WENDY: Covariance Dynamics Based Gene Regulatory Network Inference

Yue Wang[1,*], Peng Zheng[2,3], Yu-Chen Cheng[4,5,6,7], Zikun Wang[8], and Aleksandr Aravkin[9]

[1]Irving Institute for Cancer Dynamics and Department of Statistics, Columbia University, New York, NY 10027, USA
[2]Institute for Health Metrics and Evaluation, Seattle, WA 98195, USA
[3]Department of Health Metrics Sciences, University of Washington, Seattle, WA 98195, USA
[4]Department of Data Science, Dana-Farber Cancer Institute, Boston, MA 02215, USA
[5]Department of Biostatistics, Harvard T.H. Chan School of Public Health, Boston, MA 02115, USA
[6]Center for Cancer Evolution, Dana-Farber Cancer Institute, Boston, MA 02215, USA
[7]Department of Stem Cell and Regenerative Biology, Harvard University, Cambridge, MA 02138, USA
[8]Laboratory of Genetics, The Rockefeller University, New York, NY 10065, USA
[9]Department of Applied Mathematics, University of Washington, Seattle, WA 98195, USA
[*]yw4241@columbia.edu

Abstract

Determining gene regulatory network (GRN) structure is a central problem in biology, with a variety of inference methods available for different types of data. For a widely prevalent and challenging use case, namely single-cell gene expression data measured after intervention at multiple time points with unknown joint distributions, there is only one known specifically developed method, which does not fully utilize the rich information contained in this data type. We develop an inference method for the GRN in this case, netWork infErence by covariaNce DYnamics, dubbed WENDY. The core idea of WENDY is to model the dynamics of the covariance matrix, and solve this dynamics as an optimization problem to determine the regulatory relationships. To evaluate its effectiveness, we compare WENDY with other inference methods using synthetic data and experimental data. Our results demonstrate that WENDY performs well across different data sets.

1 Introduction

In general, a gene is transcribed into mRNA and then translated into proteins. This process, known as gene expression, commonly employs mRNA count or protein count to denote the expression level. In addition to directly changing cell phenotypes (Qian and Cheng, 2020; Cheng et al., 2023b), influencing extracellular processes (Axelrod et al., 2023), or even manipulating macroscopic neurological circuitry (Li et al., 2021; Vijayan et al., 2022), certain proteins can affect the transcription of other genes (mutual regulation) or their own corresponding genes (autoregulation). Genes and their regulatory relationships form a gene regulatory network (GRN).

Determining the GRN structure is a central problem in biology, as it reveals how a living organism is maintained (Axelrod et al., 2021), and provides control of essential biological processes (Wang, 2020), especially treating cancer (McDonald et al., 2023; Cheng et al., 2023a). However, directly establishing the GRN using traditional technologies is extremely difficult since they cannot measure the expression levels of many genes within the same cell. Instead, numerous methods have been developed to infer the GRN structure from gene expression data. Particularly, recent advancements in single-cell RNA-sequencing technologies have made it possible to profile the whole transcriptome of single cells at large-scale. However, single-cell RNA-seq can only measure one time point because cells have to be killed during the experimental process, making it challenging to study gene regulation relationships that require multiple observations over time.

In this paper, we focus on a specific data type arising from the following setup: First, implement an intervention that affects gene expression (e.g., drugs). Then measure the expression level (generally mRNA count) of n genes for different single cells at multiple time points, and select the data from time points where the expression has not yet reached a stationary state. Since gene expression at the single-cell level is stochastic, for each time point, we obtain many samples of an n-dimensional random vector. However, since we need to kill a cell before measuring its gene expression levels, one cell can only be measured once. Thus, we measure different cells at different time points, and we do not have a joint distribution for gene expression at different time points. Although this approach has become common in recent experimental research (Chakraborty et al., 2021), and it provides more informative data compared to most other approaches, to our knowledge, there is only one inference method developed specifically for this data type, SINCERITIES (Papili Gao et al., 2018). A major limitation of SINCERITIES is that it requires data from at least six time points to perform well. Additionally, for single-cell expression data of n genes over T time points, this method only extracts $n(T-1)$ numbers for further analyses, implying low data utilization efficiency. There have been many inference methods for single-cell time series gene expression data, where the joint distribution of expression levels at different time points is known (Huynh-Thu and Geurts, 2018; Zheng et al., 2019). Since obtaining the joint distribution of gene expression is difficult, such methods are usually not practically applicable. There are also many inference methods for single-cell gene expression data

measured at a single time point (Basso et al., 2005; Zhang et al., 2012), or bulk level gene expression data measured at multiple time points after interventions (Perrin et al., 2003; Ma et al., 2020; Wang et al., 2022). These data types are more common because of their low cost. Nevertheless, they provide less information compared to the data type we examine in this study, namely, time series data from single-cell gene expression. Therefore, while it is feasible to convert our considered data type into these more common forms and use corresponding inference methods, such transformations result in a significant loss of the rich information inherent in the original dataset.

In this paper, we introduce an algorithm named NetWork infErence by covariaNce DYnamics (WENDY), designed to connect single-cell gene expression data at different time points, even in the absence of knowledge about the joint distribution. The core idea behind WENDY is to compute the covariance matrices of gene expression levels at two time points and model the evolution of these covariance matrices over time. To infer the GRN, we formulate a non-convex optimization problem based on the dynamics of covariance matrices and derive a numerical solution. For a visual representation of WENDY's workflow, refer to Figure 1.

One of WENDY's key advantages is its requirement of only two time points worth of data. This feature is particularly valuable in scenarios where intervention and / or measurement ultimately result in cell death, precluding measurements at additional time points. However, if data from more time points are available, WENDY can still be applied to each pair of neighboring time points to detect potential rapid changes in the GRN during the experiment. Furthermore, for single cell expression data comprising n genes across T time points, WENDY extracts $(0.5n^2 + 0.5n)T$ numbers for further analyses, indicating significantly higher data utilization efficiency.

The paper proceeds as follows. In Section 2, we present a classification framework for gene expression data and review existing GRN inference methods. Section 3 details the WENDY method, including the mathematical gene expression model and the approach to solving the dynamics of this model. In Section 4, we evaluate WENDY and other GRN inference methods using synthetic data to compare their performance. In Section 5, we evaluate WENDY and other GRN inference methods using experimental data to compare their performance. Finally, we conclude with discussions in Section 6.

2 Data classification and literature review

2.1 A framework for data classification

There are different types of gene expression data that can be used to infer the GRN structure. Different data types correspond to different inference methods. We first present a framework for classifying related data types, modified from the framework by Wang and Wang (2022). See Table 1 for this classification framework. There are different dimensions to classify data types.

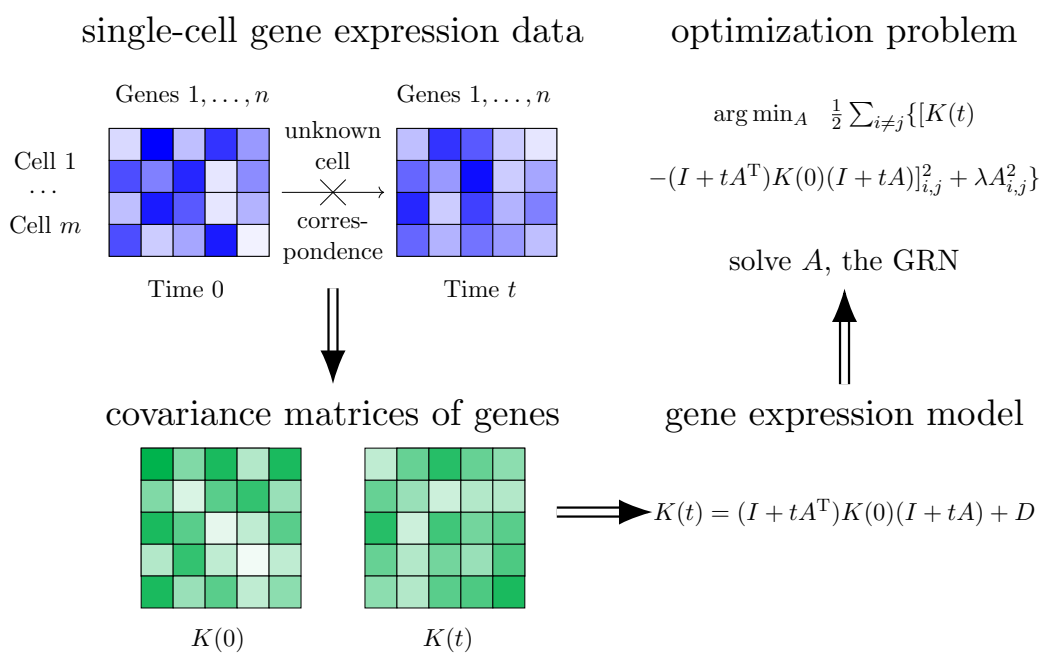

Figure 1: Workflow of the WENDY method. Given single-cell level gene expression data at two time points, where the joint distribution (cell correspondence) between two time points is unknown, first calculate the covariance matrix of gene expression for each time point. Then use the mathematical gene expression model to derive the equation of covariance matrices. Last, transform this into an optimization problem and solve the GRN numerically.

(1) We can measure the gene expression levels when the dynamics of gene expression is stationary (invariant along time), or we can add an intervention to drive the dynamics of gene expression away from stationarity, and measure the gene expression levels when they gradually return to the (possibly new) stationary state. For the intervention, we consider general interventions such as adding drugs (we cannot control which genes are affected) and specific interventions such as gene knockdown and gene knockout (we can select any genes to affect). Considering our capability to measure gene expression levels pre- and post-intervention, a specific intervention yields more informative data than a general one. Moreover, scenarios with intervention are richer in information compared to those without, where only stationary expression levels are observed.

(2) We can measure the average expression levels of many cells (bulk level) or measure the expression levels for each single cell (single-cell level). On single-cell level, gene expression is essentially stochastic, and we shall obtain a random variable for the expression level of each gene. On bulk level, the stochasticity is averaged out, and we should obtain a deterministic value for the expression level of each gene. Single-cell level measurement is more informative than bulk level measurement.

In practice, repeating the same bulk level measurement can still lead to different values, making some researchers regard such data as stochastic and apply inference methods designed for single-cell data (Basso et al., 2005). Nevertheless, at bulk level, randomness from single cells is averaged out, and the different values from bulk level measurement can only come from systematic differences, such as different cell phenotypes or different environmental factors. Such unobserved systematic differences can affect multiple genes and make them correlated, although these genes might not have direct regulatory relations. Therefore, we do not consider bulk level data that have different values for the same measurement.

(3) We can measure expression levels at one time point or multiple time points. When we measure expression level at multiple time points, one essential issue is whether we can measure the same cell multiple times. For bulk level data, this does not matter, as the data are deterministic, and whether the cells at $t+1$ are the same as the cells at t should not make a difference. However, for single-cell level data, since the measured levels are stochastic, there is an essential difference. Denote the single-cell expression level of a gene at time t as $X(t)$. If the same cell can be measured multiple times, then we have the joint probability distribution of a time series, $\mathbb{P}[X(0) = x_0, X(1) = x_1, X(2) = x_2, \ldots]$. Otherwise, we only have marginal probability distributions for each time point, $\mathbb{P}[X(0) = x_0]$, $\mathbb{P}[X(1) = x_1]$, $\mathbb{P}[X(2) = x_2]$, $\ldots$, but not the correspondence between time points, and certain quantities cannot be calculated, such as the correlation coefficients of expression levels at two time points. Time series data types are more informative than one-time data types, and joint distribution is more informative than marginal distributions.

In practice, measuring the expression levels of many genes is destructive, and we cannot measure the same cell more than once. If we only want to measure the expression level of a single gene, there are some techniques (fluorescent proteins (Wu et al., 2011), etc.) that

can measure the same cell multiple times. Another approach is to measure the amount of spliced and unspliced mRNAs, which provides both the current expression level and an approximation of its time derivative (RNA velocity (La Manno et al., 2018)). This approach provides two measurements of the same cell, and some inference methods for time series data can be applied.

	One-time		Time series		
	Bulk	Single-cell	Bulk	Single-cell	
				Marginal distribution	Joint distribution
Stationary	1: No	2: Yes	3: No	4: Ditto	5: Yes
General intervention	6: No	7: Ditto	8: Yes	9: Ditto/Yes	10: Ditto
Specific intervention	11: Yes*	12: Ditto	13: Ditto	14: Ditto	15: Ditto

Table 1: Classification of data types regarding GRN inference, modified from the framework by Wang and Wang (2022). The data types are classified by different dimensions: (1) The gene expression is at stationary, or is driven away from stationary by an intervention (on general genes that we cannot choose, or specific genes that we can choose); (2) Measure at one time point or multiple time points (time series); (3) Measure the average over many genes (bulk level) or on single-cell level. When measuring at multiple time points on single-cell level, one more dimension is whether we have the joint distribution over different time points. For different data types (scenarios), we study whether the GRN structure can be inferred. For each data type, No means that the GRN structure cannot be inferred. Ditto means that there are no specific inference methods, but the GRN structure can be inferred using the same method for a less informative data type. Yes means specific inference methods exist. Scenario 11 only has an inference method that also requires the data in Scenario 1. We focus on Scenario 9, which is not as well-studied and has only one specific inference method.

2.2 Known inference methods for different data types

Given this classification framework, we can review inference methods for different data types. In this framework, there are 15 data types (scenarios). Some data types do not have enough information that can be used to infer the GRN structure. Some data types (e.g., Scenario 13) have more information than some other data types (e.g., Scenario 8),

but the extra information cannot lead to new GRN inference methods. Thus for such scenarios, we can only use methods for other less informative scenarios. This approach loses a lot of information, and therefore cannot justify the time and money spent to obtain more informative data. Some data types have extra information that allows for inference methods that work for such scenarios but not for less informative scenarios.

For bulk level data types, since we only have a single deterministic value for each gene, it is difficult to obtain the correlation between genes. For Scenarios 1, 3, 6, the GRN structure cannot be inferred. For Scenarios 8 and 13, we can regard the gene expression time series as solution trajectories of an ordinary differential equation (ODE) system. If we assume that the ODE system is linear (Perrin et al., 2003) or has certain nonlinear forms (Ma et al., 2020), we can discretize the ODE system into an algebraic equation system and use regression to infer the ODE parameters. Here the ODE parameters represent the GRN. We can infer all the edges, including the directions. For Scenario 11, one can add an intervention on each gene and observe which genes (descendants of this gene in the GRN) are also affected. Such ancestor-descendant relationships can be used to partially infer the GRN structure (Wang and Wang, 2022). Not all regulatory relationships (edges) can be inferred, but one can infer at least $n-1$ edges for a GRN with n genes.

For single-cell level one-time data types (Scenarios 2, 7, 12), there have been numerous GRN inference methods for Scenario 2, while Scenarios 7 and 12 generally do not have extra information that supports specific inference methods. For Scenario 2, most inference methods turn the GRN inference problem into a feature selection problem: select genes whose levels can be best used to predict the level of the target gene. Then such selected genes might have regulatory relationships with the target gene. The selection can be made by calculating certain quantities between the target gene and the candidate genes that measure their similarity: covariance (Nouri et al., 2023), mutual information (Basso et al., 2005), or other information theory quantities (Zhang et al., 2012). Besides, one can apply regression (Haury et al., 2012), decision tree (Huynh-Thu et al., 2010), or other machine learning and deep learning (Shrivastava et al., 2022; Zhao et al., 2022; Zhong and Tibshirani, 2019) methods to directly select out genes that can predict the target gene. Regularization terms (e.g., L_1 (Gustafsson et al., 2005) and L_2 (Kamimoto et al., 2023) regularizers) can be added to the regression to make the result sparse. Besides the idea of feature selection, another approach is to build probabilistic models (Bayesian network (Agostinho et al., 2015; Lee et al., 2019), stochastic differential equation (SDE) (Wang et al., 2023), and others (Liu et al., 2016; Burdziak et al., 2023)), and use likelihood to determine the most probable network. Since the number of candidate networks is large, a common solution is to apply Markov chain Monte Carlo (MCMC) to approximate the network probabilities (Morrissey et al., 2010; Agostinho et al., 2015; Aalto et al., 2020; Ghosh and Zhong, 2021; Zhong, 2021). Deterministic models, such as Boolean networks (Lim et al., 2016), can also be used. There is a well-developed platform that combines different inference methods for Scenario 2 (Wen et al., 2023). One problem of Scenario 2 is to determine the direction of a regulatory relationship, since if the level of V_i can predict

the level of V_j, then generally the level of V_j can also predict the level of V_i. To solve this problem, one can add specific interventions (Scenario 12) on V_i to see whether V_j is affected.

For single-cell level time series data types without joint distribution of different time points (Scenarios 4, 9, 14), it is common to treat Scenario 4 as Scenario 2 (treat data at different time points separately), and treat Scenarios 9, 14 as Scenario 8 (average over different cells), and apply corresponding inference methods. The only GRN inference method we know that works specifically for Scenario 9 (but not any less informative scenarios) is SINCERITIES (Papili Gao et al., 2018), which considers the Kolmogorov–Smirnov distance between the distributions of the same gene at two time points, and then applies linear regression, similar to methods for Scenario 8. There are two other regression-based methods, Harissa and CARDAMOM (Herbach, 2023; Ventre, 2021), that essentially work for Scenario 2, but can partially (and insufficiently) integrate the time information when work for Scenario 9.

For single-cell level time series data types with joint distribution of different time points (Scenarios 5, 10, 15), there have been numerous GRN inference methods for Scenario 5, while Scenarios 10 and 15 generally do not have extra information that support specific inference methods. For Scenario 5, most inference methods are similar to those for Scenario 2, especially those methods on regression (Zhang et al., 2021a), tree-based feature selection (Huynh-Thu and Geurts, 2018; Zheng et al., 2019), or more advanced machine learning tools (Nauta et al., 2019; Atanackovic et al., 2023), although there are also inference methods based on more complicated biological models (Huynh-Thu and Sanguinetti, 2015). One common approach is to model the gene expression by a vector autoregressive model (either linear or nonlinear) (Fujita et al., 2007a,b; Siggiridou and Kugiumtzis, 2015), and then use Granger causality to determine whether one gene directly regulates another gene (Fujita et al., 2010a,b; Nagarajan and Upreti, 2010; Zhang et al., 2010). This approach solves the famous "correlation does not imply causation" problem from two aspects. First, when gene V_i and gene V_j are correlated, it is possible that they are not directly regulating each other, such as $V_i \leftarrow V_k \rightarrow V_j$ or $V_i \rightarrow V_k \rightarrow V_j$. In this case, given the values of V_k, V_i cannot provide more information for V_j, and Granger causality can determine that V_i does not directly regulate V_j. Second, when there is directly regulation between V_i and V_j, we do not know whether V_i is the cause or the result of V_j. Since Granger causality determines whether the past of V_i contains unique information of the future of V_j, the direction of regulation is also known, since causality can only travel forward along time. This explains why most methods for Scenario 5 can determine the direction of regulations, different from their analogies for Scenario 2.

Besides treating a more informative data type as a less informative data type, one can also use certain methods to transform a less informative data type into a more informative data type, provided there are certain assumptions about the underlying systems. For instance, from single-cell one-time data (Scenarios 2, 7, 12), one can construct the pseudotime to transform the data into time series data (Reid and Wernisch, 2016; Street et al.,

2018), and apply corresponding inference methods (Deshpande et al., 2022).

Different methods need different assumptions regarding gene expression and gene regulation. For instance, some methods need the gene expression dynamics to be linear (Perrin et al., 2003), and some other methods need the GRN to have no directed cycle (Zhang et al., 2012). Besides, different methods have different inference abilities: some methods can determine all edges, including the direction (Huynh-Thu and Geurts, 2018), while some other methods can only partially determine some edges, and/or cannot determine the edge direction (Wang and Wang, 2022).

Most GRN inference methods can only determine regulations between genes, but not autoregulation. Autoregulation inference methods generally need stronger model assumptions (Xing and Van Der Laan, 2005), more informative data types (Feigelman et al., 2016), or only produce partial results (Wang and He, 2023).

The above discussion only considers the situation of inferring GRN after obtaining all data. Another situation is to design intervention experiments, so that the GRN can be inferred with the minimal cost (Cho et al., 2016).

Readers may refer to other reviews for more details about GRN inference methods for different scenarios (Barbosa et al., 2018; Pratapa et al., 2020; Zhao et al., 2021; Nguyen et al., 2021; Wang and Wang, 2022) or for other data types (besides mRNA/protein count) that can help with GRN inference, such as ChIP-seq and ATAC-seq (Erbe et al., 2022; Badia-i Mompel et al., 2023).

3 Novel GRN inference method

In this section, we present an algorithm of netWork infErence by covariaNce DYnamics (WENDY), that works for Scenario 9, single-cell level time series data without joint distribution of different time points, measured after general interventions. It can determine all regulatory edges including their directions, but not autoregulation. Using extra information such as DNA sequence and transcription factor binding motifs, some regulatory edges can be excluded. Such prior knowledge can be incorporated by WENDY, but we first consider the original problem that any regulatory edge is possible (except autoregulation).

3.1 Mathematical model of gene expression

We start by building a model of gene expression for a single cell. Although various factors can affect gene expression, we only study regulations between genes. Assume there are n genes $V_1, \ldots, V_n$, which form a GRN. We assume that no other genes affect $V_1, \ldots, V_n$, meaning that there is no hidden variable. Denote the expression levels of $V_1, \ldots, V_n$ by $X_1, \ldots, X_n$. A common model for the dynamics of X_i is a (stochastic) differential equation

$$\frac{\mathrm{d}X_i(t)}{\mathrm{d}t} = f(X_1, \ldots, X_n) + c_{1,i} - c_{2,i}X_i + \text{noise}. \tag{1}$$

Here $c_{1,i}$ means the basal synthesis rate of X_i, and $c_{2,i}X_i$ means the total degradation rate of X_i. The function f that reflects the interaction between genes can take any form with any number of unknown parameters. The noise term is generally Gaussian. We think that the random fluctuation is proportional to X_i, and decide to follow Pinna et al. (2010); Papili Gao et al. (2018) to set the noise term to be $\sigma_i X_i \mathrm{d}W_i(t)$, where σ_i is an unknown constant, and $W_i(t)$ is a standard Brownian motion.

Since f in Eq. 1 is an arbitrary function, it is impossible to determine its parameters. We need to add restrictions on the form of f. Besides single gene regulations $V_j \to V_i$, transcription factors can bind to enhancers, whose signals are transmitted by coactivators (e.g., mediators) to promoters, which can recruit additional transcription factors and regulate the expression of one or many genes (Smith et al., 2023; Kamal et al., 2023). Regulation of gene expression by enhancers can occur over long distances, and in some instances, on different chromosomes. Some enhancers can regulate more than 1 gene (Bravo González-Blas et al., 2023). Therefore, to model the complicated gene expression, at least we need to consider cooperative regulations with two genes, $V_j + V_k \to V_i$. This means that Eq. 1 becomes

$$\frac{\mathrm{d}X_i(t)}{\mathrm{d}t} = \sum_{j=1}^{n} a_{j\to i} g(X_j) + \sum_{j<k} b_{j,k\to i} h(X_j, X_k) + c_{1,i} - c_{2,i}X_i + \sigma_i X_i \mathrm{d}W_i(t). \tag{2}$$

Here g and h are known functions (not necessarily linear), and $a_{j\to i}, b_{j,k\to i}$ are regulation coefficients.

To determine the GRN, we need to determine all coefficients $a_{j\to i}, b_{j,k\to i}$, whose total number is $n^3/2 + n^2/2$ for n genes. For single-cell gene expression data, at each time point, the data set is a matrix of m cells $\times$ n genes. From the data, we can estimate the parameters of the joint distribution of n genes. If this joint distribution is Gaussian, we can obtain n parameters for the mean, and $n(n+1)/2$ parameters for the (co)variance (since the covariance matrix is symmetric), which fully determine the distribution. Even though the joint distribution of n genes might not be Gaussian in reality, the number of parameters that can be estimated from expression data at each time point should be at the level of $n^2/2$. Therefore, to determine $n^3/2 + n^2/2$ GRN parameters in Eq. 2, we need at least n time points. This generally does not hold in reality, since a small GRN of interest might contain tens of genes, but the number of time points for type 9 data is commonly only a few (Hayashi et al., 2018; Treutlein et al., 2016; Chu et al., 2016). Thus we have to further (over)simplify Eq. 2 and drop cooperative terms:

$$\frac{\mathrm{d}X_i(t)}{\mathrm{d}t} = \sum_{j=1}^{n} a_{j\to i} g(X_j) + c_{1,i} - c_{2,i}X_i + \sigma_i X_i \mathrm{d}W_i(t). \tag{3}$$

Now there are only n^2 GRN parameters, and theoretically, single-cell data at only 2 time points can provide enough information. For instance, we can calculate the covariance

matrices of $X_1, \ldots, X_n$ at time 0 and time t, and determine what $a_{j\to i}$ in Eq. 3 can lead to such covariance matrices. This is an inverse problem of Eq. 3, which is much harder than the original problem.

In some models, the function g is nonlinear, meaning that Eq. 3 cannot be solved analytically. Thus, it is very difficult even if we only want to solve the inverse problem numerically, especially if we want the solver to be numerically stable. We need to further (over)simplify the model to assume that it is linear:

$$\frac{\mathrm{d}X_i(t)}{\mathrm{d}t} = \sum_{j=1}^{n} a_{j,i}X_j + c_i + \sigma_i X_i \mathrm{d}W_i(t). \tag{4}$$

Since Eq. 4 is linear, $a_{i\to i}g(X_i)$ (represents autoregulation) and $-c_{2,i}X_i$ (represents degradation) are combined to $a_{i,i}X_i$, while $a_{j,i} = a_{j\to i}$. This means that we cannot determine the existence of autoregulation, even if we can solve $a_{i,i}$. Since the autoregulation of V_i is masked by the natural synthesis and degradation of V_i, most GRN inference methods cannot determine the existence of autoregulation.

We combine Eq. 4 for different V_i to obtain

$$\frac{\mathrm{d}X(t)}{\mathrm{d}t} = X(t)A + c + X(t) \odot \mathrm{d}\sigma W(t). \tag{5}$$

Here $X(t) = [X_1(t), \ldots, X_n(t)]$, $c = [c_1, \ldots, c_n]$, and $\odot$ is the entrywise (Hadamard) product:

$$X(t) \odot \mathrm{d}\sigma W(t) = [X_1(t)\sigma_1 \mathrm{d}W_1(t), \ldots, X_n(t)\sigma_n \mathrm{d}W_n(t)].$$

If we directly solve the covariance matrix of $X_1(t), \ldots, X_n(t)$ from Eq. 5, then it has an integral that hinders the following optimization procedure. Thus we first ignore the noise term in Eq. 5, and its solution is

$$X(t) = [X(0) + cA^{-1}]e^{tA} - cA^{-1}.$$

Since e^{tA} is still difficult to handle in the following optimization procedure, we consider the first-order approximation

$$X(t) = X(0)(I + tA) + tc, \tag{6}$$

where I is the $n \times n$ identity matrix. Then we add back the integrated noise term, which is then approximately

$$X(t) = X(0)(I + tA) + tc + X(0) \odot \epsilon(t), \tag{7}$$

where $\epsilon(t) = [\epsilon_1(t), \ldots, \epsilon_n(t)]$ is an n-dimensional normal random noise with mean $(0, \ldots, 0)$ and covariance matrix G. Here we assume that G is diagonal, meaning that noise terms

$\epsilon_i(t), \epsilon_j(t)$ for different genes are independent, but the diagonal elements of G are unknown. Besides, $\epsilon(t)$ and $X(0)$ are also independent.

For type 9 data, after adding drugs or other interventions, we measure the expression levels of n genes at time 0, and the expression levels are n random variables: $X(0) = [X_1(0), \ldots, X_n(0)]$. Then at time t, we measure the expression levels of these n genes again to get n random variables: $X(t) = [X_1(t), \ldots, X_n(t)]$. Since we add interventions to drive the system away from stationary, $X(0)$ and $X(t)$ are not identically distributed. However, we do not have the joint distribution of $X(0)$ and $X(t)$, meaning that we do not know which sample of $X(0)$ corresponds to which sample of $X(t)$. From such data, we want to solve A, an invertible $n \times n$ matrix that represents the GRN we want: $A_{i,j} > 0$ means gene i activates gene j; $A_{i,j} < 0$ means gene i inhibits gene j; and $A_{i,j} = 0$ means gene i does not regulate gene j directly. However, as discussed above, $A_{i,i}$ is the combination of degradation and possibly autoregulation of gene i, and $A_{i,i} \neq 0$ does not necessarily mean autoregulation of gene i. In the next subsection, we will present a numerical method that calculates A in Eq. 7.

Gene expression has many biochemical subtleties, and Eq. 7 is certainly an oversimplification of Eq. 1. Here the simplification from Eq. 1 to Eq. 3 is inevitable, since type 9 data generally do not have many different time points, and cannot provide enough information to solve a model with too many unknown parameters (as in Eq. 2). If we want to learn some knowledge about the GRN from type 9 data, some simplification is necessary. The simplifications from Eq. 3 to Eq. 7 are only for numerical purposes. In Subsection 4.3, We will see that although our method is derived from Eq. 7, it has good performance on data generated by Eq. 11, which is in the form of Eq. 3. Therefore, the simplification from Eq. 3 to Eq. 7 does not harm the generalizability of our method. Besides, although our method cannot determine autoregulation $V_i \to V_i$, it can determine feedback loop $V_i \to V_j \to V_i$, which is the fundamental mechanism of some "autoregulations" observed in experiments (Crews and Pearson, 2009).

3.2 Dynamics of covariance matrix

Since we do not know the joint distribution of $X(0)$ and $X(t)$, we cannot determine which sample of $X(0)$ corresponds to which sample of $X(t)$, meaning that directly attacking Eq. 7 does not help. Instead, we can assume that some statistical quantities on both sides should be equal.

If we take expectation on Eq. 7, it returns to Eq. 6. For any A, we can find the value of c to make Eq. 6 hold. Thus we cannot solve A from Eq. 6. Instead, we can consider the

covariance matrices of $X(0)$ and $X(t)$, denoted as $K(0)$ and $K(t)$. We have

$$\begin{aligned}
K(t) =& \mathbb{E}\{[X(t)^{\mathrm{T}} - x(t)^{\mathrm{T}}][X(t) - x(t)]\} \\
=& \mathbb{E}\{(I + tA^{\mathrm{T}})[X(0)^{\mathrm{T}} - x(0)^{\mathrm{T}}][X(0) - x(0)](I + tA)\} \\
&+ \mathbb{E}\{(I + tA^{\mathrm{T}})[X(0)^{\mathrm{T}} - x(0)^{\mathrm{T}}][X(0) \odot \epsilon(t)]\} \\
&+ \mathbb{E}\{[X(0) \odot \epsilon(t)]^{\mathrm{T}}[X(0) - x(0)](I + tA)\} \\
&+ \mathbb{E}\{[X(0) \odot \epsilon(t)]^{\mathrm{T}}[X(0) \odot \epsilon(t)]\} \\
=& (I + tA^{\mathrm{T}})\mathbb{E}\{[X(0)^{\mathrm{T}} - x(0)^{\mathrm{T}}][X(0) - x(0)]\}(I + tA) \\
&+ \mathbb{E}\{(I + tA^{\mathrm{T}})[X(0)^{\mathrm{T}} - x(0)^{\mathrm{T}}][X(0) \odot \mathbb{E}\epsilon(t)]\} \\
&+ \mathbb{E}\{[X(0) \odot \mathbb{E}\epsilon(t)]^{\mathrm{T}}[X(0) - x(0)](I + tA)\} \\
&+ \mathbb{E}[X(0)^{\mathrm{T}}X(0)] \odot \mathbb{E}[\epsilon(t)^{\mathrm{T}}\epsilon(t)] \\
=& (I + tA^{\mathrm{T}})K(0)(I + tA) + D.
\end{aligned} \tag{8}$$

Here D is diagonal, with $D_{i,i} = \mathbb{E}[X_i(0)^2]G_{i,i}$.

Given $K(0)$ and $K(t)$, we cannot solve A directly from Eq. 8, even if we set $D = 0$. Assume $K(0)$ and $K(t)$ are invertible. As covariance matrices, they are positive-definite and have Cholesky decomposition $K(0) = L_0^{\mathrm{T}}L_0$ and $K(t) = L_1^{\mathrm{T}}L_1$ with upper-triangular L_0 and L_1. Then for any orthonormal matrix O with $O^{\mathrm{T}}O = I$, $A = (L_0^{-1}OL_1 - I)/t$ is a solution of Eq. 8 with $D = 0$. Thus Eq. 8 has infinitely many solutions in this case, and we need to add some conditions to obtain a unique A.

3.3 Optimization formulation for covariance dynamics

Assume we measure the expression levels of n genes for m single cells. When $m < n$, which is common in reality, if we directly calculate the covariance matrix, it will always be degenerate (non-invertible). Therefore, we need to apply a specific method, called graphical lasso, that estimates the covariance matrix K in this case, where K is invertible, and the inverse of K is sparse (Friedman et al., 2008).

For a $1 \times n$-dimensional random vector $N = [N_1, \ldots, N_n]$ with invertible covariance matrix K, there is a result that $K_{i,j}^{-1} = 0$ if and only if the partial Pearson correlation coefficient satisfies (Wasserman, 2004)

$$\rho_{i,j|1,\ldots,i-1,i+1,\ldots,j-1,j+1,\ldots,n} = 0.$$

Therefore, if N is multivariate normal, then $K_{i,j}^{-1} = 0$ if and only if N_i and N_j are independent conditioned on other variables.

By assuming the expression levels of n genes satisfy a multivariate normal distribution, there is a GRN inference method (Nouri et al., 2023): gene i and gene j have a direct regulatory relation (direction unknown) if and only if $K_{i,j}^{-1} \neq 0$. Nevertheless, this method assumes that the gene expression is in stationary.

For the data set we consider, the intervention might change the dynamics, and the gene expression is not in stationary. Therefore, $K_{i,j}^{-1} \neq 0$ might just mean that gene i and gene j has a direct regulatory relation before the intervention, not necessarily implying $A_{i,j} \neq 0$. However, the inverse should be true: if $K(0)_{i,j}^{-1} = 0$ and $K(t)_{i,j}^{-1} = 0$, then gene i and gene j should have no direct regulatory relation, whether before intervention or after intervention. Define $\mathcal{C} = \{(i,j) \mid K(0)_{i,j}^{-1} = 0 \text{ and } K(t)_{i,j}^{-1} = 0\}$. Then $A_{i,j} = 0$ if $(i,j) \in \mathcal{C}$. Since $K(0)^{-1}$ and $K(t)^{-1}$ are symmetric, $\mathcal{C}$ is also symmetric: $(i,j) \in \mathcal{C}$ implies $(j,i) \in \mathcal{C}$. Besides, since $K(0)^{-1}$ and $K(t)^{-1}$ are sparse, $\mathcal{C}$ contains most edges.

Certain data, such as DNA sequence and transcription factor binding motifs and ATAC-seq data, can provide prior knowledge that gene i cannot regulate gene j, meaning that $A_{i,j} = 0$. We denote the set of such forbidden edges as $\mathcal{F}$. Notice that $\mathcal{F}$ might not be symmetric.

From the data, after estimating the covariance matrices, we obtain invertible covariance matrices $K(0)$ and $K(t)$, where $K(0)^{-1}$ and $K(t)^{-1}$ are sparse. Now we have

$$K(t) - (I + tA^{\mathrm{T}})K(0)(I + tA) = D. \tag{9}$$

Since the diagonal matrix D is unknown, we only want to match off-diagonal elements of $K(t)$ and $(I + tA^{\mathrm{T}})K(0)(I + tA)$. Therefore, we need to solve A from Eq. 9 regardless of diagonal elements, so that $(I + tA)_{i,j} = 0$ whenever $(i,j) \in \mathcal{C} \cup \mathcal{F}$. Under this restriction, there might not be a solution. Instead, we can minimize the matching error and turn it into an optimization problem:

$$\min_A f_\lambda(A) := \frac{1}{2} \sum_{i \neq j} \{[K(t) - (I + tA^{\mathrm{T}})K(0)(I + tA)]_{i,j}\}^2 + \lambda A_{i,j}^2, \tag{10}$$

where $A_{i,j} = 0$ whenever $(i,j) \in \mathcal{C} \cup \mathcal{F}$ or $i = j$, while $\lambda \geq 0$ is a predetermined constant. The constraints are handled by only optimizing over the nonzero edges, and thus (10) simplifies to a (possibly) regularized nonlinear least squares problem. We set $\lambda = 0$ in the following simulations, but allow users to adjust λ if necessary.

Since $\mathcal{C} \cup \mathcal{F}$ contains most edges, the final A is sparse, which is biologically favorable, since we do not want a very dense GRN. If calculated $A_{i,j} > 0$ or $A_{i,j} < 0$ $(i \neq j)$, we claim that gene i regulates (activates/inhibits) gene j. The diagonal elements of A do not provide information about gene regulation, as we cannot distinguish between autoregulation and normal gene expression.

We use the BFGS algorithm to minimize (10). The overall algorithm is presented in Algorithm 1. For the WENDY method, step 4 of Algorithm 1 is much faster than step 2, since the solver terminates after a constant number of iterations. Graphical lasso has time complexity $\mathcal{O}(n^3)$ (Friedman et al., 2008), which does not depend on the cell number m. Therefore, the overall time complexity of WENDY method is $\mathcal{O}(n^3)$. In Section 4, we will see that in practice, the time cost of WENDY increases with n, but not m.

We present the workflow of WENDY in Algorithm 1. See

`https://github.com/zhengp0/genet`

and

`https://github.com/YueWangMathbio/WENDY`

for the Python implementation of WENDY.

Algorithm 1: Detailed workflow of WENDY method.

---

1. **Input**:

   Single-cell gene expression data at $T = 0$ (for m_1 cells and n genes) and at $T = t$ (for m_2 cells and the same n genes), both measured after general interventions. Prior knowledge of forbidden edges, $\mathcal{F}$

2. **Call** graphical lasso method to calculate the covariance matrix $K(0)$ for expression data at $T = 0$, so that $K(0)$ is invertible, and $K(0)^{-1}$ is sparse. Also calculate the covariance matrix $K(t)$ for expression data at $T = t$

3. **Construct** $\mathcal{C} = \{(i, j) \mid K(0)^{-1}_{i,j} = 0 \text{ and } K(t)^{-1}_{i,j} = 0\}$.

4. **Call** BFGS solver for the optimization problem

$$\arg\min_{A} \ \frac{1}{2} \sum_{i \neq j} \{[K(t) - (I + tA^{\mathrm{T}})K(0)(I + tA)]^2_{i,j} + \lambda A^2_{i,j}\}$$

   with constraints $A_{i,j} = 0$, $\forall (i, j) \in \mathcal{C} \cup \mathcal{F}$

5. **Output**:

   The GRN matrix A. $A_{i,j} > 0$ or $A_{i,j} < 0$ means gene i activates/inhibits gene j. However, $A_{i,i} \neq 0$ does not necessarily mean autoregulation of gene i

---

3.4 Theoretical comparison with other methods

To infer GRN structure in Scenario 9, besides WENDY, we have four other options: (I) apply SINCERITIES method; (II) calculate the average gene expression levels over all cells to transform the data into Scenario 8, and apply corresponding methods; (III) only consider the data at one time point, which is Scenario 2, and apply corresponding methods; (IV) treat the data at each time point as Scenario 2, and apply corresponding methods, but calibrate the inferred GRN by the GRN inferred for the previous time point, similar to Harissa and CARDAMOM.

(I, II) For an expression level data set with n genes at T time points, SINCERITIES only extracts $n(T-1)$ values, and then applies linear regression. If we transform the data into Scenario 8, we can only obtain nT values before fitting to an ODE system. If we abandon data at other time points to switch to Scenario 2, we lose the temporal information. In comparison, WENDY extracts $n^2T/2+nT/2$ values from such data before proceeding to the next step. Therefore, WENDY can extract more information from data. SINCERITIES requires at least $T=4$ time points to work normally (with 3 time points, SINCERITIES always produces an all-zero matrix), and requires at least $T=6$ time points to perform well. Methods for Scenario 8 need at least $T=3$ time points to work, and also more time points to work well. In comparison, WENDY only needs $T=2$ time points. This can be explained by information theory: the GRN (without autoregulation) has n^2-n independent values. WENDY uses data at $T=2$ time points to extract n^2+n values; SINCERITIES needs data at $T=n$ time points to extract n^2-n values; methods for Scenario 8 need data at $T=n-1$ time points to extract n^2-n values.

SINCERITIES and methods for Scenario 8 require more time points. This not only increases the cost, but also requires that the GRN does not change among such time points. This might not hold in reality, such as in the early development of embryos, where the regulatory strength might change rapidly. Also, when studying the effect of drugs on gene expression, the regulatory effect of drugs will gradually disappear. Instead, since WENDY only needs two time points, when there are data from more time points, we can apply WENDY for each pair of adjacent time points, and study how the GRN evolves along time.

(III) The theoretical foundation of some methods for Scenario 2 requires that the gene expression is at stationary. For Scenario 9, after adding the intervention, we need to wait for the gene expression to return to stationary state. However, in some experiments, the intervention (e.g., adding certain drugs) can drive the gene expression too far away from stationary, which leads to the death of cells before returning to stationary. Therefore, WENDY is a better solution to this transient scenario.

(IV) Harissa and CARDAMOM (Herbach, 2023; Ventre, 2021) have the same idea: for Scenario 2 data, build a nonlinear regression model under complicated pre-processing. Given the single-cell expression data D_t at time t, they can calculate the regression error $R(D_t, A_t)$ for any possible GRN A_t. Different from (III), they do not directly minimize $R(D_t, A_t)$, but add a calibration term that prevent the current inferred GRN A_t from being too different with the inferred GRN A_{t-1} for the previous time point $t-1$:

$$A_t = \arg\min_A R(D_t, A) + ||A - A_{t-1}||_1.$$

When $t=1$, set A_0 to be the zero matrix. For Scenario 9 data at time points $1, 2, \ldots, T$, use the last inferred GRN A_T as the final result. This idea can partially utilize the time information, but most information between two neighboring time points is wasted. For the situation that the GRN does not change along time, this idea relies heavily on data at later

time points, and wastes the data at early time points. For the situation that the GRN can change along time, this idea only reflects the GRN at the last time point, but not GRN at earlier time points. Since these two methods are based on a piecewise-deterministic Markov process model, not the stochastic differential equation model that generate the testing data for the next section, we will not test their performance.

4 Performance evaluation on synthetic data

In this section, we use two synthetic data sets with different settings to test the performance of WENDY and four other GRN inference methods that work for different data types: GENIE3, dynGENIE3, NonlinearODEs, SINCERITIES. WENDY ranks second for the DREAM4 data set, and ranks third for the SINC data set.

4.1 Methods and measurements

GENIE3 (Huynh-Thu et al., 2010) works for Scenario 2, single-cell level one-time expression data at stationary. The idea is to infer the level of one gene by the levels of other genes using random forest or extra trees. All edges can be inferred, but sometimes the direction is hard to determine.

dynGENIE3 (Huynh-Thu and Geurts, 2018) is the revised version of GENIE3 that works for time series data, such as Scenario 5 (and 10), single-cell level time series expression data at stationary or after general interventions. The idea is basically the same as GENIE3, but here it infers the level of one gene at a later time by the levels of other genes at an earlier time. Therefore, all edges including the directions can be determined. dynGENIE3 requires the correspondence of cells at different time points. This means that it cannot be applied to Scenario 9 data directly. Therefore, dynGENIE3 is included just for completeness, not as a main comparison target.

NonlinearODEs (Ma et al., 2020) works for Scenario 8, bulk level time series expression data measured after general interventions. The idea is to fit the data to a nonlinear ODE model. It uses XGBoost to determine the importance score of each edge. There is a tunable parameter α that can be chosen manually or automatically, and we use the *from_data* mode to determine the parameter α automatically. All edges including the directions can be determined.

SINCERITIES (Papili Gao et al., 2018) works for Scenario 9, single-cell level time series expression data measured after general interventions, where the joint distribution at different time points is unknown. The idea is to calculate the distance between the distributions of the same gene at two time points, and then applies linear regression. All edges including the directions can be determined.

Our WENDY method also works for Scenario 9. All edges including the directions can be determined. For the data sets in this section, there is no extra biological knowledge about regulations. Therefore, the set of forbidden edges $\mathcal{F} = \emptyset$.

To test a GRN inference method under different circumstances, a common practice is to use synthetic data (Papili Gao et al., 2018; Ma et al., 2020; Huynh-Thu and Geurts, 2018). The data are generally generated by numerically simulating an SDE system. We will test different inference methods on two data sets: DREAM4 and SINC.

Each inference method obtains a calculated GRN matrix A', whose entries take values in $\mathbb{R}$. To generate the synthetic data, there is a true GRN matrix A, which can take values $1/0/-1$. To evaluate the inference result A' with the true GRN matrix A, we calculate two measurements, AUROC and AUPR (Papili Gao et al., 2018). AUROC is the area under the curve of true positive rate versus false positive rate, and AUPR is the area under the curve of precision versus recall. They evaluate how the inferred GRN fits the true GRN from different perspectives. Since $A'_{i,i} \neq 0$ does not necessarily mean autoregulation, we do not compare the diagonal elements of A and A'.

AUROC and AUPR are originally defined for binary classification problems, where the ground truth can only take two values (e.g., 1 and 0), but the true GRN in our problem can take three values. We need to generalize the definitions of AUROC and AUPR. AUROC has another equivalent definition: for all data point pairs (i, j) and (p, q) with $A_{i,j} \neq A_{p,q}$, AUROC is the proportion of data point pairs that $(A_{i,j} - A'_{i,j})(A_{p,q} - A'_{p,q}) > 0$. This definition works for our ternary case. Notice that this ternary definition of AUROC is equivalent to calculate the binary AUROC for data points with $A_{i,j} = 1$ or $A_{p,q} = 0$, and weight it by the number of data point pairs $A_{i,j} = 1, A_{p,q} = 0$, and repeat this to $A_{i,j} = 1, A_{p,q} = -1$ and $A_{i,j} = 0, A_{p,q} = -1$. Then AUPR can be defined similarly. See Algorithm 2 for the calculation procedure of such quantities. When the true GRN can only take two values, these ternary definitions degenerate back to the original binary AUROC and AUPR. Besides, these ternary definitions also lead to values between 0 and 1, where 1 means perfect ordering.

4.2 DREAM4 data set

The most common synthetic data set used to evaluate GRN inference methods is DREAM4 – In Silico Network Challenge (Marbach et al., 2012). It has multiple challenges, each with multiple data types. We will consider the two in silico challenges and use only the time series data set. There are 5 GRNs with 10 genes, each accompanied by 5 stochastic trajectories at 21 time points. There are also 5 GRNs with 100 genes, each accompanied by 10 stochastic trajectories at 21 time points. The time points are equally distributed: 0, 50, 100, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000. Each GRN is represented by a matrix A, where $A_{i,j} = 1$ or $A_{i,j} = 0$, which means that gene i can or cannot regulate gene j. Also, $A_{i,i} = 0$ for each gene i. These two data sets are denoted as DREAM4 (10 genes) and DREAM4 (100 genes). DREAM4 data are generated by GeneNetWeaver software (Schaffter et al., 2011), which integrates some unknown SDE system.

We apply WENDY, GENIE3, dynGENIE3, NonlinearODEs, and SINCERITIES meth-

Algorithm 2: Calculation procedure of AUROC and AUPR.

1. **Input**: true GRN matrix A and calculated GRN matrix A', both with size $n \times n$. Here $A_{i,j}$ can take values $1/0/-1$, and $A'_{i,j} \in \mathbb{R}$

2. **Setup**:

   $\mathcal{U} = \{(i,j) \mid A_{i,j} = 1, i \neq j\}$, $u = |\mathcal{U}|$

   $\mathcal{Z} = \{(i,j) \mid A_{i,j} = 0, i \neq j\}$, $z = |\mathcal{Z}|$

   $\mathcal{D} = \{(i,j) \mid A_{i,j} = -1, i \neq j\}$, $d = |\mathcal{D}|$

3. **Calculate**

   $\text{AUROC}^{\text{binary}}_{1/0}$ and $\text{AUPR}^{\text{binary}}_{1/0}$ for data points in $\mathcal{U}$ and $\mathcal{Z}$

   $\text{AUROC}^{\text{binary}}_{1/-0}$ and $\text{AUPR}^{\text{binary}}_{1/-0}$ for data points in $\mathcal{U}$ and $\mathcal{D}$

   $\text{AUROC}^{\text{binary}}_{0/-1}$ and $\text{AUPR}^{\text{binary}}_{0/-1}$ for data points in $\mathcal{Z}$ and $\mathcal{D}$

4. **Average**

$$\text{AUROC}^{\text{ternary}} = \frac{uz\ \text{AUROC}^{\text{binary}}_{1/0}}{uz + ud + zd} + \frac{ud\ \text{AUROC}^{\text{binary}}_{1/-1}}{uz + ud + zd} + \frac{zd\ \text{AUROC}^{\text{binary}}_{0/-1}}{uz + ud + zd}$$

$$\text{AUPR}^{\text{ternary}} = \frac{uz\ \text{AUPR}^{\text{binary}}_{1/0}}{uz + ud + zd} + \frac{ud\ \text{AUPR}^{\text{binary}}_{1/-1}}{uz + ud + zd} + \frac{zd\ \text{AUPR}^{\text{binary}}_{0/-1}}{uz + ud + zd}$$

5. **Output**: AUROC and AUPR

ods to DREAM4 data sets. To test the performance of different methods on DREAM4 data, we compare different settings of the same method and choose the best one. Specifically, we find that for methods that can use data from multiple time points, it is not always good to use data from all 21 time points. Instead, data from some time points should be abandoned. Here we list the best settings for each method on each data set:
(1) For WENDY, we regard DREAM4 data as Scenario 9 data. For DREAM4 data (10 genes), we use the data at $t = 450$ and $t = 850$ without the cell correspondence between different time points. For DREAM4 data (100 genes), we use the data at $t = 300$ and $t = 550$ without the cell correspondence between different time points.
(2) For SINCERITIES, we regard DREAM4 data as Scenario 9 data by ignoring the cell correspondence between different time points. For DREAM4 data (10 genes), we use the data at 8 time points: t =(0, 50, 100, 150, 200, 250, 300, 350). For DREAM4 data (100 genes), we use the data at 5 time points: t =(750, 800, 850, 900, 950).
(3) For NonlinearODEs, we regard DREAM4 data as Scenario 8 data. For DREAM4 data (10 genes), we use the data at 3 time points: t =(300, 350, 400). For DREAM4 data (100 genes), we use the data at 8 time points: t =(200, 250, 300, 350, 400, 450, 500, 550).
(4) For GENIE3, we regard DREAM4 data as Scenario 2 data by considering only one time point. For DREAM4 data (10 genes), we use the data at t =450. For DREAM4 data (100 genes), we use the data at t =350.
(5) For dynGENIE3, we regard DREAM4 data as Scenario 10 data. For DREAM4 data (10 genes), we use the data at 18 time points: t =(0, 50, 100, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850). For DREAM4 data (100 genes), we use the data at 17 time points: t =(0, 50, 100, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800).

See Table 2 for the results, where the AUROC and AUPR are averaged over all 5 GRNs in the same data set. We can see that on average, WENDY is slightly better than NonlinearODEs and GENIE3, which are significantly better than SINCERITIES, while dynGENIE3 is significantly better than all other methods. This is not surprising, since dynGENIE3 can utilize the cell correspondence (joint distribution) of different time points, which is not realistic, but contains more information.

Besides, some results are lower than those values reported in corresponding papers (Huynh-Thu et al., 2010; Huynh-Thu and Geurts, 2018; Ma et al., 2020), since we only use the time series data, not combining with other data types, and we do not manually fine-tune the parameters accordingly.

One problem of the DREAM4 data is that each GRN only generates 5 or 10 stochastic trajectories (each corresponds to a measured cell population). This fits with the mainstream of bulk level data in the early 2010s, when it was difficult to repeat the measurement many times. Nevertheless, with the development of single-cell RNA sequencing, it is easier to obtain single-cell data from thousands of cells. Therefore, we want to generate single-cell expression data over more cells and test the performance of inference methods under different cell numbers.

Table 2: AUROC and AUPR of different methods on DREAM4 data sets

		WENDY	SINCE-RITIES	Nonline-arODEs	GENIE3	dynG-ENIE3
DREAM4	AUROC	0.65	0.52	0.64	0.64	0.75
(10 genes)	AUPR	0.34	0.19	0.32	0.27	0.54
DREAM4	AUROC	0.59	0.55	0.57	0.64	0.74
(100 genes)	AUPR	0.05	0.03	0.04	0.05	0.14
	Total	1.63	1.29	1.57	1.60	2.17

4.3 SINC data set

Pinna et al. (2010) and Papili Gao et al. (2018) use the following SDE to generate synthetic data:

$$\mathrm{d}X_j(t) = V\left\{\beta\prod_{i=1}^{n}\left[1 + A_{i,j}\frac{X_i(t)}{X_i(t)+1}\right] - \theta X_j(t)\right\}\mathrm{d}t + \sigma X_j(t)\mathrm{d}W_j(t). \tag{11}$$

Here $X_i(t)$ is the expression level of gene i at time t, and $W_j(t)$ is a standard Brownian motion, independent with other $W_i(\tau)$. $A_{i,j}$ describes the GRN. $V = 30$, $\beta = 1$, $\theta = 0.2$, $\sigma = 0.1$. When $t \to \infty$, each $X_i(t)$ will converge to the stationary value and fluctuate slightly around it (Yang and Cheng, 2021; Cheng and Qian, 2021; Cheng et al., 2021).

The following equation is the first-order approximation of Eq. 11 when $X_i(t)$ is small:

$$\mathrm{d}X_j(t) = V\left\{\beta\left[1 + \sum_{i=1}^{n} A_{i,j}X_i(t)\right] - \theta X_j(t)\right\}\mathrm{d}t + \sigma X_j(t)\mathrm{d}W_j(t), \tag{12}$$

Also, Eq. 7 is the discretization of Eq. 12: A, b_i, $F_{i,i}$ in Eq. 7 correspond to $V\beta A - V\theta I$, $V\beta$, $\sigma^2 t$ in Eq. 12. These facts provide the theoretical support that WENDY (derived from Eq. 7) can work for data generated by Eq. 11, since we only care about off-diagonal elements of A, which represent mutual regulations between different genes. For other gene regulation mechanisms, we can also use first-order approximation and discretization to obtain Eq. 7 or similar forms. Therefore, WENDY should be applicable to different gene regulation mechanisms.

For Eq. 11, we simulate the system with the Euler-Maruyama method (Kloeden et al., 1992) for time $t \in [0, 1]$ with time step $\Delta t = 0.01$. This means treating $\mathrm{d}X_j(t)$ as $X_j(t + \Delta t) - X_j(t)$, $\mathrm{d}t$ as Δt, and $\mathrm{d}W_j(t)$ as a normal random variable $\mathcal{N}(0, \Delta t)$.

For the GRN matrix A, we use the 40 networks in Papili Gao et al. (2018): 10 *Escherichia coli* networks with 10 genes, 10 *E. coli* networks with 20 genes, 10 *Saccharomyces cerevisiae* (yeast) networks with 10 genes, 10 yeast networks with 20 genes. Each network has $A_{i,j} = 1/0/-1$, and $A_{i,i} = 0$ for each i.

For each group of data, we run the simulation m times, where m represents the number of cells/trajectories measured in reality. We set $m = 10$, $m = 30$, and $m = 100$ to test the performance of different methods under different m. The initial state $X_i(0)$ is independently and uniformly sampled in $[0, 1)$. For each network in Papili Gao et al.'s paper, and each value of m, we generate 100 groups of data. The same as Papili Gao et al.'s paper, the simulation finishes at $t = 3.0$. We record the expression levels at the following 11 time points: t =(0.0, 0.3, 0.6, 0.9, 1.2, 1.5, 1.8, 2.1, 2.4, 2.7, 3.0). For the data sets generated by Eq. 11 on GRNs with 10 or 20 genes, we name them SINC (10 or 20 genes) data. Since the initial state is generally different from the stationary state, SINC data should be regarded as Scenario 10 data (single-cell level time series data under general interventions, where the joint distribution of different time points is known).

To test the performance of different methods on SINC data, we compare different settings of the same method and choose the best one. For WENDY, we fix the first time point to be 0.0. For SINCERITIES, NonlinearODEs, and dynGENIE3, we consider contiguous time points that start from 0.0. Specifically, we find that for methods that can use data from multiple time points, it is not always good to use data from all 11 time points. Instead, data from some time points should be abandoned. Here we list the best settings for each method on each data set:
(1) For WENDY, we regard SINC data as Scenario 9 data. For both SINC data (10 genes)and SINC data (20 genes), we use the data at $t = 0.0$ and $t = 0.3$.
(2) For SINCERITIES, we regard SINC data as Scenario 9 data by ignoring the cell correspondence between different time points. For both SINC data (10 genes) and SINC data (20 genes), we use the data at 4 time points: t =(0.0, 0.3, 0.6, 0.9).
(3) For NonlinearODEs, we regard SINC data as Scenario 8 data by taking average of gene expression levels over different cells. For both SINC data (10 genes) and SINC data (20 genes), we use the data at 10 time points: t =(0.0, 0.3, 0.6, 0.9, 1.2, 1.5, 1.8, 2.1, 2.4, 2.7).
(4) For GENIE3, we regard SINC data as Scenario 2 data by considering only one time point. For both SINC data (10 genes) and SINC data (20 genes), we use the data at $t = 0.3$.
(5) For dynGENIE3, we regard SINC data as Scenario 10 data. For SINC data (10 genes), we use the data at 4 time points: t =(0.0, 0.3, 0.6, 0.9). For SINC data (20 genes), we use the data at 3 time points: t =(0.0, 0.3, 0.6).

We apply WENDY, SINCERITIES, NonlinearODEs, GENIE3, and dynGENIE3 to SINC data sets and compare the corresponding AUROC and AUPR. Each AUROC or AUPR value is averaged over 2000 simulations (20 GRNs, each with 100 groups of data). See Table 3 for the results. We can see that WENDY is better than SINCERITIES and NonlinearODEs, but weaker than GENIE3 and dynGENIE3 (which does not work for Scenario 9 data).

For different values of cell/trajectory number m, we can see that most performance metrics increase with m, meaning that more cells provide more information.

We also test the running time of each inference method. For SINC (10 genes) and

Table 3: AUROC and AUPR of different methods on SINC data sets

		WENDY	SINCE-RITIES	Nonline-arODEs	GENIE3	dynG-ENIE3
SINC $m = 10$	AUROC	0.51	0.53	0.49	0.60	0.56
(10 genes)	AUPR	0.33	0.32	0.34	0.38	0.36
SINC $m = 10$	AUROC	0.50	0.50	0.49	0.54	0.51
(20 genes)	AUPR	0.09	0.07	0.09	0.10	0.09
SINC $m = 30$	AUROC	0.52	0.53	0.49	0.64	0.59
(10 genes)	AUPR	0.33	0.32	0.34	0.40	0.37
SINC $m = 30$	AUROC	0.50	0.50	0.49	0.58	0.53
(20 genes)	AUPR	0.08	0.08	0.09	0.12	0.09
SINC $m = 100$	AUROC	0.58	0.49	0.50	0.70	0.61
(10 genes)	AUPR	0.35	0.32	0.34	0.44	0.37
SINC $m = 100$	AUROC	0.51	0.47	0.49	0.64	0.55
(20 genes)	AUPR	0.09	0.08	0.09	0.14	0.10
	Total	4.39	4.21	4.24	5.28	4.73

SINC (20 genes) data sets, we measure the execution time (averaged over different GRNs) of each algorithm for $m = 10$, $m = 30$, and $m = 100$ cells. See Table 4 for the running time (in the form of mean $\pm$ standard deviation) in different settings. We can see that for each algorithm, the time cost increases with the gene number n. In addition, WENDY, SINCERITIES and non-linearODEs are insensitive to the cell number m, while the time costs of GENIE3 and dynGENIE3 increase significantly with m. When m and n are large, WENDY is roughly the same fast as SINCERITIES and NonlinearODEs, but much faster than GENIE3 and dynGENIE3. Therefore, WENDY has a satisfactory speed.

Table 4: Computational time for different algorithms. We consider the SINC data of $n = 10/20$ genes over $m = 10$, $m = 30$, or $m = 100$ cells. For each situation, we use 20 different GRNs, and present the computational time as mean $\pm$ standard deviation. All time costs are in seconds, and are measured on a desktop with Intel i7-13700 CPU.

		WENDY	SINCE-RITIES	Nonline-arODEs	GENIE3	dynG-ENIE3
SINC	$m = 10$	0.22 ± 0.04	0.09 ± 0.01	0.19 ± 0.02	4.27 ± 0.06	7.28 ± 0.08
$n = 10$	$m = 30$	0.15 ± 0.03	0.11 ± 0.01	0.18 ± 0.02	4.83 ± 0.05	15.86 ± 0.17
genes	$m = 100$	0.14 ± 0.03	0.12 ± 0.01	0.19 ± 0.03	7.32 ± 0.09	51.79 ± 0.63
SINC	$m = 10$	0.38 ± 0.10	0.42 ± 0.01	0.35 ± 0.03	8.55 ± 0.05	16.24 ± 0.13
$n = 20$	$m = 30$	0.34 ± 0.03	0.45 ± 0.01	0.36 ± 0.08	9.95 ± 0.07	38.21 ± 0.28
genes	$m = 100$	0.27 ± 0.02	0.64 ± 0.01	0.34 ± 0.04	16.32 ± 0.17	131.89 ± 0.76

5 Performance evaluation on experimental data

Besides synthetic data, there are also some GRNs determined by experiments, so that we can use them as the ground truth for the corresponding gene expression data (Hu et al., 2007; Reimand et al., 2010; Peter and Davidson, 2015). In this section, we use two experimental data sets in Scenario 9 with known GRNs to test the performance of WENDY and three other GRN inference methods that work for different data types: GENIE3, NonlinearODEs, SINCERITIES. Since each cell is only measured once, we do not have the joint distribution of expression levels at different time points, and dynGENIE3 does not apply. WENDY ranks first for the THP-1 data set, and ranks second for the hESC data set.

5.1 THP-1 data set

THP-1 data set considers single-cell expression levels of monocytic THP-1 human myeloid leukemia cells, measured at 8 time points, $t = 0$, 1, 6, 12, 24, 48, 72, 96h, after applying phorbol-12-myristate-13-acetate (Kouno et al., 2013). For each time point, there are 120 cells measured. However, since each cell can be measured only once, we do not know how cells at different time points correspond to each other. Therefore, this data set is in Scenario 9.

The same as Papili Gao et al. (2018), we consider the following 20 genes: BCL6, CEBPB, CEBPD, EGR2, FLI1, HOXA10, HOXA13, IRF8, MAFB, MYB, NFATC1, NFE2L1, PPARD, PPARG, PRDM1, RUNX1, SNAI3, TCFL5, TFPT, UHRF1. For such genes in THP-1 cells, there has been a GRN determined by experiments (Tomaru et al., 2009; Vitezic et al., 2010) that we can use as the ground truth.

To test the performance of different methods on THP-1 data, we compare different settings of the same method and choose the best one. Specifically, we find that for methods that can use data from multiple time points, it is not always good to use data from all 8 time points. Instead, data from some time points should be abandoned. Here we list the best settings for each method:
(1) For WENDY, we regard THP-1 data as Scenario 9 data. We use the data at $t = 0$ and $t = 48$ without the cell correspondence between different time points.
(2) For SINCERITIES, we regard THP-1 data as Scenario 9 data. We use the data at 7 time points: $t =$(1, 6, 12, 24, 48, 72, 96).
(3) For NonlinearODEs, we regard THP-1 data as Scenario 8 data by taking average of gene expression levels over different cells. We use the data at 3 time points: $t =$(0, 1, 6).
(4) For GENIE3, we regard THP-1 data as Scenario 2 data by considering only one time point. We use the data at $t = 24$.

We apply WENDY, SINCERITIES, NonlinearODEs, and GENIE3 to THP-1 data set and compare the corresponding AUROC and AUPR. See Table 5 for the results. We can see that WENDY is better than all other methods. Here GENIE3 has the worst performance,

possibly because it cannot determine the sign (activation/inhibition) of a regulation. The results for SINCERITIES are slightly different from those reported by Papili Gao et al. (2018), possibly because we do not consider diagonal elements of the GRN in the calculation of AUROC and AUPR.

Table 5: AUROC and AUPR of different methods on THP-1 data set

	WENDY	SINCE-RITIES	Nonline-arODEs	GENIE3
AUROC	0.62	0.63	0.55	0.42
AUPR	0.51	0.43	0.39	0.36

5.2 hESC data set

hESC data set considers single-cell expression levels of human embryonic stem cell-derived progenitor cells, measured at 6 time points, t =0, 12, 24, 36, 72, 96h (Chu et al., 2016). For each time point, there are 66–172 cells measured. Such cells are under development (which can be regarded as an intervention), and the regulation mechanism of their gene expression is highly complicated (Li and Belmonte, 2017). The same as THP-1 data set, there is no cell correspondence between different time points, and it is in Scenario 9.

Matsumoto et al. (2017) calculated the mean expression level of each gene at each time point, and then calculated the variance of the mean expression levels at different time points for each gene. Using this approach, they selected out top 100 highly varying genes, which might actively regulate each other. Given the data of these 100 highly varying genes, we further select out genes that express in at least 95% of the cells. The reason is that when the expression data have too many 0s, graphical lasso might fail to converge, which fails WENDY. There are 18 genes selected: POU5F1, AEBP2, MIER1, SMAD2, ZNF652, ZFX, TERF1, SOX11, BBX, ZFP42, TULP4, ZNF471, ARID4B, ZNF483, SHOX, ZNF587, ZFP14, CEBPZ. For such genes, there has been a ground truth GRN determined by experiments (Neph et al., 2012; Stergachis et al., 2014) that we can use as the ground truth.

Similar to the THP-1 data set, we apply WENDY (use $t = 12, 24$ data), SINCERITIES (use $t = 0, 12, 24, 36, 72$ data), NonlinearODEs (use $t = 24, 36, 72, 96$ data), and GENIE3 (use $t = 24$ data) to hESC data set and compare the corresponding AUROC and AUPR. See Table 6 for the results. We can see that WENDY has the highest AUROC, but the total performance ranks second, worse than SINCERITIES.

6 Discussion

In this paper, we address the GRN inference problem for single-cell time series gene expression data following general interventions, where the joint distribution of different time

Table 6: AUROC and AUPR of different methods on hESC data set

	WENDY	SINCERITIES	Nonlinear ODEs	GENIE3
AUROC	0.70	0.67	0.61	0.64
AUPR	0.05	0.33	0.06	0.05

points is unknown. Although this type of data is common in recent experiments, there are few GRN inference methods that fully utilize the information contained in the data. Therefore, we introduce WENDY, a GRN inference method developed for single-cell gene expression data spanning two time points after an intervention. This method is capable of inferring all mutual regulatory relations, including direction. We test WENDY and other GRN inference methods on two synthetic data sets and two experimental data sets, and the performance of WENDY is satisfactory: it is only weaker than dynGENIE3, which does not work on Scenario 9 data. See Table 7 for a summary. Besides, the time cost of WENDY is almost as low as the fastest method.

Table 7: Ranking of five GRN inference methods on four data sets

	WENDY	SINCERITIES	Nonlinear ODEs	GENIE3	dynGENIE3
DREAM4	2nd	5th	4th	3rd	1st
SINC	3rd	5th	4th	1st	2nd
THP1	1st	2nd	3rd	4th	-
hESC	2nd	1st	4th	3rd	-

The model in Subsection 3.1 only considers cells of the same type and under the same environment. In reality, cells of a complex organism are under different regulations by environmental factors, especially during development. For instance, during animal morphogenesis, retinoic acid can form a concentration gradient (positional information) (Wang et al., 2020), and leads cells at different positions to different fates by regulating certain genes (Wolpert, 1989). If the levels of such spatially heterogeneous factors can be measured for each single cell, then one can add them into the GRN inference procedure. Otherwise, they become hidden variables in the gene expression model, and can make the inference much more difficult (Lo et al., 2015). Besides such position-related factors, cell type can affect gene regulation (Zhang et al., 2023). One should label the cell type during experiments or after experiments using cluster analysis (Bocci et al., 2022). If the GRN inference is directly applied to data from different types of cells, the inference result might be unreliable (Wang and He, 2023). In this work, we assume that cells are of the same type, and they are under the same environment. An important future direction is to build more

realistic gene expression models and develop corresponding GRN inference methods.

Although most gene expression models are similar to Eq. 1 that only considers the (mRNA) levels of $V_1, \ldots, V_n$, some researchers argue that the actual gene expression mechanism should be more complex. In some models (Bonnaffoux et al., 2019; Ventre, 2021; Herbach, 2023; Ventre et al., 2023), each gene can switch between "on" and "off" states, which correspond to different transcription rates. Some proteins can affect the transition rates between "on" and "off" states, which is the only way of gene mutual-regulation. When one gene is turned on, the number of its mRNA increases quickly, until it is turned off, when the mRNA count starts to decay exponentially. This leads to the mRNA bursts phenomenon (Elgart et al., 2011), meaning that the mRNA count does not fluctuate locally, but has global bursts. Since proteins are translated from mRNAs, proteins also have bursts. However, proteins degrade much slower than mRNAs, meaning that mRNA count and protein count do not match exactly. Therefore, in such models, besides the mRNA counts $X_1, \ldots, X_n$, there are also hidden variables for gene states and protein counts. However, more complicated models, generally using piecewise-deterministic Markov processes, are harder to solve, and researchers need to conduct large-scale simplifications. Besides, more complicated models need more accurate data, but the current mRNA measurement techniques are not very sensitive (Zhang et al., 2021b). Therefore, we think that it is worthwhile to study differential equation based models.

Similar to most other GRN inference methods, WENDY cannot infer autoregulation. One potential future direction is to develop an autoregulation inference method inspired by the principles of WENDY. For instance, we can choose a reasonable nonlinear gene expression model that allows autoregulation, and study whether autoregulation can make a difference for the dynamics of covariance matrices.

In interventional experiments, it is customary to measure expression levels before intervention as the control group. These control data align with Scenario 2 data, enabling the application of corresponding inference methods. Therefore, a comparison of GRNs before and after intervention can elucidate the effect of the intervention on gene regulation.

As of 2024, single-cell level gene expression measurement remains relatively insensitive. It is common to miss all mRNAs of one gene in a single cell, resulting in experimental data with many zeros. This characteristic may impede graphical lasso, and consequently, WENDY. A potential future direction involves developing more robust inference methods capable of handling data with numerous missing values.

Although WENDY aims for data type 9, other types of data can help to improve the results of WENDY. For instance, one can use a CRISPR gene knockout study to verify regulations inferred by WENDY. Besides, advancements in biotechnology introduce new types of data, which are not in our classification framework, but might be applicable to GRN inference, and can enhance WENDY's capabilities through incorporation.

In our tests, we observed that WENDY performs better for data in earlier time points when dealing with a small number of genes (10 or fewer). Conversely, for data collected long after intervention, where gene expression approaches a new steady state, the inference

results are less reliable compared to earlier time points. One possible reason is that after enough time, the covariance matrix approaches its steady state, and the small changes of covariances might be covered by random noise. Therefore, when applying WENDY to experimental data, caution should be exercised with results obtained several days after the intervention.

Theoretically, the GRN matrix A calculated by WENDY is not symmetric, allowing determination of the directions of the regulatory relations ($i \to j$ or $j \to i$). However, our simulations indicate that $A_{i,j}$ and $A_{j,i}$ generally exhibit proximity, resulting in a significant decrease in AUPR. A prospective avenue involves developing a novel solver capable of producing highly asymmetric results.

In our WENDY implementation, we employ the standard graphical lasso algorithm to determine the inverse of the covariance matrix. Other algorithms with similar purposes (Vinci et al., 2019; Danaher et al., 2014) can also be considered for integration into WENDY.

Acknowledgments

The authors would like to thank anonymous reviewers for helpful comments. YW would like to thank Ulysse Herbach and Joseph Zhou for helpful discussions.

Data and code availability

The solver used in WENDY method is in

`https://github.com/zhengp0/genet`

Main function of WENDY method (including a tutorial), and other data and code files used in this paper are in

`https://github.com/YueWangMathbio/WENDY`

References

Aalto, A., Viitasaari, L., Ilmonen, P., Mombaerts, L., and Gonçalves, J. (2020). Gene regulatory network inference from sparsely sampled noisy data. *Nature communications*, 11(1):3493.

Agostinho, N. B., Machado, K. S., and Werhli, A. V. (2015). Inference of regulatory networks with a convergence improved mcmc sampler. *BMC bioinformatics*, 16:1–10.

Atanackovic, L., Tong, A., Hartford, J., Lee, L. J., Wang, B., and Bengio, Y. (2023). Dyngfn: Bayesian dynamic causal discovery using generative flow networks. *arXiv preprint arXiv:2302.04178*.

Axelrod, S., Li, X., Lincoln, S., Terceros, A., O'Neil, J., Wang, Z., Nguyen, A., and Young, M. W. (2021). A role for the drosophila blood-brain barrier in the regulation of sleep. *Available at SSRN 3895317*.

Axelrod, S., Li, X., Sun, Y., Lincoln, S., Terceros, A., O'Neil, J., Wang, Z., Nguyen, A., Vora, A., Spicer, C., et al. (2023). The drosophila blood–brain barrier regulates sleep via moody g protein-coupled receptor signaling. *Proceedings of the National Academy of Sciences*, 120(42):e2309331120.

Badia-i Mompel, P., Wessels, L., Müller-Dott, S., Trimbour, R., Ramirez Flores, R. O., Argelaguet, R., and Saez-Rodriguez, J. (2023). Gene regulatory network inference in the era of single-cell multi-omics. *Nature Reviews Genetics*, pages 1–16.

Barbosa, S., Niebel, B., Wolf, S., Mauch, K., and Takors, R. (2018). A guide to gene regulatory network inference for obtaining predictive solutions: Underlying assumptions and fundamental biological and data constraints. *Biosystems*, 174:37–48.

Basso, K., Margolin, A. A., Stolovitzky, G., Klein, U., Dalla-Favera, R., and Califano, A. (2005). Reverse engineering of regulatory networks in human b cells. *Nature genetics*, 37(4):382–390.

Bocci, F., Zhou, P., and Nie, Q. (2022). splicejac: transition genes and state-specific gene regulation from single-cell transcriptome data. *Molecular systems biology*, 18(11):e11176.

Bonnaffoux, A., Herbach, U., Richard, A., Guillemin, A., Gonin-Giraud, S., Gros, P.-A., and Gandrillon, O. (2019). Wasabi: a dynamic iterative framework for gene regulatory network inference. *BMC bioinformatics*, 20:1–19.

Bravo González-Blas, C., De Winter, S., Hulselmans, G., Hecker, N., Matetovici, I., Christiaens, V., Poovathingal, S., Wouters, J., Aibar, S., and Aerts, S. (2023). Scenic+: single-cell multiomic inference of enhancers and gene regulatory networks. *Nature methods*, 20(9):1355–1367.

Burdziak, C., Zhao, C. J., Haviv, D., Alonso-Curbelo, D., Lowe, S. W., and Pe'er, D. (2023). sckinetics: inference of regulatory velocity with single-cell transcriptomics data. *Bioinformatics*, 39(Supplement_1):i394–i403.

Chakraborty, P., George, J. T., Woodward, W. A., Levine, H., and Jolly, M. K. (2021). Gene expression profiles of inflammatory breast cancer reveal high heterogeneity across the epithelial-hybrid-mesenchymal spectrum. *Translational oncology*, 14(4):101026.

Cheng, Y.-C. and Qian, H. (2021). Stochastic limit-cycle oscillations of a nonlinear system under random perturbations. *Journal of Statistical Physics*, 182(3):47.

Cheng, Y.-C., Qian, H., and Zhu, Y. (2021). Asymptotic behavior of a sequence of conditional probability distributions and the canonical ensemble. In *Annales Henri Poincaré*, volume 22, pages 1561–1627. Springer.

Cheng, Y.-C., Stein, S., Nardone, A., Liu, W., Ma, W., Cohen, G., Guarducci, C., McDonald, T. O., Jeselsohn, R., and Michor, F. (2023a). Mathematical modeling identifies optimum palbociclib-fulvestrant dose administration schedules for the treatment of patients with estrogen receptor–positive breast cancer. *Cancer Research Communications*, 3(11):2331–2344.

Cheng, Y.-C., Zhang, Y., Tripathi, S., BV, H., Jolly, M. K., Schiebinger, G., Levine, H., McDonald, T. O., and Michor, F. (2023b). Reconstruction of single cell lineage trajectories and identification of diversity in fates during the epithelial-to-mesenchymal transition. *bioRxiv*, pages 2023–09.

Cho, H., Berger, B., and Peng, J. (2016). Reconstructing causal biological networks through active learning. *PloS one*, 11(3):e0150611.

Chu, L.-F., Leng, N., Zhang, J., Hou, Z., Mamott, D., Vereide, D. T., Choi, J., Kendziorski, C., Stewart, R., and Thomson, J. A. (2016). Single-cell rna-seq reveals novel regulators of human embryonic stem cell differentiation to definitive endoderm. *Genome biology*, 17:1–20.

Crews, S. T. and Pearson, J. C. (2009). Transcriptional autoregulation in development. *Current Biology*, 19(6):R241–R246.

Danaher, P., Wang, P., and Witten, D. M. (2014). The joint graphical lasso for inverse covariance estimation across multiple classes. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 76(2):373–397.

Deshpande, A., Chu, L.-F., Stewart, R., and Gitter, A. (2022). Network inference with granger causality ensembles on single-cell transcriptomics. *Cell reports*, 38(6).

Elgart, V., Jia, T., Fenley, A. T., and Kulkarni, R. (2011). Connecting protein and mrna burst distributions for stochastic models of gene expression. *Physical biology*, 8(4):046001.

Erbe, R., Gore, J., Gemmill, K., Gaykalova, D. A., and Fertig, E. J. (2022). The use of machine learning to discover regulatory networks controlling biological systems. *Molecular cell*, 82(2):260–273.

Feigelman, J., Ganscha, S., Hastreiter, S., Schwarzfischer, M., Filipczyk, A., Schroeder, T., Theis, F. J., Marr, C., and Claassen, M. (2016). Analysis of cell lineage trees by exact bayesian inference identifies negative autoregulation of nanog in mouse embryonic stem cells. *Cell systems*, 3(5):480–490.

Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441.

Fujita, A., Sato, J. R., Garay-Malpartida, H. M., Morettin, P. A., Sogayar, M. C., and Ferreira, C. E. (2007a). Time-varying modeling of gene expression regulatory networks using the wavelet dynamic vector autoregressive method. *Bioinformatics*, 23(13):1623–1630.

Fujita, A., Sato, J. R., Garay-Malpartida, H. M., Yamaguchi, R., Miyano, S., Sogayar, M. C., and Ferreira, C. E. (2007b). Modeling gene expression regulatory networks with the sparse vector autoregressive model. *BMC systems biology*, 1:1–11.

Fujita, A., Sato, J. R., Kojima, K., Gomes, L. R., Nagasaki, M., Sogayar, M. C., and Miyano, S. (2010a). Identification of granger causality between gene sets. *Journal of Bioinformatics and Computational Biology*, 8(04):679–701.

Fujita, A., Severino, P., Sato, J. R., and Miyano, S. (2010b). Granger causality in systems biology: Modeling gene networks in time series microarray data using vector autoregressive models. In *Advances in Bioinformatics and Computational Biology: 5th Brazilian Symposium on Bioinformatics, BSB 2010, Rio de Janeiro, Brazil, August 31-September 3, 2010. Proceedings 5*, pages 13–24. Springer.

Ghosh, S. and Zhong, C. (2021). Convergence rate of a collapsed gibbs sampler for crossed random effects models. *arXiv preprint arXiv:2109.02849*.

Gustafsson, M., Hornquist, M., and Lombardi, A. (2005). Constructing and analyzing a large-scale gene-to-gene regulatory network lasso-constrained inference and biological validation. *IEEE/ACM Transactions on computational biology and bioinformatics*, 2(3):254–261.

Haury, A.-C., Mordelet, F., Vera-Licona, P., and Vert, J.-P. (2012). Tigress: trustful inference of gene regulation using stability selection. *BMC systems biology*, 6(1):1–17.

Hayashi, T., Ozaki, H., Sasagawa, Y., Umeda, M., Danno, H., and Nikaido, I. (2018). Single-cell full-length total rna sequencing uncovers dynamics of recursive splicing and enhancer rnas. *Nature communications*, 9(1):619.

Herbach, U. (2023). Harissa: stochastic simulation and inference of gene regulatory networks based on transcriptional bursting. In *International Conference on Computational Methods in Systems Biology*, pages 97–105. Springer.

Hu, Z., Killion, P. J., and Iyer, V. R. (2007). Genetic reconstruction of a functional transcriptional regulatory network. *Nature genetics*, 39(5):683–687.

Huynh-Thu, V. A. and Geurts, P. (2018). dyngenie3: dynamical genie3 for the inference of gene networks from time series expression data. *Scientific reports*, 8(1):3384.

Huynh-Thu, V. A., Irrthum, A., Wehenkel, L., and Geurts, P. (2010). Inferring regulatory networks from expression data using tree-based methods. *PloS one*, 5(9):e12776.

Huynh-Thu, V. A. and Sanguinetti, G. (2015). Combining tree-based and dynamical systems for the inference of gene regulatory networks. *Bioinformatics*, 31(10):1614–1622.

Kamal, A., Arnold, C., Claringbould, A., Moussa, R., Servaas, N. H., Kholmatov, M., Daga, N., Nogina, D., Mueller-Dott, S., Reyes-Palomares, A., et al. (2023). Granie and granpa: inference and evaluation of enhancer-mediated gene regulatory networks. *Molecular Systems Biology*, 19(6):e11627.

Kamimoto, K., Stringa, B., Hoffmann, C. M., Jindal, K., Solnica-Krezel, L., and Morris, S. A. (2023). Dissecting cell identity via network inference and in silico gene perturbation. *Nature*, 614(7949):742–751.

Kloeden, P. E., Platen, E., Kloeden, P. E., and Platen, E. (1992). *Stochastic differential equations*. Springer.

Kouno, T., de Hoon, M., Mar, J. C., Tomaru, Y., Kawano, M., Carninci, P., Suzuki, H., Hayashizaki, Y., and Shin, J. W. (2013). Temporal dynamics and transcriptional control using single-cell gene expression analysis. *Genome biology*, 14:1–12.

La Manno, G., Soldatov, R., Zeisel, A., Braun, E., Hochgerner, H., Petukhov, V., Lidschreiber, K., Kastriti, M. E., Lönnerberg, P., Furlan, A., et al. (2018). Rna velocity of single cells. *Nature*, 560(7719):494–498.

Lee, H.-C., Danieletto, M., Miotto, R., Cherng, S. T., and Dudley, J. T. (2019). Scaling structural learning with no-bears to infer causal transcriptome networks. In *PACIFIC SYMPOSIUM ON BIOCOMPUTING 2020*, pages 391–402. World Scientific.

Li, M. and Belmonte, J. C. I. (2017). Ground rules of the pluripotency gene regulatory network. *Nature Reviews Genetics*, 18(3):180–191.

Li, W., Wang, Z., Syed, S., Lyu, C., Lincoln, S., O'Neil, J., Nguyen, A. D., Feng, I., and Young, M. W. (2021). Chronic social isolation signals starvation and reduces sleep in drosophila. *Nature*, 597(7875):239–244.

Lim, C. Y., Wang, H., Woodhouse, S., Piterman, N., Wernisch, L., Fisher, J., and Göttgens, B. (2016). Btr: training asynchronous boolean models using single-cell expression data. *BMC bioinformatics*, 17(1):1–18.

Liu, F., Zhang, S.-W., Guo, W.-F., Wei, Z.-G., and Chen, L. (2016). Inference of gene regulatory network based on local bayesian networks. *PLoS computational biology*, 12(8):e1005024.

Lo, L.-Y., Wong, M.-L., Lee, K.-H., and Leung, K.-S. (2015). Time delayed causal gene regulatory network inference with hidden common causes. *PloS one*, 10(9):e0138596.

Ma, B., Fang, M., and Jiao, X. (2020). Inference of gene regulatory networks based on nonlinear ordinary differential equations. *Bioinformatics*, 36(19):4885–4893.

Marbach, D., Costello, J. C., Küffner, R., Vega, N. M., Prill, R. J., Camacho, D. M., Allison, K. R., Kellis, M., Collins, J. J., et al. (2012). Wisdom of crowds for robust gene network inference. *Nature methods*, 9(8):796–804.

Matsumoto, H., Kiryu, H., Furusawa, C., Ko, M. S., Ko, S. B., Gouda, N., Hayashi, T., and Nikaido, I. (2017). Scode: an efficient regulatory network inference algorithm from single-cell rna-seq during differentiation. *Bioinformatics*, 33(15):2314–2321.

McDonald, T. O., Cheng, Y.-C., Graser, C., Nicol, P. B., Temko, D., and Michor, F. (2023). Computational approaches to modelling and optimizing cancer treatment. *Nature Reviews Bioengineering*, 1(10):695–711.

Morrissey, E. R., Juárez, M. A., Denby, K. J., and Burroughs, N. J. (2010). On reverse engineering of gene interaction networks using time course data with repeated measurements. *Bioinformatics*, 26(18):2305–2312.

Nagarajan, R. and Upreti, M. (2010). Granger causality analysis of human cell-cycle gene expression profiles. *Statistical applications in genetics and molecular biology*, 9(1).

Nauta, M., Bucur, D., and Seifert, C. (2019). Causal discovery with attention-based convolutional neural networks. *Machine Learning and Knowledge Extraction*, 1(1):19.

Neph, S., Stergachis, A. B., Reynolds, A., Sandstrom, R., Borenstein, E., and Stamatoyannopoulos, J. A. (2012). Circuitry and dynamics of human transcription factor regulatory networks. *Cell*, 150(6):1274–1286.

Nguyen, H., Tran, D., Tran, B., Pehlivan, B., and Nguyen, T. (2021). A comprehensive survey of regulatory network inference methods using single cell rna sequencing data. *Briefings in bioinformatics*, 22(3):bbaa190.

Nouri, N., Gaglia, G., Mattoo, H., de Rinaldis, E., and Savova, V. (2023). Comparative analysis of association networks using single-cell rna sequencing data reveals perturbation-relevant gene signatures. *bioRxiv*, pages 2023–09.

Papili Gao, N., Ud-Dean, S. M., Gandrillon, O., and Gunawan, R. (2018). Sincerities: inferring gene regulatory networks from time-stamped single cell transcriptional expression profiles. *Bioinformatics*, 34(2):258–266.

Perrin, B.-E., Ralaivola, L., Mazurie, A., Bottani, S., Mallet, J., and d'Alché Buc, F. (2003). Gene networks inference using dynamic bayesian networks. *Bioinformatics*, 19(2):138–148.

Peter, I. S. and Davidson, E. H. (2015). *Genomic control process: development and evolution*. Academic Press.

Pinna, A., Soranzo, N., and De La Fuente, A. (2010). From knockouts to networks: establishing direct cause-effect relationships through graph analysis. *PloS one*, 5(10):e12912.

Pratapa, A., Jalihal, A. P., Law, J. N., Bharadwaj, A., and Murali, T. (2020). Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data. *Nature methods*, 17(2):147–154.

Qian, H. and Cheng, Y.-C. (2020). Counting single cells and computing their heterogeneity: from phenotypic frequencies to mean value of a quantitative biomarker. *Quantitative Biology*, 8(2):172–176.

Reid, J. E. and Wernisch, L. (2016). Pseudotime estimation: deconfounding single cell time series. *Bioinformatics*, 32(19):2973–2980.

Reimand, J., Vaquerizas, J. M., Todd, A. E., Vilo, J., and Luscombe, N. M. (2010). Comprehensive reanalysis of transcription factor knockout expression data in saccharomyces cerevisiae reveals many new targets. *Nucleic acids research*, 38(14):4768–4777.

Schaffter, T., Marbach, D., and Floreano, D. (2011). Genenetweaver: in silico benchmark generation and performance profiling of network inference methods. *Bioinformatics*, 27(16):2263–2270.

Shrivastava, H., Zhang, X., Song, L., and Aluru, S. (2022). Grnular: A deep learning framework for recovering single-cell gene regulatory networks. *Journal of Computational Biology*, 29(1):27–44.

Siggiridou, E. and Kugiumtzis, D. (2015). Granger causality in multivariate time series using a time-ordered restricted vector autoregressive model. *IEEE Transactions on Signal Processing*, 64(7):1759–1773.

Smith, G. D., Ching, W. H., Cornejo-Páramo, P., and Wong, E. S. (2023). Decoding enhancer complexity with machine learning and high-throughput discovery. *Genome Biology*, 24(1):116.

Stergachis, A. B., Neph, S., Sandstrom, R., Haugen, E., Reynolds, A. P., Zhang, M., Byron, R., Canfield, T., Stelhing-Sun, S., Lee, K., et al. (2014). Conservation of trans-acting circuitry during mammalian regulatory evolution. *Nature*, 515(7527):365–370.

Street, K., Risso, D., Fletcher, R. B., Das, D., Ngai, J., Yosef, N., Purdom, E., and Dudoit, S. (2018). Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. *BMC genomics*, 19:1–16.

Tomaru, Y., Simon, C., Forrest, A. R., Miura, H., Kubosaki, A., Hayashizaki, Y., and Suzuki, M. (2009). Regulatory interdependence of myeloid transcription factors revealed by matrix rnai analysis. *Genome biology*, 10:1–13.

Treutlein, B., Lee, Q. Y., Camp, J. G., Mall, M., Koh, W., Shariati, S. A. M., Sim, S., Neff, N. F., Skotheim, J. M., Wernig, M., et al. (2016). Dissecting direct reprogramming from fibroblast to neuron using single-cell rna-seq. *Nature*, 534(7607):391–395.

Ventre, E. (2021). Reverse engineering of a mechanistic model of gene expression using metastability and temporal dynamics. *In Silico Biology*, 14(3-4):89–113.

Ventre, E., Herbach, U., Espinasse, T., Benoit, G., and Gandrillon, O. (2023). One model fits all: combining inference and simulation of gene regulatory networks. *PLoS Computational Biology*, 19(3):e1010962.

Vijayan, V., Wang, Z., Chandra, V., Chakravorty, A., Li, R., Sarbanes, S. L., Akhlaghpour, H., and Maimon, G. (2022). An internal expectation guides drosophila egg-laying decisions. *Science Advances*, 8(43):eabn3852.

Vinci, G., Dasarathy, G., and Allen, G. I. (2019). Graph quilting: graphical model selection from partially observed covariances. *arXiv preprint arXiv:1912.05573*.

Vitezic, M., Lassmann, T., Forrest, A. R., Suzuki, M., Tomaru, Y., Kawai, J., Carninci, P., Suzuki, H., Hayashizaki, Y., and Daub, C. O. (2010). Building promoter aware transcriptional regulatory networks using sirna perturbation and deepcage. *Nucleic acids research*, 38(22):8141–8148.

Wang, L., Trasanidis, N., Wu, T., Dong, G., Hu, M., Bauer, D. E., and Pinello, L. (2023). Dictys: dynamic gene regulatory network dissects developmental continuum with single-cell multiomics. *Nature Methods*, 20(9):1368–1378.

Wang, Y. and He, S. (2023). Inference on autoregulation in gene expression with variance-to-mean ratio. *Journal of Mathematical Biology*, 86(5):87.

Wang, Y., Kropp, J., and Morozova, N. (2020). Biological notion of positional information/value in morphogenesis theory. *International Journal of Developmental Biology*, 64(10-11-12):453–463.

Wang, Y. and Wang, Z. (2022). Inference on the structure of gene regulatory networks. *Journal of Theoretical Biology*, 539:111055.

Wang, Z. (2020). *Identification of Gene Expression Changes in Sleep Mutants Associated With Reduced Longevity in Drosophila*. PhD thesis, The Rockefeller University.

Wang, Z., Lincoln, S., Nguyen, A. D., Li, W., and Young, M. W. (2022). Chronic sleep loss disrupts rhythmic gene expression in drosophila. *Frontiers in Physiology*, 13:1048751.

Wasserman, L. (2004). *All of statistics: a concise course in statistical inference*, volume 26. Springer.

Wen, Y., Huang, J., Guo, S., Elyahu, Y., Monsonego, A., Zhang, H., Ding, Y., and Zhu, H. (2023). Applying causal discovery to single-cell analyses using causalcell. *Elife*, 12:e81464.

Wolpert, L. (1989). Positional information revisited. *Development*, 107(Supplement):3–12.

Wu, B., Piatkevich, K. D., Lionnet, T., Singer, R. H., and Verkhusha, V. V. (2011). Modern fluorescent proteins and imaging technologies to study gene expression, nuclear localization, and dynamics. *Current opinion in cell biology*, 23(3):310–317.

Xing, B. and Van Der Laan, M. J. (2005). A causal inference approach for constructing transcriptional regulatory networks. *Bioinformatics*, 21(21):4007–4013.

Yang, Y.-J. and Cheng, Y.-C. (2021). Potentials of continuous markov processes and random perturbations. *Journal of Physics A: Mathematical and Theoretical*, 54(19):195001.

Zhang, S., Pyne, S., Pietrzak, S., Halberg, S., McCalla, S. G., Siahpirani, A. F., Sridharan, R., and Roy, S. (2023). Inference of cell type-specific gene regulatory networks on cell lineages from single cell omic datasets. *Nature Communications*, 14(1):3064.

Zhang, X., Zhao, X.-M., He, K., Lu, L., Cao, Y., Liu, J., Hao, J.-K., Liu, Z.-P., and Chen, L. (2012). Inferring gene regulatory networks from gene expression data by path consistency algorithm based on conditional mutual information. *Bioinformatics*, 28(1):98–104.

Zhang, Y., Chang, X., and Liu, X. (2021a). Inference of gene regulatory networks using pseudo-time series data. *Bioinformatics*, 37(16):2423–2431.

Zhang, Y., Wang, D., Peng, M., Tang, L., Ouyang, J., Xiong, F., Guo, C., Tang, Y., Zhou, Y., Liao, Q., et al. (2021b). Single-cell rna sequencing in cancer research. *Journal of Experimental & Clinical Cancer Research*, 40:1–17.

Zhang, Z., Hung, Y., Chan, S., Xu, W., and Hu, Y. (2010). Modeling and identification of gene regulatory networks: A granger causality approach. In *2010 International Conference on Machine Learning and Cybernetics*, volume 6, pages 3073–3078. IEEE.

Zhao, M., He, W., Tang, J., Zou, Q., and Guo, F. (2021). A comprehensive overview and critical evaluation of gene regulatory network inference technologies. *Briefings in Bioinformatics*, 22(5):bbab009.

Zhao, M., He, W., Tang, J., Zou, Q., and Guo, F. (2022). A hybrid deep learning framework for gene regulatory network inference from single-cell transcriptomic data. *Briefings in bioinformatics*, 23(2):bbab568.

Zheng, R., Li, M., Chen, X., Wu, F.-X., Pan, Y., and Wang, J. (2019). Bixgboost: a scalable, flexible boosting-based method for reconstructing gene regulatory networks. *Bioinformatics*, 35(11):1893–1900.

Zhong, C. (2021). Mallows permutation models with l^1 and l^2 distances i: hit and run algorithms and mixing times. *arXiv preprint arXiv:2112.13456*.

Zhong, C. and Tibshirani, R. (2019). Survival analysis as a classification problem. *arXiv preprint arXiv:1909.11171*.