GRN topology in the two modes of simulation. (g) Shown is the fraction of such interactions recovered in the top-k predictions (x axis) of GENIE3 applied to clean simulated data sets. Although inclusion of cooperative regulation does not impact GRN inference from simulated data (e.g., Figure B7 a-b), interactions that were affected by cooperativity are more enriched among the top GENIE3 predictions as compared to the same interactions in the absence of cooperativity. (j) Shows the enrichment of interactions affected by cooperativity among the top-k predictions of GENIE3 applied to noisy simulated data sets before and after imputation by MAGIC. Imputation by MAGIC increases the enrichment of such interactions (purple versus red curve).

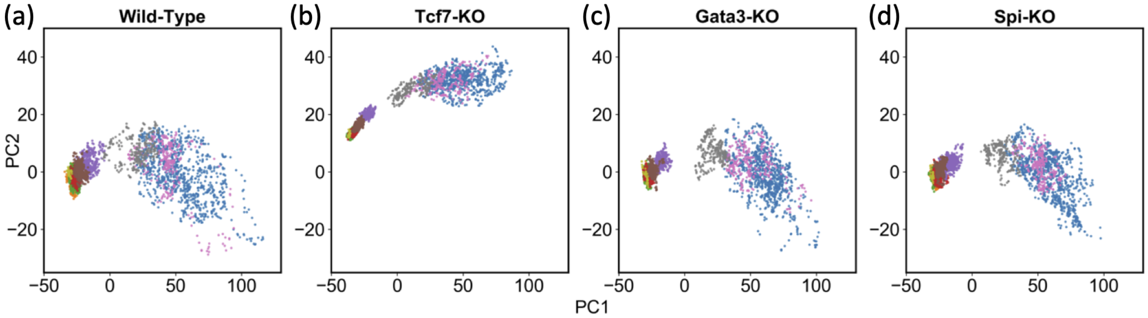

Figure B.8: PC representation of wild-type (WT) and knockout (KO) simulated trajectories using GRN obtained by GENIE3 (a) Two-dimensional PC representation of WT trajectory (identical to Figure 3.5A, right). (b) Projection of Tcf7-KO simulated trajectory on the PC space of WT trajectory. The average Euclidean distance between cluster centers of Tcf7-KO and WT trajectories in 10 dimensional PC space is 8.2. (c) Projection of Gata3-KO simulated trajectory on the PC space of WT trajectory. The average Euclidean distance between cluster centers of Gata3-KO and WT trajectories in 10 dimensional PC space is 1.0. (d) Projection of Spi-KO simulated trajectory on the PC space of WT trajectory. The average Euclidean distance between cluster centers of Spi-KO and WT trajectories in 10 dimensional PC space is 1.2.

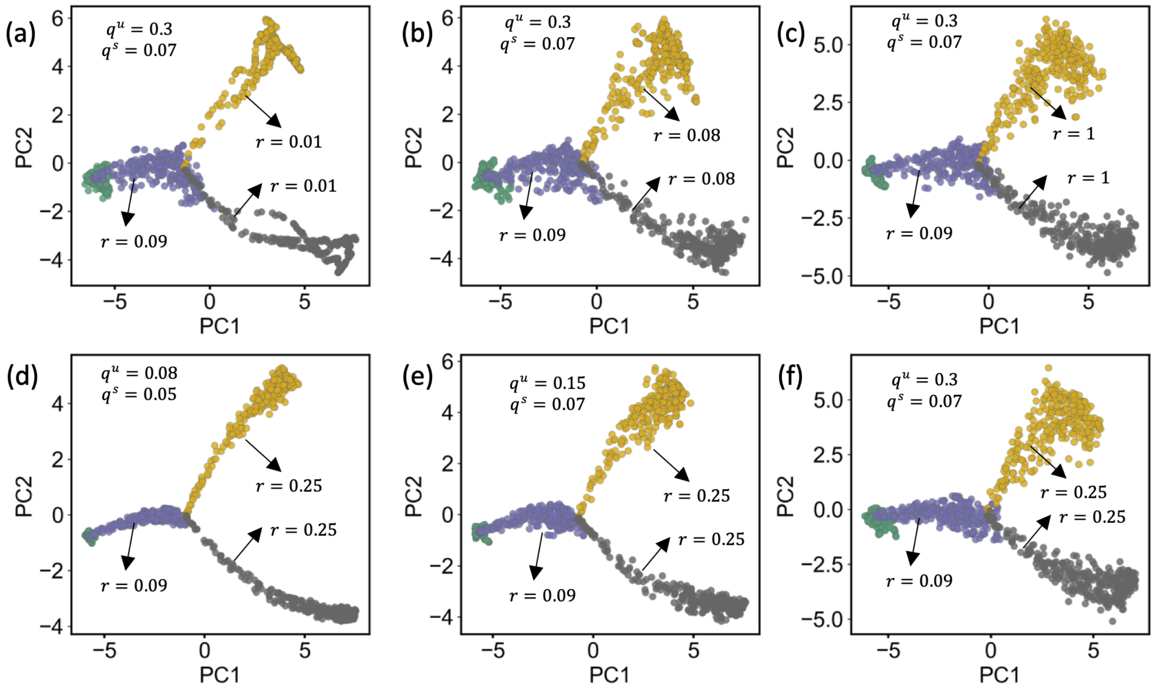

Figure B.9: User can control the thickness of differentiation path and the dispersion of cells around the trajectory. The user defined migration rate r controls the number of paths that are simulated between two cell types (each edge in the provided differentiation graph). For a given number of cells per cell type (nCells) and migration rate r, a total number of $r \times nCells$ paths is simulated between the two cell types. Finally, single-cells are randomly sampled from the aggregation of all simulated paths. (a,b,c) For fixed unspliced and spliced noise amplitudes q^u and q^s respectively, increasing the migration rate r increases the thickness of the simulated differentiation path as single-cells are sampled from a bigger pool of cells in between the two origin and end cell types. (d,e,f) For a fixed migration rate r, increasing the spliced and unspliced noise amplitudes increases the dispersion of single cells because the higher stochastic noise increases the variance among single-cells.

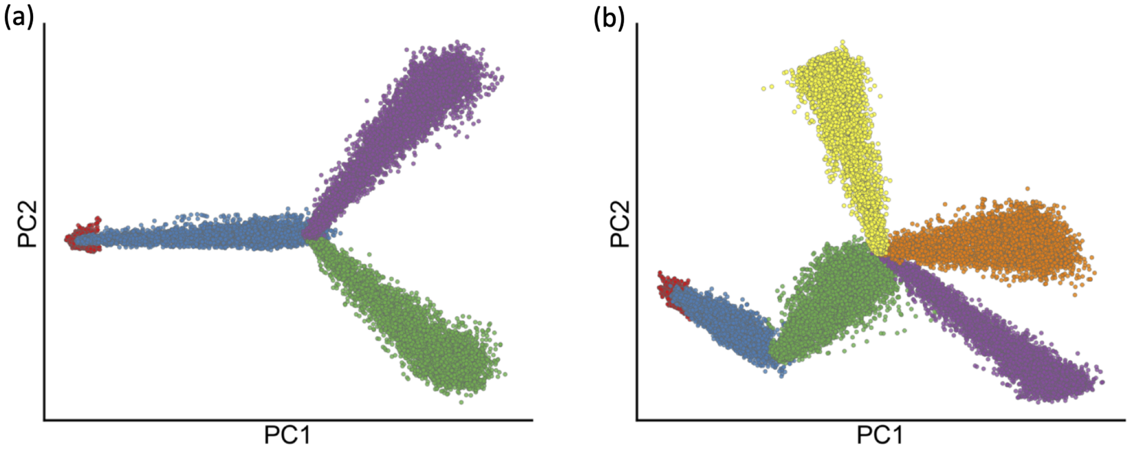

Figure B.10: (a) PCA representation of single cells in the clean simulated version of DS13. This data set contains 24000 cells in total. For simulating DS13 we used the same GRN, parameter settings, and differentiation graph as we used for DS10. (b) PCA representation of single cells in the clean simulated version of DS14. This data set contains 36000 cells in total. For simulating DS14 we used the same GRN, parameter settings, and differentiation graph as we used for DS11.

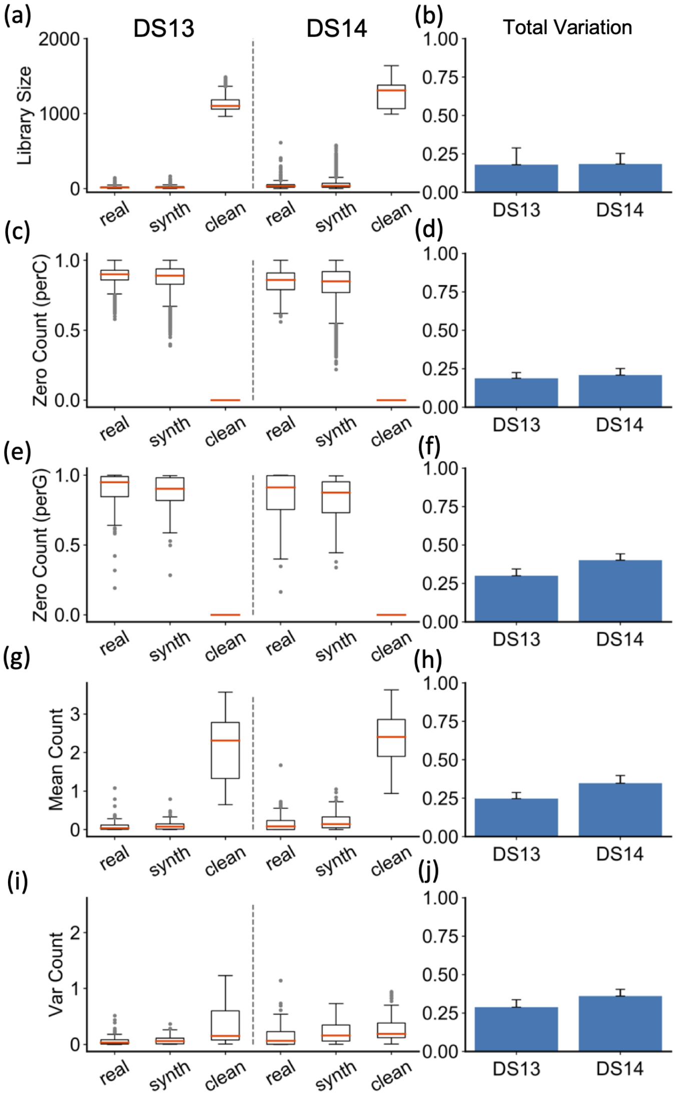

Figure B.11: Comparisons between differentiation data sets generated by SERGIO and real scRNA-seq data sets. We show the distributions of per-cell quantities in (a,c), and per-gene quantities in (e, g, i), for DS13 and DS14 separated by dashed lines. These comparisons are shown between one sample from the real data set ("real"), the clean simulated data ("clean"), and its technical noise-added version ("synth"). The real data used for DS13 is a published 10X genomics single-cell data of dentate gyrus of mouse hippocampus [137], and for DS14 we used a single-cell RNA-seq data set from the mouse cerebral cortex [132]. More comprehensive comparisons – between the noisy simulated data and every real sample – are shown in panels to the right: the total variation (see section 3.3) is calculated to compare the real and synthetic distributions and the average total variation across all comparisons is shown in panels (b, d, f, h, j). (a,b) Distributions and total variation of library sizes. (c,d) Distributions and total variation of zero counts per cell (normalized by number of genes). (e,f) Distributions and total variations of zero counts per gene (normalized by total number of cells). (g,h) Distributions and total variations of genes' mean expression. (i,j) Distributions and total variations of genes' expression variances.

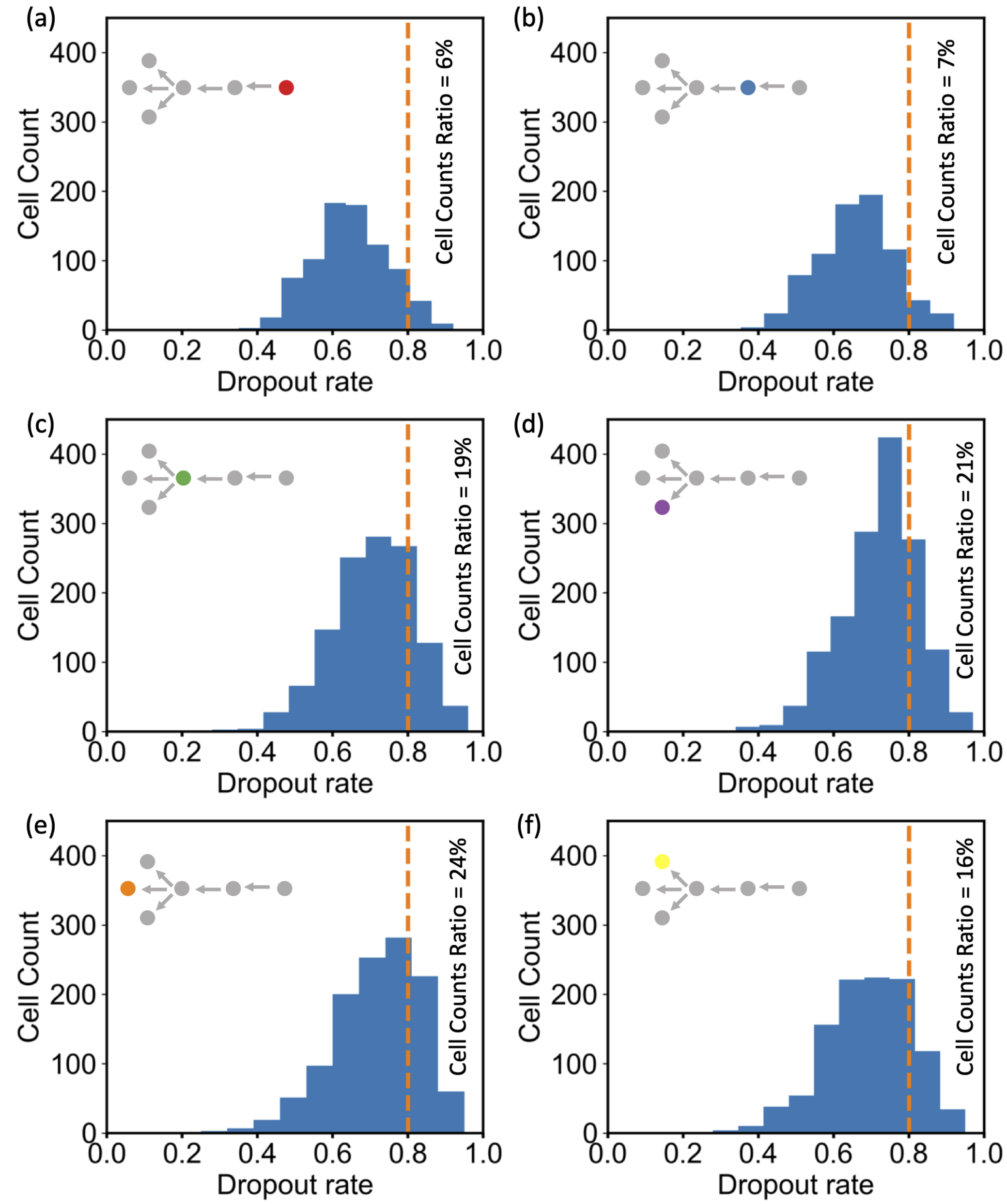

Figure B.12: Distributions of dropout rates in single cells belonging to (a) red, (b) blue, (c) green, (d) purple, (e) orange, and (f) yellow, cell types shown in Figure 3.7. For each cell type, the ratio of cells which have 80% or more dropout rate is denoted. The orange cell type suffers the most from dropout and its distribution shows the most skewness toward large dropout rates as compared to other cell types. This is consistent with the poor correlation observed for this cell type, between the inferred pseudotime from the noisy and clean expression matrices (Figure 3.7G).

B.2 SUPPLEMENTARY TABLES

Table B.1: Technical noise parameters used in this study.

DS-ID	**Outlier Genes**			**Library Size**		**Dropouts**		**Low Quality threshold ***
	π^O	μ^O	σ^O	μ^L	σ^L	k	q	τ
1	0.01	0.8	1	4.8	0.3	20	82	5
2	0.01	0.8	1	6	0.4	12	80	5
3	0.01	0.8	1	7	0.4	8	80	5
4	0.01	3	1	6	0.3	8	74	5
5	0.01	3	1	6	0.4	8	82	5
6	0.01	5	1	4.5	0.7	8	45	5
7	0.01	3	1	4.4	0.8	8	85	5
8	0.01	4.5	1	10.8	0.55	2	92	2500
13	0.01	0.8	1	3.6	0.4	8	70	5
14	0.01	0.8	1	5	0.4	4	80	5

(*) Cells with a total count $< \tau$ were considered as low quality cells and were removed from both real samples and synthetic replicates.

Table B.2: Parameter settings used for running Singe [141].

Parameter	**Value**
λ	0, 0.1, 0.01
(dT,num_lags)	(3,5), (5,9), (9,5), (5,15), (15,5)
kernel_width	0.5, 1, 2, 4
prob_zero_removal	0
prob_remove_samples	0.2
num_replicates	10

Table B.3: Low and high expression ranges from which the master regulators' production rates were sampled.

DS-ID	**Low Expression Range**	**High Expression Range**
DS1	[0.2 0.5]	[0.7 1]
DS2-8	[0 2]	[2 4]
DS9-15	[0 1]	[3 4]

Table B.4: A comparison between running times of SERGIO and BoolODE.

Experiment	**Network ID**	**Number Genes**	**Number Cell Type-/Simulation Time**	**SERGIO time (s)**	**BoolODE time (s)**
1	4	1200	9	4607	8179
2	4	1200	1	733	1058
3	3	400	1	132	1145
4	2	100	1	28	75

APPENDIX C: SUPPLEMENTARY MATERIALS FOR CHAPTER 4

C.1 SUPPLEMENTARY NOTES

C.1.1 Rules of do-calculus

Assume all the causal associations between pairs of variables are represented by a Directed Acyclic Graph (DAG) G. Let Y denote the outcome variable of interest and T,Z, and W represent three disjoint set of nodes, excluding Y, in G. Also, let $G_{\overline{V}}$, represent the graph obtained by removing all the incoming edges of V, where V denotes a subset of nodes in G, and $G_{\underline{V}}$ represent the graph obtained by removing all the outgoing edges of V. Using these notations, we may write the second and third rules of do-calculus [183] as:

Rule 2:

$$P(Y|do(T), do(Z), W) = P(Y|do(T), Z, W) \quad \text{if} \quad Y_{G_{\overline{T}\underline{Z}}} Z|T, W \tag{C.1}$$

The notation $Y_{G_{\overline{T}\underline{Z}}} Z|T, W$ should be read as: *Y and Z are d-separated by conditioning on $T \cup W$ in graph $G_{\overline{T}\underline{Z}}$*, i.e., the graph obtained by removing all the incoming edges of T and all the outgoing edges of Z. Two variables a and b are ”d-separated” by c if there exists no collider-free path between a and b in the causal graph that does not visit c, i.e., if c ”blocks” every collider-free path between the variables. (A ”collider” is a node in a graph where two arrows ”collide head-to-head”.)

Rule 3:

$$P(Y|do(T), do(Z), W) = P(Y|do(T), W) \quad \text{if} \quad Y_{(G_{\overline{T}})_{\overline{Z(W)}}} Z|T, W \tag{C.2}$$

The notation $Y_{(G_{\overline{T}})_{\overline{Z(W)}}} Z|T, W$ should be read as: *Y and Z are d-separated by conditioning on $T \cup W$ in graph $(G_{\overline{T}})_{\overline{Z(W)}}$, i.e., the graph obtained by first removing all incoming edges to T, obtaining the graph $G_{\overline{T}}$, and then removing all incoming edges into the set $Z(W)$, defined as the set of nodes in Z that are not an ancestor of any node in W in graph $G_{\overline{T}}$.*

Informally speaking, rule 2 of do-calculus allows the conversion of a do-term to a conventional statistical conditioning and rule 3 allows the elimination of a do-term from the causal quantity.

C.1.2 Lemma 4.1 and proof

Lemma 4.1: Consider the causal structure ψ shown as a DAG G in Figure C.1A, where a subset of the variables $\{X_j\}$ influence the outcome variable Y, and the variables $\{X_j\}$ are in turn influenced by a variable Z'. The variable X_i influences Y, $A(\psi)$ represents the indices of other variables that are associated with Y, and $NA(\psi)$ represents the indices of variables that are not associated with Y. The union of all directed associations from variables in $X_{NA(\psi)}$ to nodes in $X_{A(\psi)\cup\{i\}}$ (resp. from nodes in $X_{A(\psi)\cup\{i\}}$ to nodes in $X_{NA(\psi)}$) is shown by a directed edge from $X_NA(\psi)$ to $X_{A(\psi)\cup\{i\}}$ (resp. from $X_{A(\psi)\cup\{i\}}$ to $X_{NA(\psi)}$). (Note: although the directed edges shown between $X_{A(\psi)\cup\{i\}}$ and $X_{NA(\psi)}$ suggest the existence of a cycle, the actual directed edges represented by these two edges are assumed not to form a cycle.) For this causal structure,

$$E\left[Y\middle|do\left(X_{A(\psi)\cup\{i\}}=x_{A(\psi)\cup\{i\}}\right),do\left(X_{NA(\psi)}=x_{NA(\psi)}\right)\right]=E\left[Y\middle|do\left(X_{A(\psi)\cup\{i\}}=x_{A(\psi)\cup\{i\}}\right)\right] \tag{C.3}$$

Proof: This lemma is a direct result of the third rule of do-calculus. We define $Z=X_{NA(\psi)}$, $T=X_{A(\psi)\cup\{i\}}$, and $W=\emptyset$; thus, in G, all direct edges into Y are from $T=X_{A(\psi)\cup\{i\}}$ and none from $Z=X_{NA(\psi)}$. We first obtain graph $G_{\overline{T}}$ of equation C.2 by removing edges incoming into $T=X_{A(\psi)\cup\{i\}}$ (Figure C.1B). Since W is empty, we have $Z\left(W\right)=Z$, where $Z(W)$ is defined as the set of nodes in Z that are not an ancestor of any node in $W=\emptyset$. Thus, $(G_{\overline{T}})_{\overline{Z(W)}}$ is obtained from $G_{\overline{T}}$ by further removing all edges coming into $Z=X_{NA(\psi)}$, giving us the graph shown in Figure C.1C. Note that this derivation of $(G_{\overline{T}})_{\overline{Z(W)}}$ also removes all causal dependencies among variables $\{X_j\}$, regardless of their direction. Thus, in the resulting graph shown in Figure C.1C, there is no path of association between $Z=X_{NA(\psi)}$ and Y and therefore the condition of the third rule of do-calculus is satisfied and we get:

$$E[Y|do\left(T\right),do\left(Z\right),W]=E[Y|do\left(T\right),W] \tag{C.4}$$

and thus, by substituting the definitions $T=X_{A(\psi)\cup\{i\}}$, $Z=X_{NA(\psi)}$, $W=\emptyset$, we have

$$E\left[Y\middle|do\left(X_{A(\psi)\cup\{i\}}=x_{A(\psi)\cup\{i\}}\right),do\left(X_{NA(\psi)}=x_{NA(\psi)}\right)\right]=E\left[Y\middle|do\left(X_{A(\psi)\cup\{i\}}=x_{A(\psi)\cup\{i\}}\right)\right] \tag{C.5}$$

This completes the proof of Lemma 4.1.

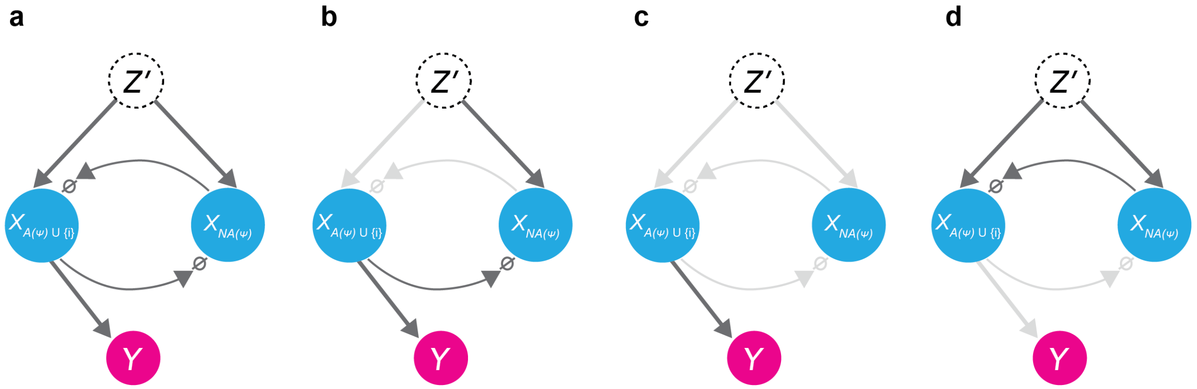

Figure C.1: (A) Causal DAG, G, representing the causal structure ψ consistent with graph in Supplementary Figure C.2. Associations between $X_{A(\psi)\cup\{i\}}$ and $X_{NA(\psi)}$ are represented by two "$\to \varnothing$" edges to emphasize that they do not introduce cycles. (B) Graph $G_{\overline{T}}$ in the notations used for the proof of Lemma 4.1. (C) Graph $(G_{\overline{T}})_{\overline{Z(W)}}$ in the notations used for the proof of Lemma 4.1. (D) Graph $G_{\underline{Z}}$ in the notations used for the proof of Lemma 4.2.

C.1.3 Lemma 4.2 and proof

Lemma 2: Consider the causal structure ψ, represented by DAG G (Figure C.1A), with variables and associations as defined in Lemma 4.1. Then

$$E\left[Y\middle|do\left(X_{A(\psi)\cup\{i\}} = x_{A(\psi)\cup\{i\}}\right)\right] = E\left[Y\middle|X_{A(\psi)\cup\{i\}} = x_{A(\psi)\cup\{i\}}\right] \tag{C.6}$$

Proof: This Lemma is a direct result of the second rule of do-calculus. Using the notations of equation C.1, we define $Z = X_{A(\psi)\cup\{i\}}$, and $T = W = \emptyset$. We obtain graph $G_{\overline{T}\underline{Z}} = G_{\underline{Z}}$ of equation C.1 by removing edges outgoing from $Z = X_{A(\psi)\cup\{i\}}$ in G. Since in the resulting graph shown in Figure C.1D, there is no path of association between $Z = X_{A(\psi)\cup\{i\}}$ and Y, the condition of the second rule of do-calculus is satisfied and we get:

$$E[Y|do\left(T\right), do\left(Z\right), W] = E[Y|do\left(T\right), Z, W] \tag{C.7}$$

and thus, by substituting the definitions $T = W = \emptyset$, $Z = X_{A(\psi)\cup\{i\}}$, we have

$$E\left[Y\middle|do\left(X_{A(\psi)\cup\{i\}} = x_{A(\psi)\cup\{i\}}\right)\right] = E\left[Y\middle|X_{A(\psi)\cup\{i\}} = x_{A(\psi)\cup\{i\}}\right] \tag{C.8}$$

This completes the proof of Lemma 2.

C.1.4 Shapley Value: explaining a multi-player game

Assume $M = \{1..m\}$ is a set of m players contributing to a game whose outcome is measured by a value function v:

$$v(S) \in \mathbb{R} \quad \forall S \subseteq M \qquad \text{and} \qquad v(\emptyset) = 0 \tag{C.9}$$

Contribution of player $i \in M$ to the game's outcome can be computed by Shapley value [242] of the i^{th} player:

$$\phi_i = \sum_{S \subseteq M \setminus \{i\}} \frac{1}{m\binom{m-1}{|S|}} \Big(v(S \cup \{i\}) - v(S) \Big) \tag{C.10}$$

Shapley value defined as above is a fair contribution allocation method as it satisfies several desirable properties discussed elsewhere [180, 242].

C.1.5 SHAP value: Explaining an arbitrary function

Next, the question is how to use Shapley value for explaining the output of an arbitrary function. Assume we have an observational dataset $\mathcal{D} = \Big\{(\mathcal{D}_X, \mathcal{D}_Y)_j\Big\}_{j \in \{1..n\}} = \Big\{(\{X_i\}_{i \in M}, \{Y\})_j\Big\}_{j \in \{1..n\}}$, where $M = \{1..m\}$, containing n observations for an outcome of interest Y and m covariates $X = \{X_i\}_{i \in M}$ that are independent and identically distributed (IID) samples from a distribution $P(X, Y)$, $\; X \in \mathbb{R}^m, Y \in \mathbb{R}$. Also, we trained a machine learning model f on the data through regression to predict Y from X, so we have:

$$\forall x \sim P(X), \quad f(x) = E[Y|X = x] \tag{C.11}$$

where $P(X)$ denotes the marginal distribution of covariates. The goal is to measure the contribution of the i^{th} dimension (henceforth called "feature i") of input $x \in \mathcal{D}_X$ to the function's output at $X = x$, i.e., $f(x)$. In order to employ the idea underlying equation C.10, we need to define how a function f may be evaluated on a subset $S \subseteq M$ of features, i.e., $f(x_S)$. By having a well-defined notion for $f(x_S)$ we can extend equation C.10 to measure the contribution of feature i to the output of function f locally at $X = x$ as follows:

$$\phi_i(f, x) = \sum_{S \subseteq M \setminus \{i\}} \frac{1}{m\binom{m-1}{|S|}} \Big(f(x_{S \cup \{i\}}) - f(x_S) \Big) \tag{C.12}$$

Lundberg and Lee [180] proposed to define

$$f(x_S) = E[f(X)|X_S = x_S] = E_{P(X_{\bar{S}}|X_S=x_S)}[f(x_S, X_{\bar{S}})] \tag{C.13}$$

where $\bar{S} = M \setminus S$. Later, Janzing et al. [182] modified this notion by defining $f(x_S) = E[Y|do(X_S = x_S)]$ which under their assumed causal graph is simplified to $E[Y|do(X_S = x_S)] = E_{P(X_{\bar{S}})}[f(x_S, X_{\bar{S}})]$, where $do(.)$ represents the Pearl's do-operator [191]. This interpretation of SHAP values was later adopted by Lundberg et al. [168]. By using the notion proposed by Janzing et al. in equation C.12 we obtain:

$$\phi_i(f, x) = \sum_{S \subseteq M \setminus \{i\}} \frac{1}{m\binom{m-1}{|S|}} \bigg(E_{X_{\bar{S}\setminus\{i\}}}[f(x_{S\cup\{i\}}, X_{\bar{S}\setminus\{i\}})] - E_{X_{\bar{S}}}[f(x_S, X_{\bar{S}})] \bigg) \tag{C.14}$$

In order to relate to the language used in Methods (section 4.3), we define $S = A(\psi)$, where ψ is the causal structure in which only the subset $S \subseteq M \setminus \{i\}$ of covariates (i.e., X_S) and feature i (i.e., X_i) are associated with outcome Y. Thus, we have $\bar{S} = NA(\psi) \cup \{i\}$. By substituting these notations in equation C.14 we get:

$$\begin{aligned}\phi_i(f, x) = \sum_{A(\psi) \subseteq M \setminus \{i\}} \frac{1}{m\binom{m-1}{|A(\psi)|}} (E_{X_{NA(\psi)}}[f(X_i = x_i, X_{A(\psi)} = x_{A(\psi)}, X_{NA(\psi)})] - \\ E_{X_{NA(\psi)\cup\{i\}}}[f(X_{A(\psi)} = x_{A(\psi)}, X_{NA(\psi)\cup\{i\}})]) \end{aligned} \tag{C.15}$$

which is equivalent to our approximation, $\alpha_i(x)$ formalized in Methods (section 4.3), for estimating local causal feature association defined by $LTE_i(x)$ in equation 4.4 in Methods (subsection 4.3.1). Therefore, SHAP value [168, 180] $\phi_i(f, x)$, under the interpretation provided by Janzing et al. [182], is equivalent to $\alpha_i(x)$ defined in equation 4.14.

C.2 SUPPLEMENTARY FIGURES

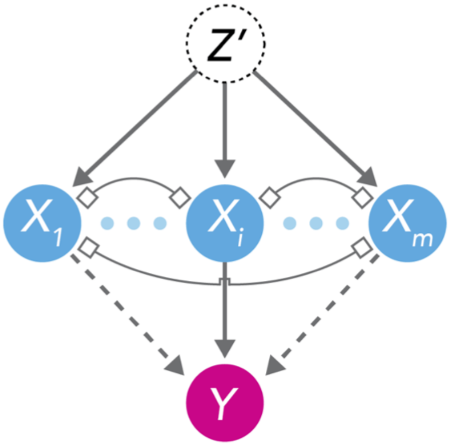

Figure C.2: A causal diagram for feature attribution problem. The diagram includes m observed covariates, $\{X_i\}_{i\in\{1..m\}}$, and an outcome of interest, Y. The goal is to identify the causal association between covariate X_i, and Y (solid arrow), while the associations between other covariates and Y are not known (dashed arrows). Causal dependencies among covariates are assumed with unknown directionality (lines with square ends). Confounder variable Z' is not observed and is assumed to be causally associated with all covariates but not with the outcome.

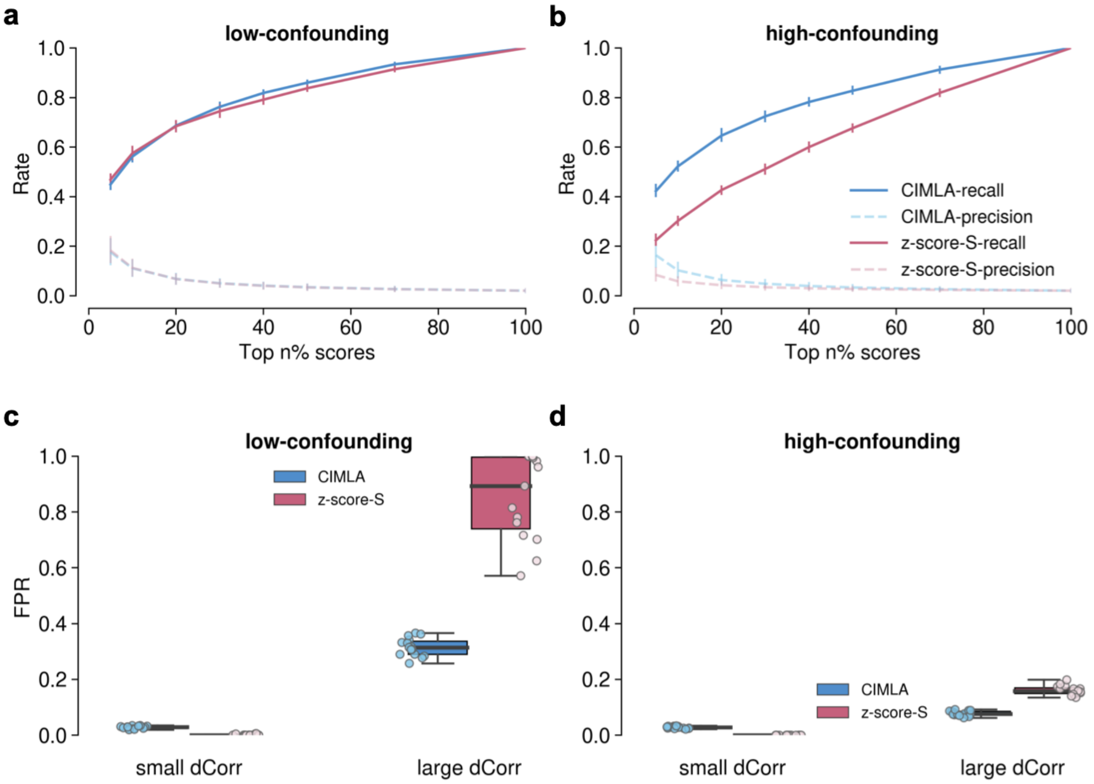

Figure C.3: (A,B) Precision and recall rates for various cutoffs on the top predictions made by CIMLA and z-score-S on (A) low-confounding, and (B) high-confounding simulated data (error bars reflect variations among 15 simulated datasets). (C,D) False Positive Rate (FPR) for the top 5% predictions of CIMLA and z-score-S for the task of discriminating differential edges from non-differential edges. Evaluations are done separately for all TF-gene pairs with small delta-correlations (dCorr$\leq$0.16) and those with large delta-correlations (dCorr>0.16), shown as two groups in (C) low-confounding, and (D) high-confounding settings.

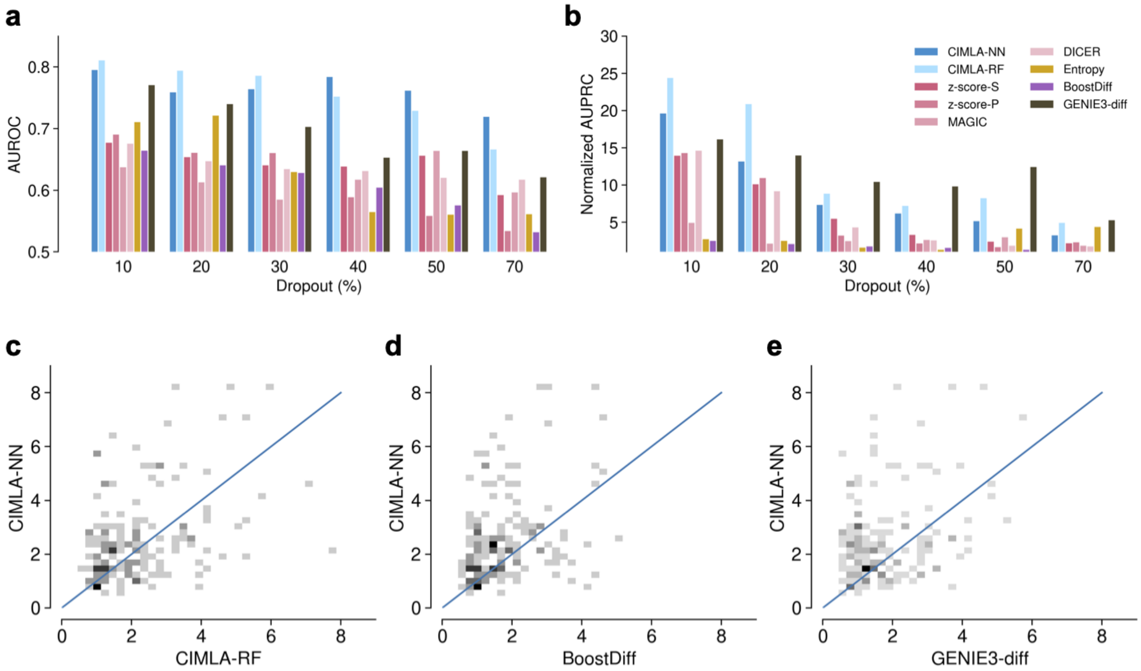

Figure C.4: (A,B) Performance of CIMLA and other methods for the "entire dGRN" prediction task on noisy simulated data at varying dropout levels in terms of (A) AUROC, and (B) Normalized AUPRC. Each bar represents the median over five simulated replicates. (C-E) Comparing the performance of different methods in terms of normalized AURPC for predicting the differential regulations of each gene in the "per-gene" prediction task. Darker colors show higher counts of genes in the 2-dimensional histogram.

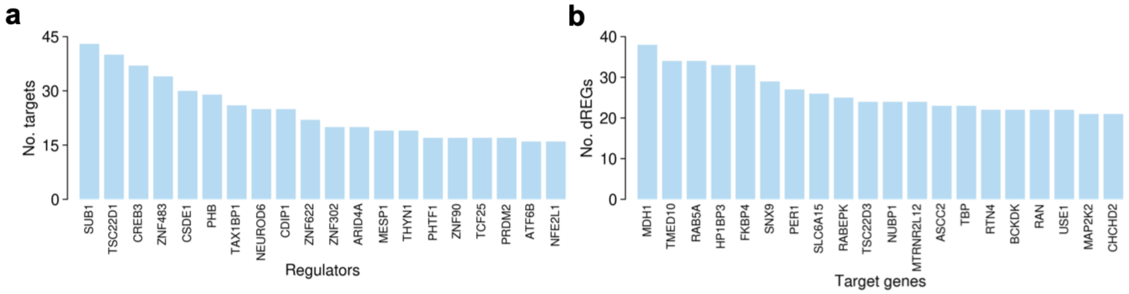

Figure C.5: (A) Top hub TFs, i.e., those targeting greatest number of DEGs, in the dGRN found by CIMLA-RF. (B) Top differentially regulated DEGs, i.e., those with greatest number of differential regulations (dREGs) found by CIMLA-RF.

C.3 SUPPLEMENTARY TABLES

Table C.1: Literature survey supporting Figure 4.5D

TF/Gene Name	**Evidence**
ZFY	One of ∼25 genes differentially expressed in superior parietal lobule of male and female AD patients [243]. The parietal cortex is important in AD [244].
ZNF433	Not known to be associated with AD, but evidence for being a susceptibility gene for Multiple Sclerosis [245], a neurodegenerative disease.
SCX	Known to be downregulated in AD and speculated to be involved in impaired neuronal plasticity and wiring [246].
ZNF154	One of 15 differentially methylated regions associated with conversion to AD [247]. One of 166 genes differentially expressed between older adults with dementia and younger adults with dementia, in a cohort of Down syndrome patients [248].
HMGB2	Plays a key role in neuroinflammation, which is associated with microglial activation and can lead to neurodegenerative diseases including AD [249]. Involved in regulation of LRP1 [250], a protein involved in pathogenesis of AD [251].
KLF16	One of 68 loci identified in GWAS study of AD [252].
FOXP4	One of nine proteins identified as AD-associated by a machine learning method and found to have higher expression in AD brains and its knockdown was shown to reduce inflammation-induced tau phosphorylation [253].
ZNF408	One of 28 genes differentially expressed between Mild Cognitive Disorder (thought to be an early form of AD in a subset of individuals) and controls, in olfactory mucosa (OM) cells [254]. (OM cells are studied in AD context because they show pathological features common with AD brains, and olfactory dysfunction is an early symptom of AD).

Table C.2: Literature survey supporting Figure 4.5E

TF/Gene Name	**Evidence**
KAT8	Involved in syndromic intellectual disability [255]. Involved in regulation of H3K16ac and AD β-amyloid generation [256]. KAT8 locus is linked to AD via multiple GWAS studies [257, 258, 259].
MTR	This is Methionine synthase, and methionine metabolism has been associated with AD [260]. Polymorphism in MTR is risk factor for AD [261, 262].
GPI	Modifier of neurodegeneration in Parkinson's models [263]. One of 10 hub genes in entorhinal cortex and hippocampus of patients with AD [264]. Related to metabolic differences in AD brains [265].
GART	Important candidate gene in Down syndrome-related AD [266].
HAGH	Associated with AD in APOE $\epsilon4$ carriers [267].
AP4M1	One of 11 candidate genes identified based on integration of AD GWAS and multi-omics analysis [268]. One of five biomarkers of AD identified based on coexpression analysis and molecular signatures [269].
CHCHD10	Mutation in gene identified in patients with AD [270, 271].
CDKN1B	Up-regulated in pre-frontal cortex of mouse models of AD [272].
ADAMTS2	Member of the ADAMTS family of ECM proteases, that are increased at early age in AD mice model [273]. Identified as potential target for AD [274].

REFERENCES

[1] A. Ay and D. N. Arnosti, “Mathematical modeling of gene expression: a guide for the perplexed biologist,” *Critical reviews in biochemistry and molecular biology*, vol. 46, no. 2, pp. 137–151, 2011. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/21417596/

[2] J. M. Dresch, M. A. Thompson, D. N. Arnosti, and C. Chiu, “Two-layer mathematical modeling of gene expression: incorporating dna-level information and system dynamics,” *SIAM Journal on Applied Mathematics*, vol. 73, no. 2, pp. 804–826, 2013. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/25328249/

[3] M. Scherer, F. Schmidt, O. Lazareva, J. Walter, J. Baumbach, M. H. Schulz, and M. List, “Machine learning for deciphering cell heterogeneity and gene regulation,” *Nature Computational Science*, vol. 1, no. 3, pp. 183–191, 2021. [Online]. Available: https://doi.org/10.1038/s43588-021-00038-7

[4] X. He, M. A. H. Samee, C. Blatti, and S. Sinha, “Thermodynamics-based models of transcriptional regulation by enhancers: the roles of synergistic activation, cooperative binding and short-range repression,” *PLoS computational biology*, vol. 6, no. 9, p. e1000935, 2010. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/20862354/

[5] M. A. Shea and G. K. Ackers, “The or control system of bacteriophage lambda: A physical-chemical model for gene regulation,” *Journal of molecular biology*, vol. 181, no. 2, pp. 211–230, 1985. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/3157005/

[6] H. Janssens, S. Hou, J. Jaeger, A.-R. Kim, E. Myasnikova, D. Sharp, and J. Reinitz, “Quantitative and predictive model of transcriptional control of the drosophila melanogaster even skipped gene,” *Nature genetics*, vol. 38, no. 10, pp. 1159–1165, 2006. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/16980977/

[7] W. D. Fakhouri, A. Ay, R. Sayal, J. Dresch, E. Dayringer, and D. N. Arnosti, “Deciphering a transcriptional regulatory code: modeling short-range repression in the drosophila embryo,” *Molecular systems biology*, vol. 6, no. 1, p. 341, 2010. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/20087339/

[8] R. P. Zinzen and D. Papatsenko, “Enhancer responses to similarly distributed antagonistic gradients in development,” *PLoS computational biology*, vol. 3, no. 5, p. e84, 2007. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/17500585/

[9] D. Mercatelli, L. Scalambra, L. Triboli, F. Ray, and F. M. Giorgi, “Gene regulatory network inference resources: A practical overview,” *Biochimica et Biophysica Acta (BBA)-Gene Regulatory Mechanisms*, vol. 1863, no. 6, p. 194430, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31678629/

[10] A. A. Margolin, I. Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, R. D. Favera, and A. Califano, "Aracne: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context," in *BMC bioinformatics*, vol. 7, no. 1. BioMed Central, 2006. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/16723010/ pp. 1–15.

[11] V. A. Huynh-Thu, A. Irrthum, L. Wehenkel, and P. Geurts, "Inferring regulatory networks from expression data using tree-based methods," *PloS one*, vol. 5, no. 9, p. e12776, 2010. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/20927193/

[12] P. Dibaeinia and S. Sinha, "Deciphering enhancer sequence using thermodynamics-based models and convolutional neural networks," *Nucleic acids research*, vol. 49, no. 18, pp. 10 309–10 327, 2021. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/34508359/

[13] F. Spitz and E. E. Furlong, "Transcription factors: from enhancer binding to developmental control," *Nature reviews genetics*, vol. 13, no. 9, pp. 613–626, 2012. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/22868264/

[14] M. Kazemian, H. Pham, S. A. Wolfe, M. H. Brodsky, and S. Sinha, "Widespread evidence of cooperative dna binding by transcription factors in drosophila development," *Nucleic acids research*, vol. 41, no. 17, pp. 8237–8252, 2013. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/23847101/

[15] O. Hobert, "Gene regulation by transcription factors and micrornas," *Science*, vol. 319, no. 5871, pp. 1785–1786, 2008. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/18369135/

[16] J.-W. Hong, D. A. Hendrix, D. Papatsenko, and M. S. Levine, "How the dorsal gradient works: insights from postgenome technologies," *Proceedings of the National Academy of Sciences*, vol. 105, no. 51, pp. 20 072–20 076, 2008. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/19104040/

[17] J. Jaeger, J. Reinitz et al., "Drosophila blastoderm patterning," *Current opinion in genetics & development*, vol. 22, no. 6, pp. 533–541, 2012. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/23290311/

[18] D. St Johnston and C. Nüsslein-Volhard, "The origin of pattern and polarity in the drosophila embryo," *Cell*, vol. 68, no. 2, pp. 201–219, 1992. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/1733499/

[19] P. Struffi, M. Corado, M. Kulkarni, and D. N. Arnosti, "Quantitative contributions of ctbp-dependent and-independent repression activities of knirps," 2004. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/15128671/

[20] Y. Nibu, K. Senger, and M. Levine, "Ctbp-independent repression in the drosophila embryo," *Molecular and cellular biology*, vol. 23, no. 11, pp. 3990–3999, 2003. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/12748300/

[21] Y. Nibu and M. S. Levine, "Ctbp-dependent activities of the short-range giant repressor in the drosophila embryo," *Proceedings of the National Academy of Sciences*, vol. 98, no. 11, pp. 6204–6208, 2001. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/11353860/

[22] V. Bhaskar and A. J. Courey, "The madf–bess domain factor dip3 potentiates synergistic activation by dorsal and twist," *Gene*, vol. 299, no. 1-2, pp. 173–184, 2002. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/12459265/

[23] P. Szymanski and M. Levine, "Multiple modes of dorsal-bhlh transcriptional synergy in the drosophila embryo." *The EMBO Journal*, vol. 14, no. 10, pp. 2229–2238, 1995. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/7774581/

[24] D. M. King, C. K. Y. Hong, J. L. Shepherdson, D. M. Granas, B. B. Maricque, and B. A. Cohen, "Synthetic and genomic regulatory elements reveal aspects of cis-regulatory grammar in mouse embryonic stem cells," *Elife*, vol. 9, p. e41279, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/32043966/

[25] M. M. Kulkarni and D. N. Arnosti, "cis-regulatory logic of short-range transcriptional repression in drosophila melanogaster," *Molecular and cellular biology*, vol. 25, no. 9, pp. 3411–3420, 2005. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/15831448/

[26] R. Sayal, J. M. Dresch, I. Pushel, B. R. Taylor, and D. N. Arnosti, "Quantitative perturbation-based analysis of gene expression predicts enhancer activity in early drosophila embryo," *Elife*, vol. 5, 2016. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/27152947/

[27] M. A. White, J. C. Kwasnieski, C. A. Myers, S. Q. Shen, J. C. Corbo, and B. A. Cohen, "A simple grammar defines activating and repressing cis-regulatory elements in photoreceptors," *Cell reports*, vol. 17, no. 5, pp. 1247–1254, 2016. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/27783940/

[28] B. Deplancke, D. Alpern, and V. Gardeux, "The genetics of transcription factor dna binding variation," *Cell*, vol. 166, no. 3, pp. 538–554, 2016. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/27471964/

[29] J. M. Vahrenkamp, C.-H. Yang, A. C. Rodriguez, A. Almomen, K. C. Berrett, A. N. Trujillo, K. P. Guillen, B. E. Welm, E. A. Jarboe, M. M. Janat-Amsbury et al., "Clinical and genomic crosstalk between glucocorticoid receptor and estrogen receptor α in endometrial cancer," *Cell reports*, vol. 22, no. 11, pp. 2995–3005, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29539426/

[30] E. K. Farley, K. M. Olson, W. Zhang, D. S. Rokhsar, and M. S. Levine, "Syntax compensates for poor binding sites to encode tissue specificity of developmental enhancers," *Proceedings of the National Academy of Sciences*, vol. 113, no. 23, pp. 6508–6513, 2016. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/27155014/

[31] G. R. Ilsley, J. Fisher, R. Apweiler, A. H. DePace, and N. M. Luscombe, “Cellular resolution models for even skipped regulation in the entire drosophila embryo,” *Elife*, vol. 2, p. e00522, 2013. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/23930223/

[32] J. Crocker, G. R. Ilsley, and D. L. Stern, “Quantitatively predictable control of drosophila transcriptional enhancers in vivo with engineered transcription factors,” *Nature genetics*, vol. 48, no. 3, pp. 292–298, 2016. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/26854918/

[33] E. Segal, T. Raveh-Sadka, M. Schroeder, U. Unnerstall, and U. Gaul, “Predicting expression patterns from regulatory sequence in drosophila segmentation,” *Nature*, vol. 451, no. 7178, pp. 535–540, 2008. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/18172436/

[34] J. Reinitz, S. Hou, and D. H. Sharp, “Transcriptional control in drosophila,” *ComPlexUs*, vol. 1, no. 2, pp. 54–64, 2003. [Online]. Available: https://www.karger.com/Article/Abstract/70462

[35] M. A. H. Samee, B. Lim, N. Samper, H. Lu, C. A. Rushlow, G. Jiménez, S. Y. Shvartsman, and S. Sinha, “A systematic ensemble approach to thermodynamic modeling of gene expression from sequence data,” *Cell systems*, vol. 1, no. 6, pp. 396–407, 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/27136354/

[36] R. Grah, B. Zoller, and G. Tkačik, “Nonequilibrium models of optimal enhancer function,” *Proceedings of the National Academy of Sciences*, vol. 117, no. 50, pp. 31 614–31 622, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/33268497/

[37] T. Ahsendorf, F. Wong, R. Eils, and J. Gunawardena, “A framework for modelling gene regulation which accommodates non-equilibrium mechanisms,” *BMC biology*, vol. 12, no. 1, pp. 1–23, 2014. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/25475875/

[38] J. Estrada, F. Wong, A. DePace, and J. Gunawardena, “Information integration and energy expenditure in gene regulation,” *Cell*, vol. 166, no. 1, pp. 234–244, 2016. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/27368104/

[39] J. Gertz, E. D. Siggia, and B. A. Cohen, “Analysis of combinatorial cis-regulation in synthetic and genomic promoters,” *Nature*, vol. 457, no. 7226, pp. 215–218, 2009. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/19029883/

[40] M. A. Beer and S. Tavazoie, “Predicting gene expression from sequence,” *Cell*, vol. 117, no. 2, pp. 185–198, 2004. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/15084257/

[41] C. Blatti, M. Kazemian, S. Wolfe, M. Brodsky, and S. Sinha, “Integrating motif, dna accessibility and gene expression data to build regulatory maps in an organism,” *Nucleic acids research*, vol. 43, no. 8, pp. 3998–4012, 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/25791631/

[42] Ž. Avsec, M. Weilert, A. Shrikumar, S. Krueger, A. Alexandari, K. Dalal, R. Fropf, C. McAnany, J. Gagneur, A. Kundaje et al., "Base-resolution models of transcription-factor binding reveal soft motif syntax," *Nature Genetics*, vol. 53, no. 3, pp. 354–366, 2021. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/33603233/

[43] D. R. Kelley, J. Snoek, and J. L. Rinn, "Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks," *Genome research*, vol. 26, no. 7, pp. 990–999, 2016. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/27197224/

[44] B. Alipanahi, A. Delong, M. T. Weirauch, and B. J. Frey, "Predicting the sequence specificities of dna-and rna-binding proteins by deep learning," *Nature biotechnology*, vol. 33, no. 8, pp. 831–838, 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/26213851/

[45] J. Zhou and O. G. Troyanskaya, "Predicting effects of noncoding variants with deep learning–based sequence model," *Nature methods*, vol. 12, no. 10, pp. 931–934, 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/26301843/

[46] T. Duque, M. A. H. Samee, M. Kazemian, H. N. Pham, M. H. Brodsky, and S. Sinha, "Simulations of enhancer evolution provide mechanistic insights into gene regulation," *Molecular biology and evolution*, vol. 31, no. 1, pp. 184–200, 2014. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/24097306/

[47] J. M. Shirokawa and A. J. Courey, "A direct contact between the dorsal rel homology domain and twist may mediate transcriptional synergy," *Molecular and cellular biology*, vol. 17, no. 6, pp. 3345–3355, 1997. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/9154833/

[48] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014. [Online]. Available: https://jmlr.org/papers/v15/srivastava14a.html

[49] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016. [Online]. Available: https://arxiv.org/abs/1607.06450

[50] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. [Online]. Available: https://arxiv.org/abs/1412.6980

[51] M. Kazemian, C. Blatti, A. Richards, M. McCutchan, N. Wakabayashi-Ito, A. S. Hammonds, S. E. Celniker, S. Kumar, S. A. Wolfe, M. H. Brodsky et al., "Quantitative analysis of the drosophila segmentation regulatory network using pattern generating potentials," *PLoS biology*, vol. 8, no. 8, p. e1000456, 2010. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/20808951/

[52] M. A. H. Samee, T. Lydiard-Martin, K. M. Biette, B. J. Vincent, M. D. Bragdon, K. B. Eckenrode, Z. Wunderlich, J. Estrada, S. Sinha, and A. H. DePace, “Quantitative measurement and thermodynamic modeling of fused enhancers support a two-tiered mechanism for interpreting regulatory dna,” *Cell reports*, vol. 21, no. 1, pp. 236–245, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28978476/

[53] Y. Nibu, H. Zhang, and M. Levine, “Interaction of short-range repressors with drosophila ctbp in the embryo,” *Science*, vol. 280, no. 5360, pp. 101–104, 1998. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/9525852/

[54] Y. Nibu, H. Zhang, E. Bajor, S. Barolo, S. Small, and M. Levine, “dctbp mediates transcriptional repression by knirps, krüppel and snail in the drosophila embryo,” *The EMBO journal*, vol. 17, no. 23, pp. 7009–7020, 1998. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/9843507/

[55] G. Chinnadurai, “Ctbp, an unconventional transcriptional corepressor in development and oncogenesis,” *Molecular cell*, vol. 9, no. 2, pp. 213–224, 2002. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/11864595/

[56] P. Struffi and D. N. Arnosti, “Functional interaction between the drosophila knirps short range transcriptional repressor and rpd3 histone deacetylase,” *Journal of Biological Chemistry*, vol. 280, no. 49, pp. 40 757–40 765, 2005. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/16186109/

[57] C. I. Swanson, N. C. Evans, and S. Barolo, “Structural rules and complex regulatory circuitry constrain expression of a notch-and egfr-regulated eye enhancer,” *Developmental cell*, vol. 18, no. 3, pp. 359–370, 2010. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/20230745/

[58] J. Crocker, Y. Tamori, and A. Erives, “Evolution acts on enhancer organization to fine-tune gradient threshold readouts,” *PLoS biology*, vol. 6, no. 11, p. e263, 2008. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/18986212/

[59] C. D. Arnold, D. Gerlach, C. Stelzer, Ł. M. Boryń, M. Rath, and A. Stark, “Genome-wide quantitative enhancer activity maps identified by starr-seq,” *Science*, vol. 339, no. 6123, pp. 1074–1077, 2013. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/23328393/

[60] B. B. Maricque, J. D. Dougherty, and B. A. Cohen, “A genome-integrated massively parallel reporter assay reveals dna sequence determinants of cis-regulatory activity in neural cells,” *Nucleic acids research*, vol. 45, no. 4, pp. e16–e16, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28204611/

[61] A. Melnikov, X. Zhang, P. Rogov, L. Wang, and T. S. Mikkelsen, “Massively parallel reporter assays in cultured mammalian cells,” *JoVE (Journal of Visualized Experiments)*, no. 90, p. e51719, 2014. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/25177895/

[62] J. Gertz, E. D. Siggia, and B. A. Cohen, “Analysis of combinatorial cis-regulation in synthetic and genomic promoters,” *Nature*, vol. 457, no. 7226, pp. 215–218, 2009. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/19029883/

[63] D. Papatsenko and M. S. Levine, “Dual regulation by the hunchback gradient in the drosophila embryo,” *Proceedings of the National Academy of Sciences*, vol. 105, no. 8, pp. 2901–2906, 2008. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/18287046/

[64] A.-R. Kim, C. Martinez, J. Ionides, A. F. Ramos, M. Z. Ludwig, N. Ogawa, D. H. Sharp, and J. Reinitz, “Rearrangements of 2.5 kilobases of noncoding dna from the drosophila even-skipped locus define predictive rules of genomic cis-regulatory logic,” *PLoS genetics*, vol. 9, no. 2, p. e1003243, 2013. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/23468638/

[65] S. Gray, P. Szymanski, and M. Levine, “Short-range repression permits multiple enhancers to function autonomously within a complex promoter.” *Genes & development*, vol. 8, no. 15, pp. 1829–1838, 1994. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/7958860/

[66] A. J. Courey and S. Jia, “Transcriptional repression: the long and the short of it,” *Genes & development*, vol. 15, no. 21, pp. 2786–2796, 2001. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/11691830/

[67] B. Settles, “Active learning. synth lect artif intell mach learn 6: 1–114,” 2012. [Online]. Available: https://link.springer.com/book/10.1007/978-3-031-01560-1

[68] F. Khajouei and S. Sinha, “An information theoretic treatment of sequence-to-expression modeling,” *PLoS computational biology*, vol. 14, no. 9, p. e1006459, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30256780/

[69] A. Lal, Z. D. Chiang, N. Yakovenko, F. M. Duarte, J. Israeli, and J. D. Buenrostro, “Deep learning-based enhancement of epigenomics data with atacworks,” *Nature communications*, vol. 12, no. 1, pp. 1–11, 2021. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/33686069/

[70] C. Angermueller, H. J. Lee, W. Reik, and O. Stegle, “Deepcpg: accurate prediction of single-cell dna methylation states using deep learning,” *Genome biology*, vol. 18, no. 1, pp. 1–13, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28395661/

[71] V. Agarwal and J. Shendure, “Predicting mrna abundance directly from genomic sequence using deep convolutional neural networks,” *Cell reports*, vol. 31, no. 7, p. 107663, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/32433972/

[72] M. T. Weirauch, A. Yang, M. Albu, A. G. Cote, A. Montenegro-Montero, P. Drewe, H. S. Najafabadi, S. A. Lambert, I. Mann, K. Cook et al., “Determination and inference of eukaryotic transcription factor sequence specificity,” *Cell*, vol. 158, no. 6, pp. 1431–1443, 2014. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/25215497/

[73] Y. Liu, K. Barr, and J. Reinitz, “Fully interpretable deep learning model of transcriptional control,” *Bioinformatics*, vol. 36, no. Supplement_1, pp. i499–i507, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/32657418/

[74] F. Khajouei, N. Samper, N. Djabrayan, B. Lunt, G. Jiménez, and S. Sinha, “Model-based analysis of polymorphisms in an enhancer reveals cis-regulatory mechanisms,” *bioRxiv*, 2020. [Online]. Available: https://doi.org/10.1101/2020.02.07.939264

[75] S. Tabe-Bordbar, Y. J. Song, B. J. Lunt, K. V. Prasanth, and S. Sinha, “Mechanistic analysis of enhancer sequences in the estrogen receptor transcriptional program,” *bioRxiv*, 2020. [Online]. Available: https://doi.org/10.1101/2020.11.08.373555

[76] P. Dibaeinia and S. Sinha, “Sergio: a single-cell expression simulator guided by gene regulatory networks,” *Cell systems*, vol. 11, no. 3, pp. 252–271, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/32871105/

[77] E. Hedlund and Q. Deng, “Single-cell rna sequencing: technical advancements and biological applications,” *Molecular aspects of medicine*, vol. 59, pp. 36–46, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28754496/

[78] G. Kelsey, O. Stegle, and W. Reik, “Single-cell epigenomics: Recording the past and predicting the future,” *Science*, vol. 358, no. 6359, pp. 69–75, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28983045/

[79] E. Papalexi and R. Satija, “Single-cell rna sequencing to explore immune cell heterogeneity,” *Nature Reviews Immunology*, vol. 18, no. 1, pp. 35–45, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28787399/

[80] J. Park, R. Shrestha, C. Qiu, A. Kondo, S. Huang, M. Werth, M. Li, J. Barasch, and K. Suszták, “Single-cell transcriptomics of the mouse kidney reveals potential cellular targets of kidney disease,” *Science*, vol. 360, no. 6390, pp. 758–763, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29622724/

[81] F. Buettner, K. N. Natarajan, F. P. Casale, V. Proserpio, A. Scialdone, F. J. Theis, S. A. Teichmann, J. C. Marioni, and O. Stegle, “Computational analysis of cell-to-cell heterogeneity in single-cell rna-sequencing data reveals hidden subpopulations of cells,” *Nature biotechnology*, vol. 33, no. 2, pp. 155–160, 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/25599176/

[82] A. Butler, P. Hoffman, P. Smibert, E. Papalexi, and R. Satija, “Integrating single-cell transcriptomic data across different conditions, technologies, and species,” *Nature biotechnology*, vol. 36, no. 5, pp. 411–420, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29608179/

[83] O. Stegle, S. A. Teichmann, and J. C. Marioni, “Computational and analytical challenges in single-cell transcriptomics,” *Nature Reviews Genetics*, vol. 16, no. 3, pp. 133–145, 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/25628217/

[84] F. A. Wolf, P. Angerer, and F. J. Theis, "Scanpy: large-scale single-cell gene expression data analysis," *Genome biology*, vol. 19, no. 1, pp. 1–5, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29409532/

[85] S. Aibar, C. B. González-Blas, T. Moerman, V. A. Huynh-Thu, H. Imrichova, G. Hulselmans, F. Rambow, J.-C. Marine, P. Geurts, J. Aerts et al., "Scenic: single-cell regulatory network inference and clustering," *Nature methods*, vol. 14, no. 11, pp. 1083–1086, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28991892/

[86] V. Y. Kiselev, K. Kirschner, M. T. Schaub, T. Andrews, A. Yiu, T. Chandra, K. N. Natarajan, W. Reik, M. Barahona, A. R. Green et al., "Sc3: consensus clustering of single-cell rna-seq data," *Nature methods*, vol. 14, no. 5, pp. 483–486, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28346451/

[87] R. Satija, J. A. Farrell, D. Gennert, A. F. Schier, and A. Regev, "Spatial reconstruction of single-cell gene expression data," *Nature biotechnology*, vol. 33, no. 5, pp. 495–502, 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/25867923/

[88] C. A. Herring, A. Banerjee, E. T. McKinley, A. J. Simmons, J. Ping, J. T. Roland, J. L. Franklin, Q. Liu, M. J. Gerdes, R. J. Coffey et al., "Unsupervised trajectory analysis of single-cell rna-seq and imaging data reveals alternative tuft cell origins in the gut," *Cell systems*, vol. 6, no. 1, pp. 37–51, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29153838/

[89] K. Street, D. Risso, R. B. Fletcher, D. Das, J. Ngai, N. Yosef, E. Purdom, and S. Dudoit, "Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics," *BMC genomics*, vol. 19, no. 1, pp. 1–16, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29914354/

[90] T. E. Chan, M. P. Stumpf, and A. C. Babtie, "Gene regulatory network inference from single-cell data using multivariate information measures," *Cell systems*, vol. 5, no. 3, pp. 251–267, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28957658/

[91] S. Mohammadi, V. Ravindra, D. F. Gleich, and A. Grama, "A geometric approach to characterize the functional identity of single cells," *Nature communications*, vol. 9, no. 1, pp. 1–10, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29666373/

[92] D. Van Dijk, R. Sharma, J. Nainys, K. Yim, P. Kathail, A. J. Carr, C. Burdziak, K. R. Moon, C. L. Chaffer, D. Pattabiraman et al., "Recovering gene interactions from single-cell data using data diffusion," *Cell*, vol. 174, no. 3, pp. 716–729, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29961576/

[93] G. Eraslan, L. M. Simon, M. Mircea, N. S. Mueller, and F. J. Theis, "Single-cell rna-seq denoising using a deep count autoencoder," *Nature communications*, vol. 10, no. 1, pp. 1–14, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30674886/

[94] W. V. Li and J. J. Li, "An accurate and robust imputation method scimpute for single-cell rna-seq data," *Nature communications*, vol. 9, no. 1, pp. 1–9, 2018.

[95] R. Hou, E. Denisenko, and A. R. Forrest, "scmatch: a single-cell gene expression profile annotation tool using reference datasets," *Bioinformatics*, vol. 35, no. 22, pp. 4688–4695, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31028376/

[96] T. M. Consortium et al., "Single-cell transcriptomics of 20 mouse organs creates a tabula muris," *Nature*, vol. 562, no. 7727, pp. 367–372, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30283141/

[97] K. Van den Berge, F. Perraudeau, C. Soneson, M. I. Love, D. Risso, J.-P. Vert, M. D. Robinson, S. Dudoit, and L. Clement, "Observation weights unlock bulk rna-seq tools for zero inflation and single-cell applications," *Genome biology*, vol. 19, no. 1, pp. 1–17, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29478411/

[98] K. R. Campbell and C. Yau, "Uncovering pseudotemporal trajectories with covariates from single cell and bulk expression data," *Nature communications*, vol. 9, no. 1, pp. 1–12, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29934517/

[99] C. Chen, C. Wu, L. Wu, X. Wang, M. Deng, and R. Xi, "scrmd: imputation for single cell rna-seq data via robust matrix decomposition," *Bioinformatics*, vol. 36, no. 10, pp. 3156–3161, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/32119079/

[100] W. Gong, B. N. Singh, P. Shah, S. Das, J. Theisen, S. Chan, M. Kyba, M. G. Garry, D. Yannopoulos, W. Pan et al., "A novel algorithm for the collective integration of single cell rna-seq during embryogenesis," *BioRxiv*, p. 543314, 2019. [Online]. Available: https://doi.org/10.1101/543314

[101] K. D. Korthauer, L.-F. Chu, M. A. Newton, Y. Li, J. Thomson, R. Stewart, and C. Kendziorski, "A statistical approach for identifying differential distributions in single-cell rna-seq experiments," *Genome biology*, vol. 17, no. 1, pp. 1–15, 2016. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/27782827/

[102] D. Risso, F. Perraudeau, S. Gribkova, S. Dudoit, and J.-P. Vert, "A general and flexible method for signal extraction from single-cell rna-seq data," *Nature communications*, vol. 9, no. 1, pp. 1–17, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29348443/

[103] L. Holm, "Benchmarking fold detection by dalilite v. 5," *Bioinformatics*, vol. 35, no. 24, pp. 5326–5327, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31263867/

[104] M. Marouf, P. Machart, V. Bansal, C. Kilian, D. S. Magruder, C. F. Krebs, and S. Bonn, "Realistic in silico generation and augmentation of single-cell rna-seq data using generative adversarial networks," *Nature communications*, vol. 11, no. 1, pp. 1–12, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31919373/

[105] N. Papadopoulos, P. R. Gonzalo, and J. Söding, “Prosstt: probabilistic simulation of single-cell rna-seq data for complex differentiation processes,” *Bioinformatics*, vol. 35, no. 18, pp. 3517–3519, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30715210/

[106] B. Vieth, C. Ziegenhain, S. Parekh, W. Enard, and I. Hellmann, “powsimr: power analysis for bulk and single cell rna-seq experiments,” *Bioinformatics*, vol. 33, no. 21, pp. 3486–3488, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29036287/

[107] L. Zappia, B. Phipson, and A. Oshlack, “Splatter: simulation of single-cell rna sequencing data,” *Genome biology*, vol. 18, no. 1, pp. 1–15, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28899397/

[108] X. Zhang, C. Xu, and N. Yosef, “Simulating multiple faceted variability in single cell rna sequencing,” *Nature communications*, vol. 10, no. 1, pp. 1–16, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31197158/

[109] J. Intosalmi, H. Mannerström, S. Hiltunen, and H. Lähdesmäki, “Schirm: Single cell hierarchical regression model to detect dependencies in read count data,” *BioRxiv*, p. 335695, 2018. [Online]. Available: https://doi.org/10.1101/335695

[110] G. Karlebach and R. Shamir, “Modelling and analysis of gene regulatory networks,” *Nature reviews Molecular cell biology*, vol. 9, no. 10, pp. 770–780, 2008. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/18797474/

[111] T. B. Kepler and T. C. Elston, “Stochasticity in transcriptional regulation: origins, consequences, and mathematical representations,” *Biophysical journal*, vol. 81, no. 6, pp. 3116–3136, 2001. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/11720979/

[112] H. El Samad, M. Khammash, L. Petzold, and D. Gillespie, “Stochastic modelling of gene regulatory networks,” *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, vol. 15, no. 15, pp. 691–711, 2005. [Online]. Available: https://doi.org/10.1002/rnc.1018

[113] D. J. Wilkinson, “Stochastic modelling for quantitative description of heterogeneous biological systems,” *Nature Reviews Genetics*, vol. 10, no. 2, pp. 122–133, 2009. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/19139763/

[114] T. Schaffter, D. Marbach, and D. Floreano, “Genenetweaver: in silico benchmark generation and performance profiling of network inference methods,” *Bioinformatics*, vol. 27, no. 16, pp. 2263–2270, 2011. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/21697125/

[115] P. Bellot, C. Olsen, P. Salembier, A. Oliveras-Vergés, and P. E. Meyer, “Netbenchmark: a bioconductor package for reproducible benchmarks of gene regulatory network inference,” *BMC bioinformatics*, vol. 16, no. 1, pp. 1–15, 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/26415849/

[116] D. Marbach, R. J. Prill, T. Schaffter, C. Mattiussi, D. Floreano, and G. Stolovitzky, "Revealing strengths and weaknesses of methods for gene network inference," *Proceedings of the national academy of sciences*, vol. 107, no. 14, pp. 6286–6291, 2010. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/20308593/

[117] W. Saelens, R. Cannoodt, and Y. Saeys, "A comprehensive evaluation of module detection methods for gene expression data," *Nature communications*, vol. 9, no. 1, pp. 1–12, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29545622/

[118] C. Siegenthaler and R. Gunawan, "Assessment of network inference methods: how to cope with an underdetermined problem," *PloS one*, vol. 9, no. 3, p. e90481, 2014. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/24603847/

[119] S. Chen and J. C. Mar, "Evaluating methods of inferring gene regulatory networks highlights their lack of performance for single cell gene expression data," *BMC bioinformatics*, vol. 19, no. 1, pp. 1–21, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29914350/

[120] D. T. Gillespie, "The chemical langevin equation," *The Journal of Chemical Physics*, vol. 113, no. 1, pp. 297–306, 2000. [Online]. Available: https://doi.org/10.1063/1.481811

[121] D. T. Gillespie, "A rigorous derivation of the chemical master equation," *Physica A: Statistical Mechanics and its Applications*, vol. 188, no. 1-3, pp. 404–425, 1992. [Online]. Available: https://doi.org/10.1016/0378-4371(92)90283-V

[122] R. Khanin and D. J. Higham, "Chemical master equation and langevin regimes for a gene transcription model," *Theoretical Computer Science*, vol. 408, no. 1, pp. 31–40, 2008. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-540-75140-3_1

[123] D. T. Gillespie, "A general method for numerically simulating the stochastic time evolution of coupled chemical reactions," *Journal of computational physics*, vol. 22, no. 4, pp. 403–434, 1976. [Online]. Available: https://doi.org/10.1016/0021-9991(76)90041-3

[124] D. T. Gillespie, "Exact stochastic simulation of coupled chemical reactions," *The journal of physical chemistry*, vol. 81, no. 25, pp. 2340–2361, 1977. [Online]. Available: https://doi.org/10.1021/j100540a008

[125] T. Schaffter, "Numerical integration of sdes: a short tutorial," Tech. Rep., 2010. [Online]. Available: https://infoscience.epfl.ch/record/143450?ln=en

[126] D. Chu, N. R. Zabet, and B. Mitavskiy, "Models of transcription factor binding: sensitivity of activation functions to model assumptions," *Journal of Theoretical Biology*, vol. 257, no. 3, pp. 419–429, 2009. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/19121637/

[127] R. J. Prill, R. Vogel, G. A. Cecchi, G. Altan-Bonnet, and G. Stolovitzky, "Noise-driven causal inference in biomolecular networks," *PloS one*, vol. 10, no. 6, p. e0125777, 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/26030907/

[128] M. A. Basson, "Signaling in cell differentiation and morphogenesis," *Cold Spring Harbor perspectives in biology*, vol. 4, no. 6, p. a008151, 2012. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/22570373/

[129] G. Balázsi, A. Van Oudenaarden, and J. J. Collins, "Cellular decision making and biological noise: from microbes to mammals," *Cell*, vol. 144, no. 6, pp. 910–925, 2011. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/21414483/

[130] G. La Manno, R. Soldatov, A. Zeisel, E. Braun, H. Hochgerner, V. Petukhov, K. Lidschreiber, M. E. Kastriti, P. Lönnerberg, A. Furlan et al., "Rna velocity of single cells," *Nature*, vol. 560, no. 7719, pp. 494–498, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30089906/

[131] V. Svensson and L. Pachter, "Rna velocity: molecular kinetics from single-cell rna-seq," *Molecular cell*, vol. 72, no. 1, pp. 7–9, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30290149/

[132] A. Zeisel, A. B. Muñoz-Manchado, S. Codeluppi, P. Lönnerberg, G. La Manno, A. Juréus, S. Marques, H. Munguba, L. He, C. Betsholtz et al., "Cell types in the mouse cortex and hippocampus revealed by single-cell rna-seq," *Science*, vol. 347, no. 6226, pp. 1138–1142, 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/25700174/

[133] C. Mayer, C. Hafemeister, R. C. Bandler, R. Machold, R. Batista Brito, X. Jaglin, K. Allaway, A. Butler, G. Fishell, and R. Satija, "Developmental diversification of cortical inhibitory interneurons," *Nature*, vol. 555, no. 7697, pp. 457–462, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29513653/

[134] N. O. Lindström, J. Guo, A. D. Kim, T. Tran, Q. Guo, G. D. S. Brandine, A. Ransick, R. K. Parvez, M. E. Thornton, L. Basking et al., "Conserved and divergent features of mesenchymal progenitor cell types within the cortical nephrogenic niche of the human and mouse kidney," *Journal of the American Society of Nephrology*, vol. 29, no. 3, pp. 806–824, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29449449/

[135] K. Paulson, V. Voillet, M. McAfee, D. Hunter, F. Wagener, M. Perdicchio, W. Valente, S. Koelle, C. Church, N. Vandeven et al., "Acquired cancer resistance to combination immunotherapy from transcriptional loss of class i hla," *Nature communications*, vol. 9, no. 1, pp. 1–10, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30250229/

[136] F. A. Vieira Braga, G. Kar, M. Berg, O. A. Carpaij, K. Polanski, L. M. Simon, S. Brouwer, T. Gomes, L. Hesse, J. Jiang et al., "A cellular census of human lungs identifies novel cell states in health and in asthma," *Nature medicine*, vol. 25, no. 7, pp. 1153–1163, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31209336/

[137] H. Hochgerner, A. Zeisel, P. Lönnerberg, and S. Linnarsson, "Conserved properties of dentate gyrus neurogenesis across postnatal development revealed by single-cell rna sequencing," *Nature neuroscience*, vol. 21, no. 2, pp. 290–299, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29335606/

[138] A. Pratapa, A. P. Jalihal, J. N. Law, A. Bharadwaj, and T. Murali, "Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data," *Nature methods*, vol. 17, no. 2, pp. 147–154, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31907445/

[139] H. Matsumoto, H. Kiryu, C. Furusawa, M. S. Ko, S. B. Ko, N. Gouda, T. Hayashi, and I. Nikaido, "Scode: an efficient regulatory network inference algorithm from single-cell rna-seq during differentiation," *Bioinformatics*, vol. 33, no. 15, pp. 2314–2321, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28379368/

[140] N. Papili Gao, S. M. Ud-Dean, O. Gandrillon, and R. Gunawan, "Sincerities: inferring gene regulatory networks from time-stamped single cell transcriptional expression profiles," *Bioinformatics*, vol. 34, no. 2, pp. 258–266, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28968704/

[141] A. Deshpande, L.-F. Chu, R. Stewart, and A. Gitter, "Network inference with granger causality ensembles on single-cell transcriptomics," *Cell reports*, vol. 38, no. 6, p. 110333, 2022. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/35139376/

[142] A. A. Kolodziejczyk, J. K. Kim, V. Svensson, J. C. Marioni, and S. A. Teichmann, "The technology and biology of single-cell rna sequencing," *Molecular cell*, vol. 58, no. 4, pp. 610–620, 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/26000846/

[143] T. S. Andrews and M. Hemberg, "M3drop: dropout-based feature selection for scrnaseq," *Bioinformatics*, vol. 35, no. 16, pp. 2865–2867, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30590489/

[144] E. Pierson and C. Yau, "Zifa: Dimensionality reduction for zero-inflated single-cell gene expression analysis," *Genome biology*, vol. 16, no. 1, pp. 1–10, 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/26527291/

[145] R. D. Dar, S. M. Shaffer, A. Singh, B. S. Razooky, M. L. Simpson, A. Raj, and L. S. Weinberger, "Transcriptional bursting explains the noise–versus–mean relationship in mrna and protein levels," *PloS one*, vol. 11, no. 7, p. e0158298, 2016. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/27467384/

[146] K. Franz, A. Singh, and L. S. Weinberger, "Lentiviral vectors to study stochastic noise in gene expression," in *Methods in enzymology*. Elsevier, 2011, vol. 497, pp. 603–622. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/21601105/

[147] D. J. McCarthy, Y. Chen, and G. K. Smyth, "Differential expression analysis of multifactor rna-seq experiments with respect to biological variation," *Nucleic acids research*, vol. 40, no. 10, pp. 4288–4297, 2012. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/22287627/

[148] Z.-P. Liu, C. Wu, H. Miao, and H. Wu, "Regnetwork: an integrated database of transcriptional and post-transcriptional regulatory networks in human and mouse," *Database*, vol. 2015, 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/26424082/

[149] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008. [Online]. Available: https://www.jmlr.org/papers/v9/vandermaaten08a.html

[150] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018. [Online]. Available: https://doi.org/10.48550/arXiv.1802.03426

[151] E. Becht, L. McInnes, J. Healy, C.-A. Dutertre, I. W. Kwok, L. G. Ng, F. Ginhoux, and E. W. Newell, "Dimensionality reduction for visualizing single-cell data using umap," *Nature biotechnology*, vol. 37, no. 1, pp. 38–44, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30531897/

[152] P. S. Swain, M. B. Elowitz, and E. D. Siggia, "Intrinsic and extrinsic contributions to stochasticity in gene expression," *Proceedings of the National Academy of Sciences*, vol. 99, no. 20, pp. 12 795–12 800, 2002. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/12237400/

[153] A. F. Siahpirani and S. Roy, "A prior-based integrative framework for functional transcriptional regulatory network inference," *Nucleic acids research*, vol. 45, no. 4, pp. e21–e21, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/27794550/

[154] T. Peng, Q. Zhu, P. Yin, and K. Tan, "Scrabble: single-cell rna-seq imputation constrained by bulk rna-seq data," *Genome biology*, vol. 20, no. 1, pp. 1–12, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31060596/

[155] L. Zhang and S. Zhang, "Comparison of computational methods for imputing single-cell rna-sequencing data," *IEEE/ACM transactions on computational biology and bioinformatics*, vol. 17, no. 2, pp. 376–389, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29994128/

[156] W. Zhou, M. A. Yui, B. A. Williams, J. Yun, B. J. Wold, L. Cai, and E. V. Rothenberg, "Single-cell analysis reveals regulatory gene expression dynamics leading to lineage commitment in early t cell development," *Cell systems*, vol. 9, no. 4, pp. 321–337, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31629685/

[157] W. J. Longabaugh, W. Zeng, J. A. Zhang, H. Hosokawa, C. S. Jansen, L. Li, M. Romero-Wolf, P. Liu, H. Y. Kueh, A. Mortazavi et al., "Bcl11b and combinatorial resolution of cell fate in the t-cell gene regulatory network," *Proceedings of the National Academy of Sciences*, vol. 114, no. 23, pp. 5800–5807, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28584128/

[158] C. Trapnell, D. Cacchiarelli, J. Grimsby, P. Pokharel, S. Li, M. Morse, N. J. Lennon, K. J. Livak, T. S. Mikkelsen, and J. L. Rinn, “The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells,” *Nature biotechnology*, vol. 32, no. 4, pp. 381–386, 2014. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/24658644/

[159] P. Langfelder and S. Horvath, “Wgcna: an r package for weighted correlation network analysis,” *BMC bioinformatics*, vol. 9, no. 1, pp. 1–13, 2008. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/19114008/

[160] E. Uffelmann, Q. Q. Huang, N. S. Munung, J. De Vries, Y. Okada, A. R. Martin, H. C. Martin, T. Lappalainen, and D. Posthuma, “Genome-wide association studies,” *Nature Reviews Methods Primers*, vol. 1, no. 1, pp. 1–21, 2021. [Online]. Available: https://www.nature.com/articles/s43586-021-00056-9

[161] A. C. Nica and E. T. Dermitzakis, “Expression quantitative trait loci: present and future,” *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 368, no. 1620, p. 20120362, 2013. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/23650636/

[162] A. C. Cote, H. E. Young, and L. M. Huckins, “Comparison of confound adjustment methods in the construction of gene co-expression networks,” *Genome Biology*, vol. 23, no. 1, pp. 1–13, 2022. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/35115012/

[163] C. Angermueller, H. J. Lee, W. Reik, and O. Stegle, “Deepcpg: accurate prediction of single-cell dna methylation states using deep learning,” *Genome biology*, vol. 18, no. 1, pp. 1–13, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28395661/

[164] L. Sundaram, H. Gao, S. R. Padigepati, J. F. McRae, Y. Li, J. A. Kosmicki, N. Fritzilas, J. Hakenberg, A. Dutta, J. Shon et al., “Predicting the clinical impact of human mutation with deep neural networks,” *Nature genetics*, vol. 50, no. 8, pp. 1161–1170, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30038395/

[165] Ž. Avsec, M. Weilert, A. Shrikumar, S. Krueger, A. Alexandari, K. Dalal, R. Fropf, C. McAnany, J. Gagneur, A. Kundaje et al., “Base-resolution models of transcription-factor binding reveal soft motif syntax,” *Nature Genetics*, vol. 53, no. 3, pp. 354–366, 2021. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/33603233/

[166] Ž. Avsec, V. Agarwal, D. Visentin, J. R. Ledsam, A. Grabska-Barwinska, K. R. Taylor, Y. Assael, J. Jumper, P. Kohli, and D. R. Kelley, “Effective gene expression prediction from sequence by integrating long-range interactions,” *Nature methods*, vol. 18, no. 10, pp. 1196–1203, 2021. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/34608324/

[167] K. M. Chen, A. K. Wong, O. G. Troyanskaya, and J. Zhou, "A sequence-based global map of regulatory activity for deciphering human genetics," *Nature genetics*, vol. 54, no. 7, pp. 940–949, 2022. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/35817977/

[168] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee, "From local explanations to global understanding with explainable ai for trees," *Nature machine intelligence*, vol. 2, no. 1, pp. 56–67, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/32607472/

[169] M. T. Ribeiro, S. Singh, and C. Guestrin, "" why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016. [Online]. Available: https://doi.org/10.1145/2939672.2939778 pp. 1135–1144.

[170] E. Štrumbelj and I. Kononenko, "Explaining prediction models and individual predictions with feature contributions," *Knowledge and information systems*, vol. 41, no. 3, pp. 647–665, 2014. [Online]. Available: https://doi.org/10.1007/s10115-013-0679-x

[171] D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe, K. Hansen, and K.-R. Müller, "How to explain individual classification decisions," *The Journal of Machine Learning Research*, vol. 11, pp. 1803–1831, 2010. [Online]. Available: http://jmlr.org/papers/v11/baehrens10a.html

[172] Q. Sun, "Individualized and global feature attributions for gradient boosted trees in the presence of ℓ_2 regularization," *arXiv preprint arXiv:2211.04409*, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2211.04409

[173] M. Loecher, "Unbiased variable importance for random forests," *Communications in Statistics-Theory and Methods*, vol. 51, no. 5, pp. 1413–1425, 2022. [Online]. Available: https://doi.org/10.1080/03610926.2020.1764042

[174] Z. Zhou and G. Hooker, "Unbiased measurement of feature importance in tree-based methods," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 15, no. 2, pp. 1–21, 2021. [Online]. Available: https://doi.org/10.1145/3429445

[175] X. Li, Y. Wang, S. Basu, K. Kumbier, and B. Yu, "A debiased mdi feature importance measure for random forests," *Advances in Neural Information Processing Systems*, vol. 32, 2019. [Online]. Available: https://proceedings.neurips.cc/paper/2019/file/702cafa3bb4c9c86e4a3b6834b45aedd-Paper.pdf

[176] E. Hariton and J. J. Locascio, "Randomised controlled trials—the gold standard for effectiveness research," *BJOG: an international journal of obstetrics and gynaecology*, vol. 125, no. 13, p. 1716, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29916205/

[177] C. W. Granger, "Investigating causal relations by econometric models and cross-spectral methods," *Econometrica: journal of the Econometric Society*, pp. 424–438, 1969. [Online]. Available: https://doi.org/10.2307/1912791

[178] M. Höfler, "Causal inference based on counterfactuals," *BMC medical research methodology*, vol. 5, no. 1, pp. 1–12, 2005. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/16159397/

[179] M. Prosperi, Y. Guo, M. Sperrin, J. S. Koopman, J. S. Min, X. He, S. Rich, M. Wang, I. E. Buchan, and J. Bian, "Causal inference and counterfactual prediction in machine learning for actionable healthcare," *Nature Machine Intelligence*, vol. 2, no. 7, pp. 369–375, 2020. [Online]. Available: https://doi.org/10.1038/s42256-020-0197-y

[180] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30, 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf

[181] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in *International conference on machine learning.* PMLR, 2017. [Online]. Available: https://doi.org/10.48550/arXiv.1704.02685 pp. 3145–3153.

[182] D. Janzing, L. Minorics, and P. Blöbaum, "Feature relevance quantification in explainable ai: A causal problem," in *International Conference on artificial intelligence and statistics.* PMLR, 2020. [Online]. Available: https://proceedings.mlr.press/v108/janzing20a.html pp. 2907–2916.

[183] J. Pearl, "Causal diagrams for empirical research," *Biometrika*, vol. 82, no. 4, pp. 669–688, 1995. [Online]. Available: https://doi.org/10.1093/biomet/82.4.669

[184] D. D. Bhuva, J. Cursons, G. K. Smyth, and M. J. Davis, "Differential co-expression-based detection of conditional relationships in transcriptional data: comparative analysis and application to breast cancer," *Genome biology*, vol. 20, no. 1, pp. 1–21, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31727119/

[185] A. Belyaeva, C. Squires, and C. Uhler, "Dci: learning causal differences between gene regulatory networks," *Bioinformatics*, vol. 37, no. 18, pp. 3067–3069, 2021. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/33704425/

[186] Y. Kim, J. Hao, Y. Gautam, T. B. Mersha, and M. Kang, "Diffgrn: differential gene regulatory network analysis," *International journal of data mining and bioinformatics*, vol. 20, no. 4, p. 362, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31114627/

[187] Y. Li, D. Liu, T. Li, and Y. Zhu, "Bayesian differential analysis of gene regulatory networks exploiting genetic perturbations," *BMC bioinformatics*, vol. 21, no. 1, pp. 1–13, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31918656/

[188] C. Wang, F. Gao, G. B. Giannakis, G. D'urso, and X. Cai, "Efficient proximal gradient algorithm for inference of differential gene networks," *BMC bioinformatics*, vol. 20, no. 1, pp. 1–15, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31046666/

[189] T. Moerman, S. Aibar Santos, C. Bravo González-Blas, J. Simm, Y. Moreau, J. Aerts, and S. Aerts, "Grnboost2 and arboreto: efficient and scalable inference of gene regulatory networks," *Bioinformatics*, vol. 35, no. 12, pp. 2159–2161, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30445495/

[190] J. Sławek and T. Arodź, "Ennet: inferring large gene regulatory networks from expression data using gradient boosting," *BMC systems biology*, vol. 7, no. 1, pp. 1–13, 2013. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/24148309/

[191] J. Pearl et al., "Models, reasoning and inference," *Cambridge, UK: CambridgeUniversityPress*, vol. 19, no. 2, 2000. [Online]. Available: http://library.mpib-berlin.mpg.de/toc/z2008_2219.pdf

[192] B. Xing and M. J. Van Der Laan, "A causal inference approach for constructing transcriptional regulatory networks," *Bioinformatics*, vol. 21, no. 21, pp. 4007–4013, 2005. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/16131521/

[193] D. Heckerman, C. Meek, and G. Cooper, "A bayesian approach to causal discovery," in *Innovations in Machine Learning*. Springer, 2006, pp. 1–28. [Online]. Available: https://doi.org/10.1007/3-540-33486-6_1

[194] A. Hyttinen, F. Eberhardt, and M. Järvisalo, "Do-calculus when the true graph is unknown." in *UAI*, 2015. [Online]. Available: https://dl.acm.org/doi/10.5555/3020847.3020889 pp. 395–404.

[195] L. Hu, C. Gu, M. Lopez, J. Ji, and J. Wisnivesky, "Estimation of causal effects of multiple treatments in observational studies with a binary outcome," *Statistical methods in medical research*, vol. 29, no. 11, pp. 3218–3234, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/32450775/

[196] H. Mathys, J. Davila-Velderrain, Z. Peng, F. Gao, S. Mohammadi, J. Z. Young, M. Menon, L. He, F. Abdurrob, X. Jiang et al., "Single-cell transcriptomic analysis of alzheimer's disease," *Nature*, vol. 570, no. 7761, pp. 332–337, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31042697/

[197] D. Wang, S. Liu, J. Warrell, H. Won, X. Shi, F. C. Navarro, D. Clarke, M. Gu, P. Emani, Y. T. Yang et al., "Comprehensive functional genomic resource and integrative model for the human brain," *Science*, vol. 362, no. 6420, p. eaat8464, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30545857/

[198] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and Édouard Duchesnay, "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, no. 85, pp. 2825–2830, 2011. [Online]. Available: http://jmlr.org/papers/v12/pedregosa11a.html

[199] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin et al., "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016. [Online]. Available: https://doi.org/10.48550/arXiv.1603.04467

[200] J. Zhang, Y. Ji, and L. Zhang, "Extracting three-way gene interactions from microarray data," *Bioinformatics*, vol. 23, no. 21, pp. 2903–2909, 2007. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/17921496/

[201] T.-H. Hsiao, Y.-C. Chiu, P.-Y. Hsu, T.-P. Lu, L.-C. Lai, M.-H. Tsai, T. H.-M. Huang, E. Y. Chuang, and Y. Chen, "Differential network analysis reveals the genome-wide landscape of estrogen receptor modulation in hormonal cancers," *Scientific reports*, vol. 6, no. 1, pp. 1–16, 2016. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/26972162/

[202] D. Amar, H. Safer, and R. Shamir, "Dissection of regulatory networks that are altered in disease via differential co-expression," *PLoS computational biology*, vol. 9, no. 3, p. e1002955, 2013. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/23505361/

[203] Y.-Y. Ho, L. Cope, M. Dettling, and G. Parmigiani, "Statistical methods for identifying differentially expressed gene combinations," in *Gene Function Analysis*. Springer, 2007, pp. 171–191. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/18314583/

[204] Y. Lai, B. Wu, L. Chen, and H. Zhao, "A statistical method for identifying differential gene–gene co-expression patterns," *Bioinformatics*, vol. 20, no. 17, pp. 3146–3155, 2004. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/15231528/

[205] J. A. Dawson and C. Kendziorski, "An empirical bayesian approach for identifying differential coexpression in high-throughput experiments," *Biometrics*, vol. 68, no. 2, pp. 455–465, 2012. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/22004327/

[206] G. G. Galindez, M. List, J. Baumbach, D. B. Blumenthal, and T. Kacprowski, "Inference of differential gene regulatory networks from gene expression data using boosted differential trees," *bioRxiv*, 2022. [Online]. Available: https://doi.org/10.1101/2022.09.26.509450

[207] T. K. Ho, "Random decision forests," in *Proceedings of 3rd international conference on document analysis and recognition*, vol. 1. IEEE, 1995. [Online]. Available: https://doi.org/10.1109/ICDAR.1995.598994 pp. 278–282.

[208] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014. [Online]. Available: http://jmlr.org/papers/v15/srivastava14a.html

[209] J. Piñero, J. M. Ramírez-Anguita, J. Saüch-Pitarch, F. Ronzano, E. Centeno, F. Sanz, and L. I. Furlong, “The disgenet knowledge platform for disease genomics: 2019 update,” *Nucleic acids research*, vol. 48, no. D1, pp. D845–D855, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31680165/

[210] H. Hu, Y.-R. Miao, L.-H. Jia, Q.-Y. Yu, Q. Zhang, and A.-Y. Guo, “Animaltfdb 3.0: a comprehensive resource for annotation and prediction of animal transcription factors,” *Nucleic acids research*, vol. 47, no. D1, pp. D33–D38, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30204897/

[211] J. H. Reiling, A. J. Olive, S. Sanyal, J. E. Carette, T. R. Brummelkamp, H. L. Ploegh, M. N. Starnbach, and D. M. Sabatini, “A creb3–arf4 signalling pathway mediates the response to golgi stress and susceptibility to pathogens,” *Nature cell biology*, vol. 15, no. 12, pp. 1473–1485, 2013. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/24185178/

[212] L. Sampieri, P. Di Giusto, and C. Alvarez, “Creb3 transcription factors: Er-golgi stress transducers as hubs for cellular homeostasis,” *Frontiers in cell and developmental biology*, vol. 7, p. 123, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31334233/

[213] L. Sampieri, M. Funes Chabán, P. Di Giusto, V. Rozés-Salvador, and C. Alvarez, “Creb3l2 modulates nerve growth factor-induced cell differentiation,” *Frontiers in Molecular Neuroscience*, p. 150, 2021. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/34421533/

[214] J. Penney, T. Taylor, N. MacLusky, and R. Lu, “Luman/creb3 plays a dual role in stress responses as a cofactor of the glucocorticoid receptor and a regulator of secretion,” *Frontiers in molecular neuroscience*, vol. 11, p. 352, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30337854/

[215] J. Hoozemans, R. Veerhuis, E. Van Haastert, J. Rozemuller, F. Baas, P. Eikelenboom, and W. Scheper, “The unfolded protein response is activated in alzheimer’s disease,” *Acta neuropathologica*, vol. 110, no. 2, pp. 165–172, 2005. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/15973543/

[216] N. Jin, W. Qian, X. Yin, L. Zhang, K. Iqbal, I. Grundke-Iqbal, C.-X. Gong, and F. Liu, “Creb regulates the expression of neuronal glucose transporter 3: a possible mechanism related to impaired brain glucose uptake in alzheimer’s disease,” *Nucleic acids research*, vol. 41, no. 5, pp. 3240–3256, 2013. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/23341039/

[217] M. Uittenbogaard, K. K. Baxter, and A. Chiaramello, "The neurogenic basic helix-loop-helix transcription factor neurod6 confers tolerance to oxidative stress by triggering an antioxidant response and sustaining the mitochondrial biomass," *ASN neuro*, vol. 2, no. 2, p. AN20100005, 2010. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/20517466/

[218] J.-i. Satoh, Y. Yamamoto, N. Asahina, S. Kitano, and Y. Kino, "Rna-seq data mining: downregulation of neurod6 serves as a possible biomarker for alzheimer's disease brains," *Disease markers*, vol. 2014, 2014. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/25548427/

[219] A. Gatt, H. Lee, G. Williams, S. Thuret, and C. Ballard, "Expression of neurogenic markers in alzheimer's disease: a systematic review and metatranscriptional analysis," *Neurobiology of Aging*, vol. 76, pp. 166–180, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30716542/

[220] S. Barral, R. Cheng, C. Reitz, B. Vardarajan, J. Lee, B. Kunkle, G. Beecham, L. S. Cantwell, M. A. Pericak-Vance, L. A. Farrer et al., "Linkage analyses in caribbean hispanic families identify novel loci associated with familial late-onset alzheimer's disease," *Alzheimer's & Dementia*, vol. 11, no. 12, pp. 1397–1406, 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/26433351/

[221] K. D. Fowler, J. M. Funt, M. N. Artyomov, B. Zeskind, S. E. Kolitz, and F. Towfic, "Leveraging existing data sets to generate new insights into alzheimer's disease biology in specific patient subsets," *Scientific reports*, vol. 5, no. 1, pp. 1–14, 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/26395074/

[222] P. Vanhoutte, J. L. Nissen, B. Brugg, B. Della Gaspera, M.-J. Besson, R. A. Hipskind, and J. Caboche, "Opposing roles of elk-1 and its brain-specific isoform, short elk-1, in nerve growth factor-induced pc12 differentiation," *Journal of Biological Chemistry*, vol. 276, no. 7, pp. 5189–5196, 2001. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/11050086/

[223] A. Sharma, L. M. Callahan, J.-Y. Sul, T. K. Kim, L. Barrett, M. Kim, J. M. Powers, H. Federoff, and J. Eberwine, "A neurotoxic phosphoform of elk-1 associates with inclusions from multiple neurodegenerative diseases," *PLoS One*, vol. 5, no. 2, p. e9002, 2010. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/20126313/

[224] A. Besnard, B. Galan-Rodriguez, P. Vanhoutte, and J. Caboche, "Elk-1 a transcription factor with multiple facets in the brain," *Frontiers in neuroscience*, vol. 5, p. 35, 2011. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/21441990/

[225] E. M. Szatmari, A. F. Oliveira, E. J. Sumner, and R. Yasuda, "Centaurin-α1-ras-elk-1 signaling at mitochondria mediates β-amyloid-induced synaptic dysfunction," *Journal of Neuroscience*, vol. 33, no. 12, pp. 5367–5374, 2013. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/23516302/

[226] M. Tremblay, O. Sanchez-Ferras, and M. Bouchard, “Gata transcription factors in development and disease,” *Development*, vol. 145, no. 20, p. dev164384, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30348673/

[227] J. Liu, X. Wang, J. Li, H. Wang, G. Wei, and J. Yan, “Reconstruction of the gene regulatory network involved in the sonic hedgehog pathway with a potential role in early development of the mouse brain,” *PLoS computational biology*, vol. 10, no. 10, p. e1003884, 2014.

[228] E. Conti, L. Tremolizzo, M. E. Santarone, M. Tironi, I. Radice, C. P. Zoia, A. Aliprandi, A. Salmaggi, R. Dominici, M. Casati et al., “Donepezil modulates the endogenous immune response: implications for alzheimer’s disease,” *Human Psychopharmacology: Clinical and Experimental*, vol. 31, no. 4, pp. 296–303, 2016. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/27297668/

[229] X. Zhang, M. Zou, Y. Wu, D. Jiang, T. Wu, Y. Zhao, D. Wu, J. Cui, and G. Li, “Regulation of the late onset alzheimer’s disease associated hla-dqa1/drb1 expression,” *American Journal of Alzheimer’s Disease & Other Dementias®*, vol. 37, p. 15333175221085066, 2022. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/35341343/

[230] J. M. Eissman, L. Dumitrescu, E. R. Mahoney, A. N. Smith, S. Mukherjee, M. L. Lee, P. Scollard, S. E. Choi, W. S. Bush, C. D. Engelman et al., “Sex differences in the genetic architecture of cognitive resilience to alzheimer’s disease,” *Brain*, vol. 145, no. 7, pp. 2541–2554, 2022. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/35552371/

[231] M. Fagny, J. N. Paulson, M. L. Kuijjer, A. R. Sonawane, C.-Y. Chen, C. M. Lopes-Ramos, K. Glass, J. Quackenbush, and J. Platig, “Exploring regulation in tissues with eqtl networks,” *Proceedings of the National Academy of Sciences*, vol. 114, no. 37, pp. E7841–E7850, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28851834/

[232] R. S. Laskar, D. C. Muller, P. Li, M. J. Machiela, Y. Ye, V. Gaborieau, M. Foll, J. N. Hofmann, L. Colli, J. N. Sampson et al., “Sex specific associations in genome wide association analysis of renal cell carcinoma,” *European Journal of Human Genetics*, vol. 27, no. 10, pp. 1589–1598, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31231134/

[233] N. J. Prescott, K. M. Dominy, M. Kubo, C. M. Lewis, S. A. Fisher, R. Redon, N. Huang, B. E. Stranger, K. Blaszczyk, B. Hudspith et al., “Independent and population-specific association of risk variants at the irgm locus with crohn’s disease,” *Human molecular genetics*, vol. 19, no. 9, pp. 1828–1839, 2010. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/20106866/

[234] S. R. Cole and M. A. Hernán, “Constructing inverse probability weights for marginal structural models,” *American journal of epidemiology*, vol. 168, no. 6, pp. 656–664, 2008. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/18682488/

[235] U. Shalit, F. D. Johansson, and D. Sontag, “Estimating individual treatment effect: generalization bounds and algorithms,” in *International Conference on Machine Learning.* PMLR, 2017. [Online]. Available: https://proceedings.mlr.press/v70/shalit17a.html pp. 3076–3085.

[236] B. Baur, J. Shin, S. Zhang, and S. Roy, “Data integration for inferring context-specific gene regulatory networks,” *Current opinion in systems biology*, vol. 23, pp. 38–46, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/33225112/

[237] T. Heskes, E. Sijben, I. G. Bucur, and T. Claassen, “Causal shapley values: Exploiting causal knowledge to explain individual predictions of complex models,” *Advances in neural information processing systems*, vol. 33, pp. 4778–4789, 2020. [Online]. Available: https://proceedings.neurips.cc/paper/2020/file/32e54441e6382a7fbacbbbaf3c450059-Paper.pdf

[238] Y. Kwon and J. Zou, “Weightedshap: analyzing and improving shapley based feature attributions,” *arXiv preprint arXiv:2209.13429*, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2209.13429

[239] H. Shu, J. Zhou, Q. Lian, H. Li, D. Zhao, J. Zeng, and J. Ma, “Modeling gene regulatory networks using neural network architectures,” *Nature Computational Science*, vol. 1, no. 7, pp. 491–501, 2021. [Online]. Available: https://doi.org/10.1038/s43588-021-00099-8

[240] X. Chen, H. Sun, C. Ellington, E. Xing, and L. Song, “Multi-task learning of order-consistent causal graphs,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 11 083–11 095, 2021. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2021/file/5c3a3b139a11689e0bc55abd95e20e39-Paper.pdf

[241] L. Lorch, S. Sussex, J. Rothfuss, A. Krause, and B. Schölkopf, “Amortized inference for causal structure learning,” *arXiv preprint arXiv:2205.12934*, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2205.12934

[242] E. Winter, “The shapley value,” *Handbook of game theory with economic applications*, vol. 3, pp. 2025–2054, 2002. [Online]. Available: https://doi.org/10.1016/S1574-0005(02)03016-3

[243] L.-L. Sun, S.-L. Yang, H. Sun, W.-D. Li, and S.-R. Duan, “Molecular differences in alzheimer’s disease between male and female patients determined by integrative network analysis,” *Journal of Cellular and Molecular Medicine*, vol. 23, no. 1, pp. 47–58, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30394676/

[244] H. I. Jacobs, M. P. Van Boxtel, J. Jolles, F. R. Verhey, and H. B. Uylings, “Parietal cortex matters in alzheimer’s disease: an overview of structural, functional and metabolic findings,” *Neuroscience & Biobehavioral Reviews*, vol. 36, no. 1, pp. 297–309, 2012. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/21741401/

[245] S. Nischwitz, S. Cepok, A. Kroner, C. Wolf, M. Knop, F. Müller-Sarnowski, H. Pfister, D. Roeske, P. Rieckmann, B. Hemmer et al., "Evidence for vav2 and znf433 as susceptibility genes for multiple sclerosis," *Journal of neuroimmunology*, vol. 227, no. 1-2, pp. 162–166, 2010. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/20598377/

[246] K. Yeghiazaryan, D. Turhani-Schatzmann, O. Labudova, E. Schuller, E. Olson, N. Cairns, and G. Lubec, *Downregulation of the transcription factor scleraxis in brain of patients with Down syndrome.* Springer, 1999. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/10666685/

[247] R. Lardenoije, J. A. Roubroeks, E. Pishva, M. Leber, H. Wagner, A. Iatrou, A. R. Smith, R. G. Smith, L. M. Eijssen, L. Kleineidam et al., "Alzheimer's disease-associated (hydroxy) methylomic changes in the brain and blood," *Clinical epigenetics*, vol. 11, no. 1, pp. 1–15, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31775875/

[248] M. Bik-Multanowski and A. Dobosz, "Detection of high expression of complex i mitochondrial genes can indicate low risk of alzheimer's disease," *International Journal of Clinical and Experimental Pathology*, vol. 10, no. 2, 2017. [Online]. Available: https://e-century.us/files/ijcep/10/2/ijcep0031015.pdf

[249] S. Lee, Y. Nam, J. Y. Koo, D. Lim, J. Park, J. Ock, J. Kim, K. Suk, and S. B. Park, "A small molecule binding hmgb1 and hmgb2 inhibits microglia-mediated neuroinflammation," *Nature chemical biology*, vol. 10, no. 12, pp. 1055–1060, 2014. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/25306442/

[250] Y. Liu, X. Chen, Y. Che, H. Li, Z. Zhang, W. Peng, and J. Yang, "Lncrnas as the regulators of brain function and therapeutic targets for alzheimer's disease." *Aging & Disease*, vol. 13, no. 3, 2022. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/35656102/

[251] M. Shinohara, M. Tachibana, T. Kanekiyo, and G. Bu, "Role of lrp1 in the pathogenesis of alzheimer's disease: evidence from clinical and preclinical studies: Thematic review series: Apoe and lipid homeostasis in alzheimer's disease," *Journal of lipid research*, vol. 58, no. 7, pp. 1267–1281, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28381441/

[252] C. Bellenguez, F. Küçükali, I. E. Jansen, L. Kleineidam, S. Moreno-Grau, N. Amin, A. C. Naj, R. Campos-Martin, B. Grenier-Boley, V. Andrade et al., "New insights into the genetic etiology of alzheimer's disease and related dementias," *Nature genetics*, vol. 54, no. 4, pp. 412–436, 2022. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/35379992/

[253] J. Binder, O. Ursu, C. Bologa, S. Jiang, N. Maphis, S. Dadras, D. Chisholm, J. Weick, O. Myers, P. Kumar et al., "Machine learning prediction and tau-based screening identifies potential alzheimer's disease genes relevant to immunity," *Communications Biology*, vol. 5, no. 1, p. 125, 2022. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/35149761/

[254] R. Lampinen, V. Górová, S. Avesani, J. R. Liddell, E. Penttilä, T. Závodná, Z. Krejčík, J.-M. Lehtola, T. Saari, J. Kalapudas et al., "Biometal dyshomeostasis in olfactory mucosa of alzheimer's disease patients," *International Journal of Molecular Sciences*, vol. 23, no. 8, p. 4123, 2022. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/35456941/

[255] L. Li, M. Ghorbani, M. Weisz-Hubshman, J. Rousseau, I. Thiffault, R. E. Schnur, C. Breen, R. Oegema, M. M. Weiss, Q. Waisfisz et al., "Lysine acetyltransferase 8 is involved in cerebral development and syndromic intellectual disability," *The Journal of clinical investigation*, vol. 130, no. 3, pp. 1431–1445, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31794431/

[256] F. Chen, H. Chen, Y. Jia, H. Lu, Q. Tan, and X. Zhou, "mir-149-5p inhibition reduces alzheimer's disease β-amyloid generation in 293/appsw cells by upregulating h4k16ac via kat8," *Experimental and therapeutic medicine*, vol. 20, no. 5, pp. 1–1, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/32973937/

[257] R. E. Marioni, S. E. Harris, Q. Zhang, A. F. McRae, S. P. Hagenaars, W. D. Hill, G. Davies, C. W. Ritchie, C. R. Gale, J. M. Starr et al., "Gwas on family history of alzheimer's disease," *Translational psychiatry*, vol. 8, no. 1, p. 99, 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29777097/

[258] S. K. Jaladanki, A. Elmas, G. S. Malave, and K.-l. Huang, "Genetic dependency of alzheimer's disease-associated genes across cells and tissue types," *Scientific Reports*, vol. 11, no. 1, p. 12107, 2021. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/34103633/

[259] W. Zhao, J. A. Smith, M. Yu, P. Moorjani, A. Ganna, A. B. Dey, J. Lee, and S. L. Kardia, "Common and rare variants in alzheimer's disease genes are associated with episodic memory in south asians from the lasi-dad study: Genetics/genetic factors of alzheimer's disease," *Alzheimer's & Dementia*, vol. 16, p. e045189, 2020. [Online]. Available: https://doi.org/10.1002/alz.045189

[260] A. Alachkar, S. Agrawal, M. Baboldashtian, K. Nuseir, J. Salazar, and A. Agrawal, "L-methionine enhances neuroinflammation and impairs neurogenesis: Implication for alzheimer's disease," *Journal of Neuroimmunology*, vol. 366, p. 577843, 2022. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/35299077/

[261] K. Beyer, J. I. Lao, P. Latorre, N. Riutort, B. Matute, T. M. Fernández-Figueras, J. L. Mate, and A. Ariza, "Methionine synthase polymorphism is a risk factor for alzheimer disease," *Neuroreport*, vol. 14, no. 10, pp. 1391–1394, 2003. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/12876480/

[262] P. Bosco, R. Guéant-Rodríguez, G. Anello, A. Romano, B. Namour, R. Spada, F. Caraci, G. Tringali, R. Ferri, and J. Guéant, "Association of il-1 rn* 2 allele and methionine synthase 2756 aa genotype with dementia severity of sporadic alzheimer's disease," *Journal of Neurology, Neurosurgery & Psychiatry*, vol. 75, no. 7, pp. 1036–1038, 2004. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/15201366/

[263] A. L. Knight, X. Yan, S. Hamamichi, R. R. Ajjuri, J. R. Mazzulli, M. W. Zhang, J. G. Daigle, S. Zhang, A. R. Borom, L. R. Roberts et al., "The glycolytic enzyme, gpi, is a functionally conserved modifier of dopaminergic neurodegeneration in parkinson's models," *Cell metabolism*, vol. 20, no. 1, pp. 145–157, 2014. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/24882066/

[264] H. Li, L. Zou, J. Shi, and X. Han, "Bioinformatics analysis of differentially expressed genes and identification of an mirna–mrna network associated with entorhinal cortex and hippocampus in alzheimer's disease," *Hereditas*, vol. 158, pp. 1–13, 2021. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/34243818/

[265] D. A. Butterfield, M. Favia, I. Spera, A. Campanella, M. Lanza, and A. Castegna, "Metabolic features of brain function with relevance to clinical features of alzheimer and parkinson diseases," *Molecules*, vol. 27, no. 3, p. 951, 2022. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/35164216/

[266] D. Banerjee and K. Nandagopal, "Potential interaction between the gars-airs-gart gene and cp2/lbp-1c/lsf transcription factor in down syndrome-related alzheimer disease," *Cellular and Molecular Neurobiology*, vol. 27, pp. 1117–1126, 2007. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/17902044/

[267] S. Ahmad, M. d. C. Milan, O. Hansson, A. Demirkan, R. Agustin, M. E. Sáez, N. Giagtzoglou, A. Cabrera-Socorro, M. H. Bakker, A. Ramirez et al., "Cdh6 and hagh protein levels in plasma associate with alzheimer's disease in apoe $\varepsilon 4$ carriers," *Scientific Reports*, vol. 10, no. 1, pp. 1–13, 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/32427856/

[268] G. Novikova, M. Kapoor, J. Tcw, E. M. Abud, A. G. Efthymiou, S. X. Chen, H. Cheng, J. F. Fullard, J. Bendl, Y. Liu et al., "Integration of alzheimer's disease genetics and myeloid genomics identifies disease risk regulatory elements and genes," *Nature communications*, vol. 12, no. 1, p. 1610, 2021. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/33712570/

[269] S. Zhou, G. Ma, H. Luo, S. Shan, J. Xiong, and G. Cheng, "Identification of 5 potential predictive biomarkers for alzheimer's disease by integrating the unified test for molecular signatures and weighted gene coexpression network analysis," *The Journals of Gerontology: Series A*, vol. 78, no. 4, pp. 653–658, 2023. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/36048437/

[270] T. Xiao, B. Jiao, W. Zhang, C. Pan, J. Wei, X. Liu, Y. Zhou, L. Zhou, B. Tang, and L. Shen, "Identification of chchd10 mutation in chinese patients with alzheimer disease," *Molecular Neurobiology*, vol. 54, pp. 5243–5247, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/27578015/

[271] X.-Q. Che, Q.-H. Zhao, Y. Huang, X. Li, R.-J. Ren, S.-D. Chen, G. Wang, and Q.-H. Guo, "Genetic features of mapt, grn, c9orf72 and chchd10 gene mutations in chinese patients with frontotemporal dementia," *Current Alzheimer Research*, vol. 14, no. 10, pp. 1102–1108, 2017. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/28462717/

[272] L. Caberlotto, T. Nguyen, M. Lauria, C. Priami, R. Rimondini, S. Maioli, A. Cedazo-Minguez, G. Sita, F. Morroni, M. Corsi et al., "Cross-disease analysis of alzheimer's disease and type-2 diabetes highlights the role of autophagy in the pathophysiology of two highly comorbid diseases," *Scientific reports*, vol. 9, no. 1, pp. 1–13, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30850634/

[273] M. S. Gurses, M. N. Ural, M. A. Gulec, O. Akyol, and S. Akyol, "Pathophysiological function of adamts enzymes on molecular mechanism of alzheimer's disease," *Aging and disease*, vol. 7, no. 4, p. 479, 2016. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/27493839/

[274] Y. Yamakage, M. Kato, A. Hongo, H. Ogino, K. Ishii, T. Ishizuka, T. Kamei, H. Tsuiji, T. Miyamoto, H. Oishi et al., "A disintegrin and metalloproteinase with thrombospondin motifs 2 cleaves and inactivates reelin in the postnatal cerebral cortex and hippocampus, but not in the cerebellum," *Molecular and Cellular Neuroscience*, vol. 100, p. 103401, 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31491533/