Briefings in Bioinformatics, 22(5), 2021, 1–15

https://doi.org/10.1093/bib/bbab009
Advance Access Publication Date: 00 Month 0000
Method review

A comprehensive overview and critical evaluation of gene regulatory network inference technologies

Mengyuan Zhao†, Wenying He†, Jijun Tang, Quan Zou and Fei Guo

Corresponding authors: Fei Guo, School of Computer Science and Technology, College of Intelligence and Computing, Tianjin University, Tianjin, China. E-mail: fguo@tju.edu.cn; Quan Zou, Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu, China. E-mail: zouquan@nclab.net
†Mengyuan Zhao and Wenying are co- first author.

Abstract

Gene regulatory network (GRN) is the important mechanism of maintaining life process, controlling biochemical reaction and regulating compound level, which plays an important role in various organisms and systems. Reconstructing GRN can help us to understand the molecular mechanism of organisms and to reveal the essential rules of a large number of biological processes and reactions in organisms. Various outstanding network reconstruction algorithms use specific assumptions that affect prediction accuracy, in order to deal with the uncertainty of processing. In order to study why a certain method is more suitable for specific research problem or experimental data, we conduct research from model-based, information-based and machine learning-based method classifications. There are obviously different types of computational tools that can be generated to distinguish GRNs. Furthermore, we discuss several classical, representative and latest methods in each category to analyze core ideas, general steps, characteristics, etc. We compare the performance of state-of-the-art GRN reconstruction technologies on simulated networks and real networks under different scaling conditions. Through standardized performance metrics and common benchmarks, we quantitatively evaluate the stability of various methods and the sensitivity of the same algorithm applying to different scaling networks. The aim of this study is to explore the most appropriate method for a specific GRN, which helps biologists and medical scientists in discovering potential drug targets and identifying cancer biomarkers.

Key words: gene regulatory network; gene expression data; network inference methods; machine learning

Introduction

Gene expression regulation is a complex process in which intricate interactions are formed between genes and gene products (such as proteins, etc.). If these interactions are depicted by lines, they will present a network structure, that is, gene regulatory network (GRN). GRN plays an important role in every kind of organism and system [1–3] because it is the core element to maintain life processes, control biochemical reactions, regulate the level of compounds, etc. Therefore, the reconstruction of GRN can deeply understand the molecular mechanism of organisms and reveal the essential rules of a large number of biological processes and reactions in organisms [4–7]. Therefore, the reconstruction of GRN is the hot issue in the field of current bioinformatics research. GRN can be applied in many fields such as medicine, industry, biological research, etc. [8–10]. GRN provides important support for the discovery of potential drug targets [11, 12] and identification of cancer biomarkers in medical field [13].

Genetic information is stored in DNA in the form of genes and, through transcription and translation, forms a

Mengyuan Zhao is currently a master's degree candidate at Tianjin University. Her research interests include bioinformatics and machine learning.
Wenying He is a PhD candidate at Tianjin University. Her research interests include bioinformatics and machine learning.
Jijun Tang is a professor at the University of South Carolina. His main research interests include computational biology and algorithm.
Quan Zou is a professor at the University of Electronic Science and Technology of China. His main research interests include bioinformatics, machine learning and parallel computing.
Fei Guo is an associate professor at Tianjin University. Her research interests include bioinformatics and computational biology.
Submitted: 28 September 2020; **Received (in revised form):** 11 December 2020

functional gene product—protein—which is the process of gene expression. Therefore, gene regulation occurs at multiple levels. Direct observation of interactions in organisms is undoubtedly difficult, but it is feasible to measure element abundance (e.g. mRNA) by experimental means, and the expression thread of these elements often reflects the regulatory role of genes [14]. The microarray technology developed by Brown *et al*. [15] in 1999 opened the door to massive parallel detection [16]. They simultaneously measured the concentration of thousands of biomolecules by hybridization to obtain the amount of mRNA produced during transcription. Next-generation sequencing (NGS) technology is a revolutionary advance in sequencing technology [17], which greatly improves throughput by parallelizing sequence reactions. RNA-seq is an important application of NGS to detect RNA abundance in samples at a certain time through reverse transcription, fragmentation and other processes [18]. Chromatin immunoprecipitation followed by sequencing (ChIP-seq) is a powerful tool for studying protein–DNA interactions *in vivo* [19]. Based on experimental means above, many data repositories provide access to biomedical and genomic information and further provide data resources for computational methods. Gene Expression Omnibus (GEO) is an open gene expression abundance database developed by the National Center for Biotechnology Information in 2000 [20], which contains gene expression information based not only on ChIP-seq technology but also on non-chip technologies such as SAGE and mass spectrometry. ArrayExpress records gene expression data obtained from microarrays and sequencing platforms [21]. RegulonDB extensively and detailedly records information on *Escherichia coli* K-12 regulatory relationships, including not only the data obtained from original scientific publications but also the regulatory relationships validated by experimental means and those obtained by computational prediction [22]. In addition to the above-mentioned, commonly used databases also include signaling pathway databases (KEGG), Gene Ontology Consortium, Encyclopedia of DNA Elements, etc. [14, 23–25].

Although a large number of known regulatory relationships in organisms have been documented in various databases, they are still far from the number of interactions and complex relationships that actually exist in biological systems. Experiments are generally able to measure the abundance of elements, but it is difficult to directly discover the complex relationships among them. Moreover, with the development of high-throughput sequencing technology and experimental means, the number of known gene expression profile data is rapidly increasing and the types are diverse. It is undoubtedly unrealistic to carry out experimental verification of the potential regulatory relationship one by one only by human resources. Based on the above limitations, many kinds of computational methods and improvements have been proposed to infer GRNs, but their suitable scenarios and hypothetical conditions are quite different [26–29]. Existing methods for reverse engineering of GRNs mainly include correlation methods, Boolean network methods, Bayesian network methods, differential equation methods, neural network methods, regression methods and others [30–34]. In this study, we roughly divide them into three categories: model-based methods, information theory-based methods and machine learning-based methods.

Model-based methods construct a model that can accurately describe the relationship among genes by fitting gene expression profiles and then use this model to construct GRN. This type of methods can choose different models according to the data types and network characteristics, with high flexibility and scalability. Boolean network methods, Bayesian network methods, differential equation methods and neural network methods all fall into this category. Boolean network methods use logical values (0,1) to represent the states of genes (expression, non-expression) [35] and construct a Boolean model that can describe the relationship between genes. The Boolean model represents a regulatory process of genes by one logical function. Probabilistic Boolean model is an extension of Boolean model. Different from Boolean model, probabilistic Boolean model uses multiple logical functions to represent a regulatory process and each logical function works with different probabilities [36, 37], which is consistent with the randomness and uncertainty of gene interactions in biological systems. Bayesian network methods use Bayesian model to predict intergenic relationships combing prior information such as gene expression profile. However, Bayesian model has some inherent shortcomings which induce the emergence of extension methods. Dynamic Bayesian network methods overcome the weakness of Bayesian model that it is only suitable for steady-state data. Dynamic Bayesian network takes time-series data as input, which opens the door for Bayesian model to utilize time information of genes effectively [38–40]. For the problem that applying Bayesian model on large-scale networks needs long running time, local Bayesian network methods divide the whole network into many small-scale local networks and apply Bayesian model on each local network, which significantly reduces the calculation time [41]. Differential equation methods use differential equation to construct models to describe intergenic relationships, which are mainly divided into linear differential equation methods [42–44] and nonlinear differential equation methods [45, 46]. By contrast, the latter can capture more complex interactions which are consistent with genetic relationships in biological system. Neural network, which can model any functional relationship and data structure [47], is a highly flexible and scalable method that can represent the dynamic relationship of genes. Different neural network models can be selected to deal with different types of gene data, such as convolutional neural network for gene expression images, recurrent neural network for gene time-series data, etc. [48–50]. Neural network methods show the potential of deep learning to solve GRN inference problems.

Information theory-based methods, also known as correlation methods, are the most widely used methods [51, 52]. Because genes of the same group often exhibit similar expression patterns in physiological processes, such methods set a standard that can measure the correlation between genes and the higher the value, the higher the probability of their interaction. Such methods have low computational complexity and can build large networks based on a small amount of data, that is, low demand for sample size. However, the limitations of such methods are also obvious. Since the correlation is bidirectional, the inferred network is often undirected. Commonly used metrics include Pearson correlation coefficient (PCC), mutual information (MI), conditional mutual information (CMI), etc. [53–56]. Each metric has its merit and demerit. PCC can distinguish regulation mode between promoting and inhibiting but only recognizes linear correlations between genes. MI can capture linear and nonlinear correlation between genes but is prone to misidentify indirect regulation as direct regulation. CMI identifies indirect relationships effectively but generally underestimates the regulation strength and, as a result, loses some true positive edges of GRN (see Method for details). Due to the limitations mentioned above, many methods make targeted improvements based on commonly used standards, propose new measurement methods and obtain accurate prediction results. Based on MI, MICRAT

applies maximal information coefficient as the standard to mine functional and nonfunctional associations between genes and calculate CMI for genes involved in triangle cycles to remove redundant edges [56]. PCA-PMI combines partial independence with CMI and proposes novel, metric, part mutual information (PMI), which overcomes the underestimation problem of CMI to a large extent [57]. Similarly, for the underestimation problem of CMI, CMI2NI considers both CMI and Kullback–Leibler divergence and further proposes conditional mutual inclusive information as the standard of measuring correlation between genes [58]. Various improvements of existing metrics and new information theory concepts proposed constantly make the solution of GRN inference problem has more directions worth researching in the future.

Machine learning-based methods mainly use machine learning calculation methods and data structures to fit gene expression data and regression methods belong to this category [59–61]. This kind of methods has strong interpretability and can identify the direction of regulation, so the obtained GRN is often a directed network. Machine learning algorithms commonly used in GRN reconstruction include random forest (RF), Extra-tree, boosting, etc. [62–68]. RF and Extra-tree both use the structure of decision tree (DT) to solve regression or classification problems. Extra-tree is similar to RF in that it avoids over-fitting by random sampling, except that each DT in RF has a playback sampling to determine the training dataset, while Extra-tree's random selection occurs when leaf nodes are split. By contrast, RF fits better while Extra-tree has stronger randomness and better generalization ability. GENIE3 utilizes RF to infer GRN and won the first place in the DREAM5 *In Silico Network* Challenge in 2010 [63]. Based on the architecture of GENIE3, GRNBoost2 selects gradient boosting machine instead of RF to solve regression problem and gets more accurate prediction results [66]. BiXGBoost adopts XGBoost, which not only makes use of regularization term to prevent overfitting but also supports parallel computing, which greatly accelerates the training speed [69]. Of course, there are some methods that do not directly adopt classical machine learning algorithms but give improvements to achieve higher accuracy, such as Jump3 [70]. The methods mentioned above all use tree-based ensemble methods to construct network, which proves the strong applicability of such machine learning algorithms and the effectiveness of mining regulation laws behind gene expression data. In addition, support vector machines (SVMs) are a widely used classifier to infer GRN nowadays. SIRENE transforms the network inference problem into many target gene identification problems for transcription factors (TFs), that is, binary classification problems, and uses SVM as the classifier to predict gene interactions [71]. GRADIS applies k-means clustering algorithm for data preprocessing before SVM classification, which further improves the prediction result [67]. The performance improvement brought by data preprocessing implies that feature extraction before training model is worth further study.

According to the underlying design of different methods, we introduce the principle, overall steps, hypotheses and characteristics of each category in details and compare representative methods for three categories according to the most classical and the latest specific technologies. We conduct experiments on datasets of different scales of simulated and real networks to explore the stability of various methods and their sensitivity on different scales of networks. The output we obtain mainly includes two types: directed network and undirected network. We study the effect of input data (steady-state and time-series data) on the accuracy of reconstructing network. Finally, we give recommendation methods with strong applicability and high robustness for different data types and network scales. The overall process of reconstructing GRN based on gene expression data is shown in Figure 1.

Dataset

Different methods deal with different types of data, mainly including steady-state data and time-series data. The steady-state data refer to the expression level of genes in a stable state under different experimental environments, such as gene knockout, gene knockdown, etc. The time-series data reflect how the network responds to perturbations and recovers after removing perturbations. It contains the expression level of each gene at different time points. From another point of view, input data can be divided into simulated data and real data according to the source of the dataset. The simulated data use synthetic network as standard network and simulate different experimental conditions through mathematical means. Such data can provide not only steady-state and time-series data but also other information such as TFs to help network inference. The simulated data are mainly derived from DREAM Challenges (especially DREAM4 and DREAM5) [72], an open platform that hosts biomedical competitions. Real data are derived from biological experiments on both standard network and expression level data. Furthermore, such data may collect either steady-state data or time-series data for partial methods. However, there are real regulatory relationships that have not been verified by experiments but exist within biological mechanisms. Therefore, in addition to known regulatory relationships, we should mine more unknown relationships. Here, real data mainly come from GEO, RegulonDB and other databases.

We conduct comparative analysis on simulated and real datasets of different scales. For simulated data, we choose *in silico* dataset from DREAM4 *In Silico Network* Challenge, which is also the most popular benchmark dataset. For real data, we choose a small-scale SOS DNA repair network dataset and a large-scale *E. coli* dataset from DREAM5 and RegulonDB. The specific information of each dataset is shown in Table 1.

DREAM4 dataset

DREAM Challenges is community-designed and run by researchers from various organizations, inviting participants to come up with solutions to basic biomedical problems, promote collaboration and further build communities in the process. DREAM4 *In Silico Network* Challenge [73–75] is one of the challenges raised by the community and the most popular simulated data in research articles nowadays. The data provided by DREAM4 comes from an open-source software called GeneNetWeaver [76], which can not only generate benchmark networks and datasets but also analyze the performance of network construction methods. It extracts the topological structure of regulatory networks on *E. coli* and *Saccharomyces cerevisiae* and improves some detailed information to form the benchmark network. Then, the benchmark is simulated to generate gene steady-state expression data and time-series expression data. DREAM4 provides training datasets and directed benchmark networks for regulatory networks of 10 genes and 100 genes, respectively. Each scale contains five directed networks, each of which provides gene expression data under six different experimental conditions including knockout, multifactorial, time-series, etc. Especially, we select multifactorial data to represent gene steady-state data. Each sample can be regarded as the change of

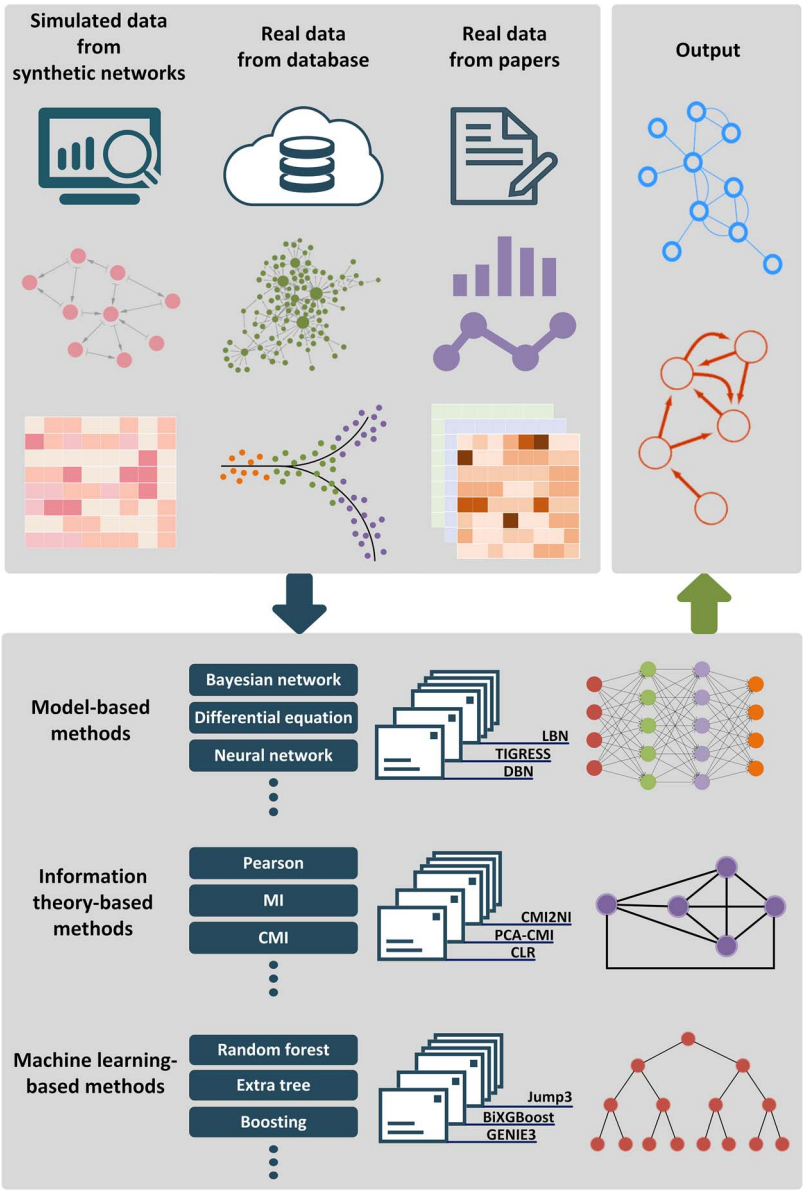

Figure 1. An overview of the framework for constructing GRN. According to different sources, the input mainly includes simulated data from synthetic networks, real data from database and real data from papers. By means of process and calculation, various computational methods then identify the regulatory relationship behind data.

gene expression after the cell being physically or chemically interfered. We perform a comprehensive assessment of all selected GRN inference methods on DREAM4 dataset.

SOS DNA repair network dataset

SOS DNA repair network dataset is a real gene expression dataset with experimental verification [77, 78]. It often serves as standard GRN to test the effectiveness of algorithms. This is a small-scale directed network with 9 genes and 24 regulations. Technologies suitable for steady-data are compared on this dataset.

E. coli dataset

GSE20305 [79] provides real gene time-series data of *E. coli* under different experimental environments such as coldstress, control, heatstress, etc. We select the data under coldstress, heatstress and oxidativestress to constitute the training data. We combine the benchmark network provided by DREAM5 with the *E. coli* data recorded in RegulonDB to form the final benchmark network. The network is a large-scale directed real regulatory network containing 1484 genes and 163 regulators. Since the data stream has the characteristic of evolution, we analyze corresponding models handling time-series data.

Method

The GRN inference methods can use a series of statistical analysis means to mine the essential rules on partial gene expression data, discover interactions reflected by the gene expression level and finally present complex regulatory relationships in the form of network. Real and simulated expression data are all tried

Table 1. Detailed dataset information

Type	Dataset	Network	Genes	Candidate regulators	Steady state		Time series		Edges	Density
					No. of experiments		No. of experiments	Time points		
Simulation dataset	DREAM4 *in silico* size 10	5	10	10	10		5	21	14	0.15778
	DREAM4 *in silico* size 100	5	100	100	100		10	21	101	0.09546
Real dataset	SOS DNA repair network	1	9	9	9		–	–	24	0.33333
	E. coli dataset	1	1484	163	–		3	8	3080	0.0014
							3	8		
							3	11		

Note: Network: number of networks included in dataset. Edges: number of edges in gold GRN. If there is more than one network included in dataset, average edges are showed in Table 1. Density: Density = Edges/Candidate edges. Self-regulations are removed in gold GRNs.

as the input data. Simulation data are widely used to compare the performance of methods, and real data with biological significance can discover potential regulatory relationships. The simulation data are mainly derived from DREAM Challenges, an open, perennial platform for biological and pharmaceutical competitions. The real data mainly come from GEO, ArrayExpress and other databases. In this paper, we roughly sort these methods into three categories according to their underlying design and characteristics. Model-based methods use classical models to describe intergenic relationships, information theory-based methods reflect their regulatory relationships by measuring intergenic correlations and machine learning-based methods use machine learning algorithms to discover the essential rules behind gene expression levels. We analyze classical, current mainstream approaches (12 in total) in order to descript each category of approaches in details, summarized as Table 2.

Model-based methods

Model-based methods first define a specific model using nonlinear differential equation, Bayesian network or other models and then infer the network by identifying model parameters. The general steps are as follows.

(i) Select a reasonable quantitative dynamical model according to the type of input data.
(ii) Optimize the model by means of methods such as expectation–maximization (EM), least angle regression (LARS), etc.
(iii) Construct GRN based on optimized model.

The most importance among these three steps is Step 2. Different optimization methods have a great influence on the fitting degree of final model to the input data, the speed of model optimization, etc., which is also one of the important factors determining the accuracy in constructing GRN.

DBN [38] is a classic model-based method, which utilizes Bayesian network as the basic model. Based on the hypothesis that the unit time of data measurement is lower than evolution time of gene, it adds a hidden layer to the original Bayesian model to record all expression levels, change rates, random noises and other information. Then, the EM algorithm [80] is used to learn the parameters and obtain the final model which can fit the input data well. DBN makes full use of time information of genes and is suitable for data with noise. TIGRESS [42] supposes that there is a linear correlation between regulatory genes and target genes. Based on this assumption, TIGRESS transforms the network reconstruction problem into a feature selection problem, establishes a linear model between regulatory genes and target genes and solves the feature selection problem by LARS [81] combined with stability selection [82]. This method achieves top one rank in the DREAM5 challenge for the same type of methods but is less effective in real biological datasets, which is caused by its assumption of linear correlation that does not match the complex gene relationship in the organism. Unlike TIGRESS, NonlinearODEs [45] is proposed based on the framework of nonlinear ordinary differential equations combining steady-state data and time-series data. Compared with linear ordinary differential model, the nonlinear model requires more gene expression data and higher computational complexity, but it can characterize the nonlinear, combinatorial and multiple interactions among genes, which is also consistent with the complex relationship of genes in the organism.

Various basic models can apply on different datasets and network types, whether such specific methods dealing with gene

Table 2. Summary of GRN inference algorithms

Category	Method	Year	Description	Data type		Directed	Characteristic
				Steady	Time-series		
Model-based methods	DBN	2003	Regulatory genes have a cumulative effect on target gene.	–	√	Y	Simulate circular interaction and dynamic processes.
	TIGRESS	2012	Linear regression is taken as estimation for the relational expression.	–	√	Y	Formulate the GRN inference problem as a feature selection problem.
	NonlinearODEs	2020	Nonlinear ODEs framework is to model dynamic gene regulation.	√	√	Y	Mine the nonlinear correlation between genes.
Information theory-based methods	CLR	2007	Spline estimation of MI scores	√	–	N	Infer potential causality and predict feed forward loops.
	ARACNe	2005	Kernel estimation of MI scores	√	–	N	Inference a hierarchical scale-free network.
	PCA-CMI	2011	CMI scores describe conditional independence.	√	–	N	Both nonlinear dependence and sparse structure are considered.
	CMI2NI	2014	Conditional mutual inclusive information describes regulation.	√	–	N	Quantify the regulation strength.
	PCA-PMI	2016	PMI measure direct dependencies.	√	–	N	Overcome overestimation and underestimation of MI and CMI.
Machine learning-based methods	BiXGBoost	2018	XGBoost evaluates the regulation importance.	–	√	Y	Bidirectional model mining complex regulation
	GENIE3	2010	RFs identify the regulators of each target gene.	√	–	Y	Effectively identify linear, nonlinear and other regulation.
	dynGENIE3	2018	Improved GENIE3 method to process time-series data.	√	√	Y	Complex intergenic correlations can be found.
	Jump3	2015	Dynamical model based on DTs	–	√	Y	Detect multivariate interacting effects between genes.

steady-state data or time-series data depends on the model selection and the implementation details of the algorithm. For example, Bayesian model-based methods are suitable for steady-state data, dynamic Bayesian model-based methods are most applicable for time-series data and differential equation-based methods are suitable for both of them [83].

Information theory-based methods

Information theory-based methods infer regulatory relationship through correlation between genes. Due to its simple core idea and least gene expression data, the information theory-based method is the most popular method nowadays. However, since the correlation between genes is bidirectional, GRN obtained by this kind of methods is often an undirected network, which is also a disadvantage of it. The method based on information theory can basically be divided into the following steps.

(i) Customize a standard to measure the correlation between variables or choose a classic correlation coefficient.
(ii) Calculate the correlation and ignore values below the threshold.
(iii) Identify indirect regulatory and remove redundant edges.
(iv) Rank all regulatory relationships by correlation and construct network.

The correlation metrics of different methods may be different, mainly including PCC, Spearman correlation coefficient, Kendall correlation coefficient, MI, etc.

ARACNe and CLR [53, 54] apply MI as the standard to measure the correlation between gene pairs. However, GRN determined solely on the basis of MI is prone to high false positive rate. As shown in Figure 2A, if both X and Y have a high correlation with Z but there is no direct regulatory relationship between them, the MI score is greater than 0. If the threshold is set low, it is mistaken for a direct regulation relationship between X and Y. To avoid this problem, ARACNe adds data processing inequality with tolerance threshold to delete indirect regulatory edges. Due to the deficiency of MI above, a number of methods choose CMI as a standard of measurement such as PCA-CMI [84]. Since CMI takes into account the information of given Z, in the case of Figure 2A, the CMI score is equal to 0, that is, the indirect correlation between X and Y can be identified. However, the CMI score between two variables being 0 cannot indicate that there is no correlation between them. If both X and Y have a high correlation with gene Z, then there is indeed a direct regulatory relationship between them, as shown in Figure 2B, but the CMI score may be equal to 0. So, GRN constructed on CMI merely often loses some true regulatory relationships. In order to avoid this shortcoming, PCA-PMI and CMI2NI [57, 58] are proposed based on CMI and both improve the accuracy to a certain extent.

Since information theory-based methods are based on the assumption that the sample values of gene expression profiles are independent of each other, they are particularly suitable for steady-state data.

Machine learning-based methods

Machine learning-based method usually decomposes the GRN inference into specific classification or regression problem based on feature engineering and divide-and-conquer strategy. The feature selection technology of machine learning obtains the subset of regulatory factors with strong correlation to the target factors. Using feature selection algorithm to reconstruct GRN is to take gene expression data as learning samples. The expression value of each gene is taken as the learning objection to select the gene subset with strong correlation to current gene, which is the regulatory gene of the current gene. The general steps are shown as follows.

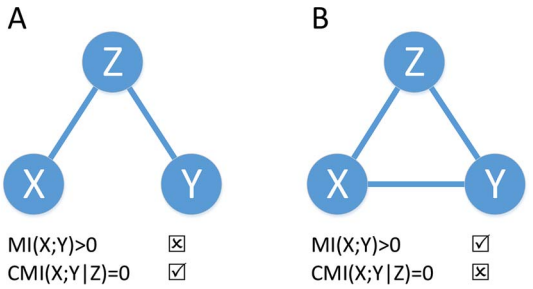

Figure 2. A brief interpretation of differences among MI and CMI. (**A**) X and Y have no direct regulation, but they both have a high correlation with Z. In this way, the MI score between X and Y must be high because of their common neighbor Z, resulting in wrong signal. However, the CMI score is zero and gives the correct signal. (**B**) X and Y have direct regulation, but X or Y is strongly associated with Z. However, the CMI score between X and Y approaches zero, thus giving a wrong signal.

(i) Transform into classification or regression problem.
(ii) Select the corresponding machine learning algorithm to solve the problem.
(iii) Rank according to the weight of the regulatory relationship, and build network.

According to the characteristics of known gene information, suitable machine learning algorithm can be selected to improve its strengths and avoid its weaknesses, so as to make the results more accurate and meet some specific requirements of users such as computational complexity, network directionality, etc.

GENIE3 [63] transforms the network construction problem into several sub-problems of regression and then applies RF to construct a DT group for each target gene. It makes use of the characteristics of RF to obtain the importance of each gene to target gene, the higher the importance indicates the greater the possibility that this gene regulates the target gene, and finally constructs GRN according to the importance ranking. Due to the advantage of RF, GENIE3 won the first place in DREAM5 challenge, achieving the highest accuracy. However, GENIE3 can only be applied to gene steady-state data, which is also the main shortcoming of it. DynGENIE3 [62] is an improved method based on GENIE3. By preprocessing the input data, dynGENIE3 can be applied to both gene steady-state data and time-series data while ensuring accuracy. Jump3 [70] reconstruct the GRN using Jump Tree with consideration of temporal information. At the same time, the on/off model [85] is used to indicate the change of gene expression level over time after taking into account information such as decay rate. Considering real biological details and simulating complex biological processes, Jump3 can discover potential key regulatory genes and help scientists make biologically reasonable hypotheses. BiXGBoost [69] believes that the expression level of target gene at one time point is the sum of regulatory effects of all the regulatory genes in previous time intervals. Simultaneously, the same gene regulates the expression levels of its target genes at the subsequent time intervals. According to this idea, BiXGBoost defines a bidirectional model to mine candidate regulatory genes and target genes of a specific gene. In addition, this method utilizes XGBoost to construct DT

group, which is used to fit gene expression data and evaluates the importance of regulatory relationships.

Machine learning algorithms are usually used to solve classification or regression problems and not directly related to the type of input data. Therefore, which type of data one machine learning-based method applies on relies heavily upon model assumptions and implementation details.

Performance metrics

All results of each method include inferred regulatory relationships and corresponding weights. Then, GRN can be predicted according to the given threshold. Comparing the predicted GRN with gold standard, the following indicators can be obtained.

False positive (FP): the edges exist in the predicted GRN but not in the gold standard.

False negative (FN): the edges do not exist in the predicted GRN but exist in the gold standard.

True positive (TP): the edges exist in the predicted GRN and exist in the gold standard.

True negative (TN): the edges do not exist in the predicted GRN or in the gold standard.

$FPR = \frac{FP}{FP+TN}$

$TPR = \frac{TP}{TP+FN}$

$Precision = \frac{TP}{TP+FP}$

$Recall = \frac{TP}{TP+FN}$

The experimental results of each method provide all the predicted edges and their corresponding weights. The higher the weight, the higher the credibility of the regulatory relationship. Therefore, different GRNs can be constructed by selecting different thresholds. Correspondingly, FPR, TPR, Precision and Recall values are different. In order to comprehensively consider the experimental results under different thresholds, we select AUROC and AUPR values as evaluation criteria. ROC curve can be drawn from FPR and TPR. AUROC refers to the area under the ROC curve. PR curve can be drawn by Precision and Recall and AUPR refers to the area under the PR curve.

Result

In this study, we have evaluated twelve methods, including three model-based methods, five information theory-based methods and four machine learning-based methods. These methods are tested on different simulated datasets (DREAM4 datasets of 10 genes and 100 gene networks) and real experimental datasets (SOS DNA repair network dataset and *E. coli* dataset). In order to make a fair comparison among the above methods in this survey, we consistently use default parameters when running programs. Based on the performance and data need for each computational method, we provide the guidance for the application and development of them.

Comparison of various methods

We evaluate the performance of three categories of various computational methods on simulated and real datasets. The results are summarized in Figure 3. Importantly, we test more than one subset on both DREAM4 network (subnetworks) and *E. coli* network (multiple experiment conditions), so we use arithmetic means of AUROC (AUPR) scores for analyzing the average performance of different methods. Detailed information is given in Supplementary file (Supplementary Tables S1 and S2).

Experiments show that various categories of methods have different performance on different data types (Figure 3). On the simulated data, the machine learning-based methods show the overall superiority to the others and the model-based methods come next. On the contrary, model-based methods perform better on real data. Moreover, performance can vary greatly between methods of the same category and no category of methods is consistently superior to a particular data type. To a great extent, the performance of a particular method depends on the implementation of any particular type. For example, NonlinearODEs is a model-based approach based on nonlinear ordinary differential equations, but its implementation incorporates machine learning algorithm XGBoost to determine how important candidate regulatory genes are to target genes.

For *in silico* datasets (Figure 3A and B), it is obvious that machine learning-based methods achieve better results than other categories in general (except GENIE3). The main reason is related to the construction process of *in silico* datasets and the advantages of machine learning algorithms in mining essential laws of data and feature selection. Moreover, NonlinearODEs, one of the model-based methods, performs higher than the other methods; although, this category is not the best one. The nonlinear differential equation designed by NonlinearODEs is consistent with the complex interactions among genes, so it is able to capture more true regulatory relationships. Besides, the results of information theory-based methods are close to each other and PCA-CMI is the best of them. It mainly links to the effectiveness of strategy for PCA-CMI that calculates high-order CMI to delete redundant edges.

For real datasets (Figure 3C and D), since they provide only one type of data (steady-state data for SOS DNA repair network and time-series data for *E. coli* network), we only consider some methods that can apply on corresponding type of data. Since correlation coefficient is calculated from the steady-state information of genes, such methods are often not applicable to datasets that provide only time-related information, as shown in Figure 3D. In particular, all available methods obtain worse results with less than 0.02 AUPR values on *E. coli* network (Figure 3D). This is due to the fact that AUPR tends to present smaller values on large-scale networks. Large-scale biological networks are hierarchical and scale-free [86, 87]. However, some methods do not distinguish hub genes from common genes in the process of calculation, resulting in more false positive edges. DynGENIE3 and BiXGBoost perform well on simulation datasets (second and third of *in silico* size 10, first and third of *in silico* size 100), while perform poorly on *E. coli* network. When dealing with time-series data, dynGENIE3 assumes that the change rate of gene expression levels at adjacent time points reflects the regulation from other genes. However, due to the limitation of experimental conditions, in many real datasets such as *E. coli* network, the interval of adjacent time points does not necessarily correspond to the time of regulation effect, that is, time lag problem. Taking the time lag into account in the process of training model may get better results. In large-scale biological networks, one regulatory gene generally corresponds to many target genes, while one target gene corresponds to only a few regulatory genes. BiXGBoost searches for target genes and regulatory genes with the same treatment, resulting in high false positive rate. We believe the identification of hub genes is a reliable improving direction.

From the above analysis, we find that machine learning-based methods tend to perform well on simulation data and large-scale networks. This is mainly due to the fact that machine learning-based methods generally convert network inference problems into feature selection problems. Appropriate feature extraction models can accurately extract features for each

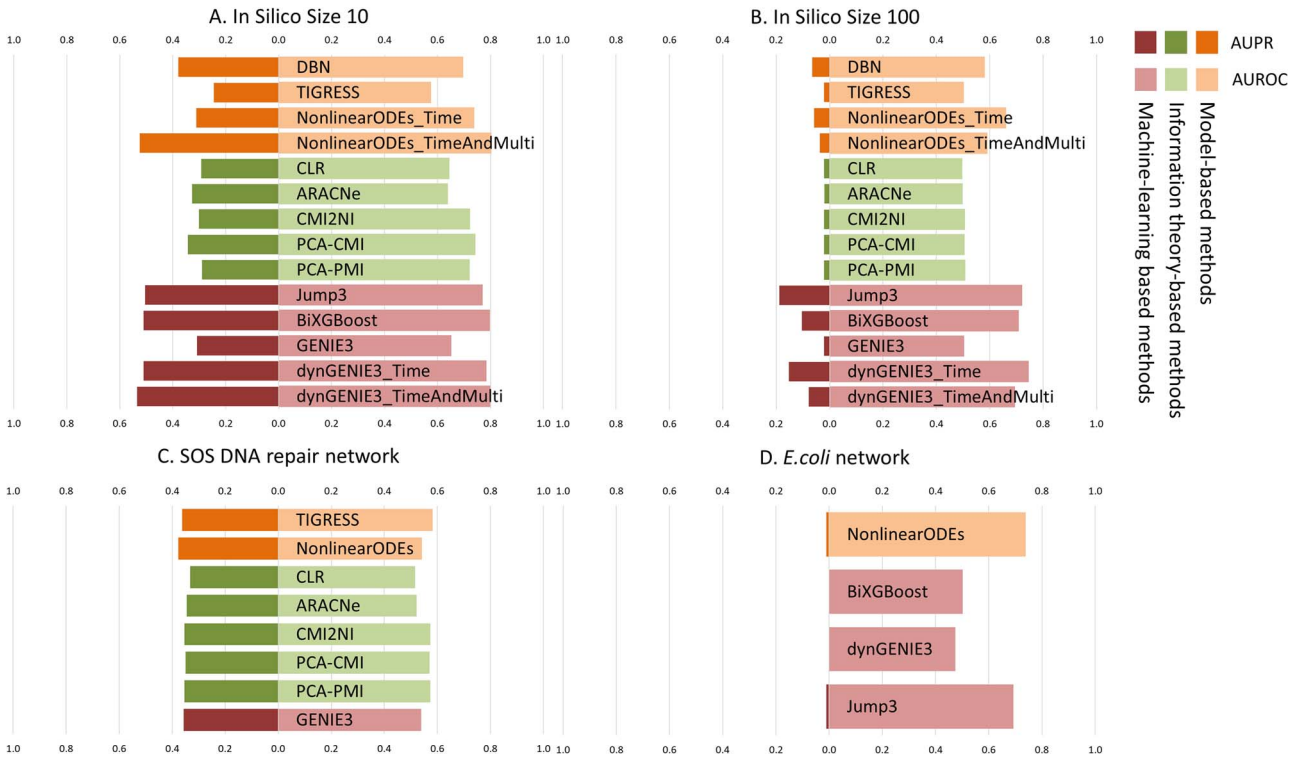

Figure 3. Overall performance of different network inference methods. (**A**) DREAM4 *in silico* size 10, (**B**) DREAM4 *in silico* size 100, (**C**) SOS DNA repair network including nine genes and (**D**) *E. coli* network including 1484 genes. Each bar represents the performance of one method in which the abscissas are the corresponding AUROC (right) and AUPR (left) values, and each color indicates same category of methods.

gene, that is, distinguishing the regulatory genes or target genes from all genes effectively. There is no direct causal relationship between the effectiveness of a certain method and its corresponding category, such as GENIE3 in machine learning-based methods and NonlinearODEs in model-based methods. The most important factor in determining the performance of one method is specific model construction and implementation under the same assumption. Inference of large GRNs often get worse AUPR values, which reflects the complexity of large networks and the difficulty of reconstruction tasks.

Performance on simulation datasets

In order to evaluate the effectiveness of different methods on simulation datasets, we use radar chart to represent the distribution of AUROC and AUPR scores on subnetworks of DREAM4 datasets (Figure 4). Details are recorded in Supplementary Tables S1–S3.

For small-scale network datasets *in silico* size 10 (Figure 4A and B), even the same method, its results on different subnetworks are quite different. Overall, NonlinearODEs_TimeAndMulti (AUROC: 0.80355, AUPR: 0.52444), BiXGBoost (AUROC: 0.79743, AUPR: 0.50980), dynGENIE3_Time (AUROC: 0.78423, AUPR: 0.51067) and dynGENIE3_TimeAndMulti (AUROC: 0.79941, AUPR: 0.53401) performs well, and it is obvious on AUPR scores. NonlinearODEs_TimeAndMulti stands out in the model-based methods, which proves that it is wise to choose the nonlinear differential equation as basic model. DynGENIE3 is an improvement on GENIE3 and overcomes the disadvantage that GENIE3 can only be applied to gene steady-state data. It takes into account changes in gene expression over time while considering gene expression levels under different experimental conditions. It can be seen that dynGENIE3 performs significantly better than GENIE3.

For large-scale network datasets *in silico* size 100 (Figure 4C and D), the same method gets similar AUROCs on five different subnetworks, which can better reflect the effective of this method. The results of Jump3 (AUROC: 0.72084, AUPR: 0.18847), BiXGBoost (AUROC: 0.70880, AUPR: 0.10406), dynGENIE3_Time (AUROC: 0.74517, AUPR: 0.15203) and dynGENIE3_TimeAndMulti (AUROC: 0.69487, AUPR: 0.07766) are slightly higher than others. They all belong to machine learning-based methods, which also prove the effectiveness of this category on large-scale networks from *in silico* datasets. However, among them, Jump3 and dynGENIE3_Time have significantly higher AUPRs. Different from most other methods, Jump3 and dynGENIE3_Time both select time-series data as input, indicating that time information of genes can effectively reflect the regulatory interactions.

From the above comparison, we find that *_TimeAndMulti performs better on *in silico* size 10; however, on *in silico* size 100, *_Time always get better results. We also observe that, as the type of input data increasing, more noise is inevitably introduced. At the same time, the genetic regulation information contained in different types of data may be conflicting; leading to the accuracy of inferred network cannot meet expectations. When inferring large-scale GRNs, compared with taking multiple types of data as input simultaneously, taking only one kind of data which contains comprehensive genetic information as input will get better results, as illustrated by the results of dynGENIE3 and NonlinearODEs here.

In order to compare the stability of each method, we visualize their AUROC values on each subnetwork of DREAM4 datasets with the form of heat map (Figure 5A and B). For small-scale network datasets *in silico* size 10 (Figure 5A),

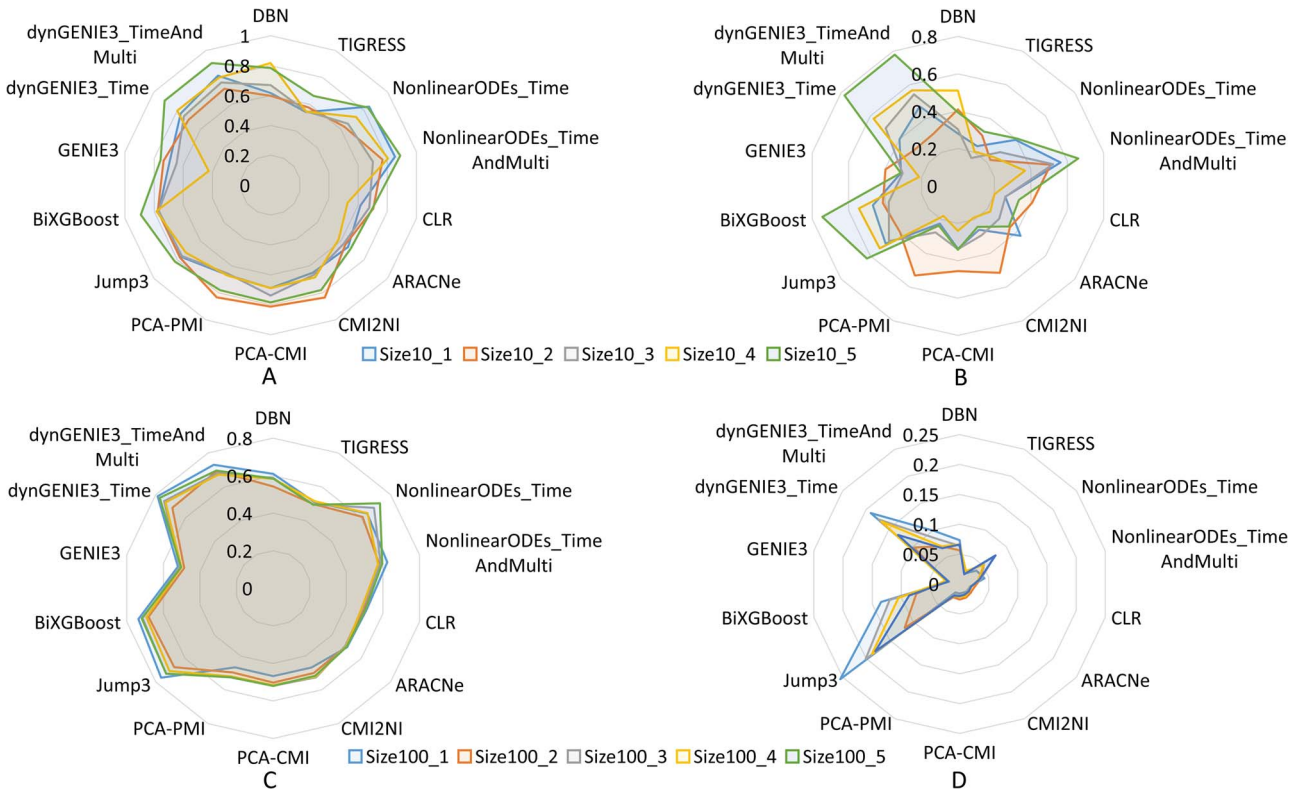

Figure 4. Results obtained from different methods on DREAM4 *in silico* networks. (**A**) AUROC values distribution on *in silico* size 10, (**B**) AUPR values distribution on *in silico* size 10, (**C**) AUROC values distribution on *in silico* size 100 and (**D**) AUPR values distribution on *in silico* size 100. In radar chart, each vertex represents one method and five curves with different colors correspond to the results on five subnetworks. Note that some methods that support both steady-state data and time-series data, such as NonlinearODEs and dynGENIE3 in our study, we set the combination of steady-state and time-series data as input to construct ∗_TimeAndMulti version and we set only time-series data as input to construct get ∗_Time version.

there is a little difference in the results of each categories while the machine learning-based methods are slightly more stable than others. Compared with the small-scale network, the AUROC of all methods on *in silico* size 100 (Figure 5B) decreases; although, the overall standard deviation is slightly smaller (see Supplementary Table S3 for details). Especially, the results of methods based on information theory are greatly reduced. In information theory, the standards to measure the relationship between variables are usually bidirectional, such as PCC, MI, etc. In other words, such methods often cannot identify the regulation direction of genes, which is the main reason for the substantial reduction of AUROC. Therefore, the information theory-based methods are more suitable for small-scale networks. In addition, machine learning-based methods can maintain robustness and accuracy as network scale expansion (except for GENIE3), so it is wise to select such methods for large networks. The heat map of AUPR values of each method on DREAM4 datasets gets the same conclusions (Supplementary Figure S1).

Performance on real datasets

For analyzing the performance on real datasets, we represent the distribution of AUROC and AUPR scores of different methods on real datasets with different sizes (Figures 6 and 7). The details are recorded in Supplementary Tables S4 and S5.

For SOS DNA repair network dataset (Figure 6), we only calculate the AUROC and AUPR values of methods applicable to gene steady-state data. From Figure 6A, we notice that the highest AUROC is obtained by TIGRESS which belongs to model-based methods. From Figure 6B, we find that the highest AUPR is arrived from NonlinearODEs which also belongs to model-based methods. This demonstrates the effectiveness of the model-based methods for constructing small scale real networks. In the methods based on information theory, whether AUROC or AUPR, PCA-PMI, PCA-CMI and CMI2NI are significantly higher than CLR and ARACNE, which shows that CMI is more efficient to measure the correlation between genes than MI.

For *E. coli* dataset (Figure 7), we calculate the results of all methods that can be applied to time-series data, which are taking the sample data obtained in different experimental environments (cold, heat and Oxidativestress) as input data. Figure 7A corresponds to AUROCs and Figure 7B corresponds to AUPRs. The machine learning-based methods (BiXGBoost, dynGENIE3 and Jump3) obtain very different results: Jump3 (AUROC: 0.69231, AUPR: 0.01073) is the better one and dynGENIE3 (AUROC: 0.47455, AUPR: 0.00132) gets the worse. DynGENIE3 utilizes RF to solve the regression problem. When constructing DTs in RF, each split node randomly selects K candidate features and then selects the best segmentation feature. In dynGENIE3, K is set to $\sqrt{P}$ where p is the number of genes. As the number of genes increases, it is more possible to ignore the critical genes, which will lead to performance degradation of the inferred network. However, if we set a larger K, time cost raises at the same time. Therefore, it is better to well fine-tune the parameters of RF before using it or to directly choose other machine learning methods such as Jump Tree from Jump3. NonlinearODEs (AUROC: 0.73830, AUPR: 0.01037) achieves best results regardless of category, except the

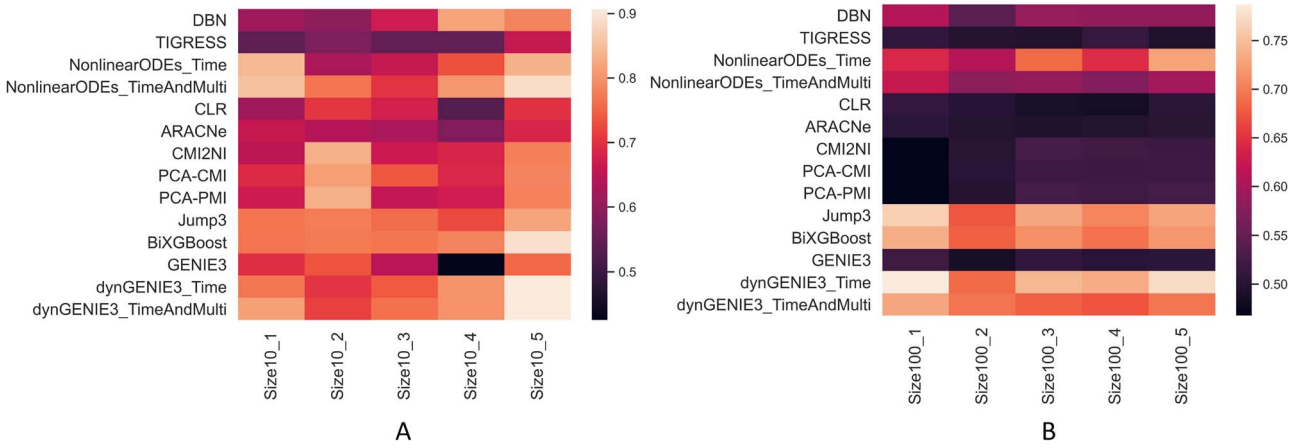

Figure 5. Stability analysis of each method on DREAM4 datasets. (**A**) Heat map of AUROC values on *in silico* size 10 and (**B**) heat map of AUROC values on *in silico* size 100. Each row corresponds to a method. Each column corresponds to a subnetwork. Each block in the heat map corresponds to one AUROC value, which means a lighter color corresponding to a higher value.

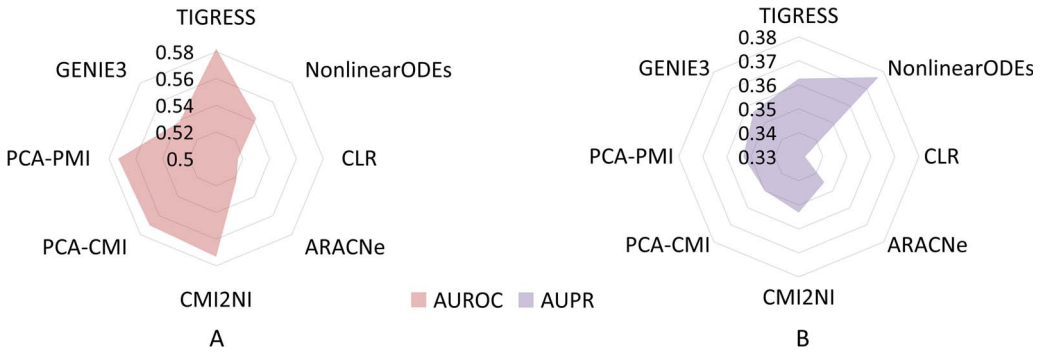

Figure 6. Results obtained from different methods on SOS DNA repair network. (**A**) AUROC and (**B**) AUPR. The value represented by polygon increases from center to edge and each vertex indicates a method.

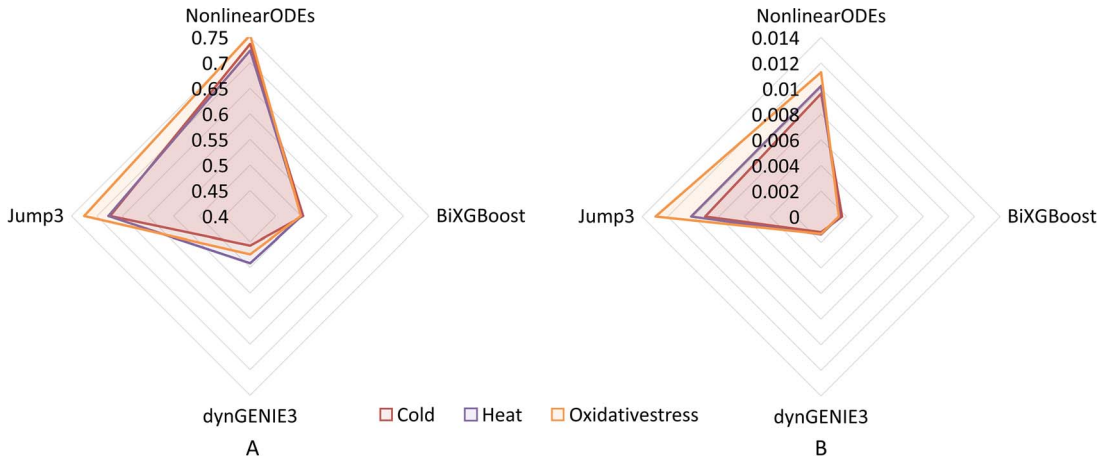

Figure 7. Results obtained from different methods on *E. coli* datasets. (**A**) AUROC and (**B**) AUPR. Different colors denote different experimental conditions (red: cold; purple: heat; orange: oxidativestress).

metric of AUPR (just less than Jump3). This proves that the selection of nonlinear ODEs as basic model is in line with the real interactions among large-scale network genes. In addition, the consideration of TFs information is another advantage for NonlinearODEs.

Obviously, the results on *E. coli* dataset are quite different from those on SOS DNA repair network dataset (Figures 6 and 7). Although the number of genes included in *E. coli* dataset is much more than SOS DNA repair network (Table 1), the highest AUROC obtained by various methods of *E. coli* dataset is much higher than SOS DNA repair network. This unexpected phenomenon is related to the inference ability of various methods, and it also proves the great potential of gene time-series data in inferring GRNs. On *E. coli* dataset, values of AUROC obtained by various

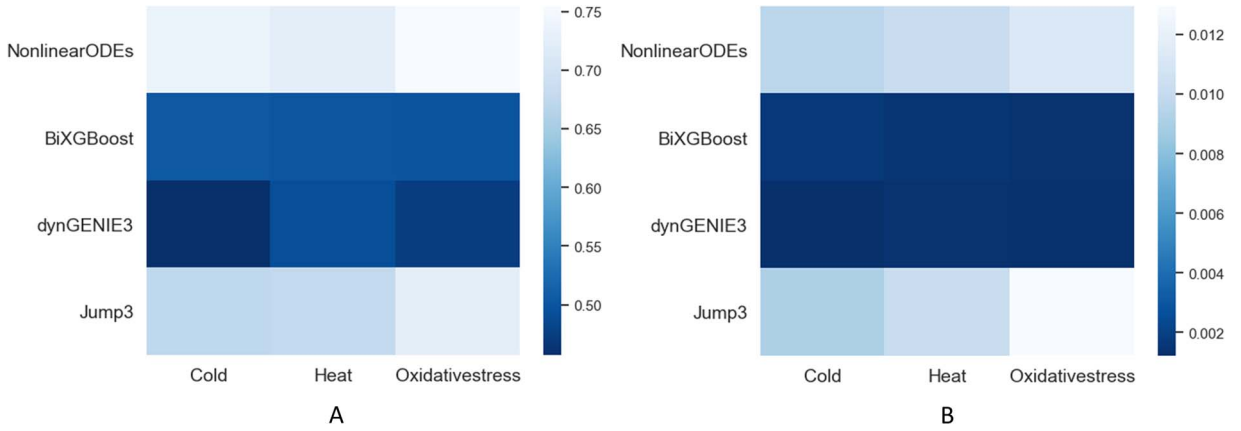

Figure 8. Stability analysis of each method on *E. coli* network. **(A)** Heat map of AUROC values and **(B)** heat map of AUPR values. Each row corresponds to a method. Each column corresponds to an experimental condition. Each block in the heat map corresponds to one value, which means a lighter color corresponding to a higher value.

Table 3. Mean and standard deviation of each method on *E. coli* dataset

Category	Method	AUROC		AUPR	
		Mean	SD	Mean	SD
Model-based methods	NonlinearODEs	***0.73830***	**0.01249**	**0.01037**	**0.00069**
Machine learning-based methods	BiXGBoost	0.50126	***0.00192***	0.00149	0.00011
	dynGENIE3	0.47455	0.01410	0.00132	***0.00007***
	Jump3	**0.69231**	0.02326	***0.01073***	0.00163

Note: Mean: bold is the maximum in the same category and italic is the maximum in all categories. SD: bold is the minimum in the same category and italic is the minimum in all categories.

methods are more different, which shows that compared with small-scale network dataset, large-scale network dataset can reflect the quality of a method more effectively and help us to evaluate a method comprehensively. Compared with SOS DNA repair network dataset, AUPRs of *E. coli* dataset is significantly lower. The main reason is that large-scale GRNs in biological systems are generally very sparse, as the density of *E. coli* dataset is 0.0014 (Table 1), which undoubtedly increases the difficulty of network inference.

To compare the stability of methods on real data, we use a heat map (Figure 8) to represent the results obtained by each method on different samples of *E. coli* network and show mean and standard deviation in Table 3. In Figure 8, we notice that although BiXGBoost is not the most accuracy method, it is the most stable method (SD of AUROC: 0.00192, SD of AUPR: 0.00011). This highlights the superiority of BiXGBoost, which profits from the effective consideration of time information and integration of a scalable and flexible gradient enhancement algorithm XGBoost. We think that NonlinearODEs and Jump3 are also good choices with the acceptable variance.

Conclusion

In this review, we classify and introduce the existing GRN reconstruction methods, which include not only classic and popular methods (DBN, GENIE3, etc.) but also newly proposed methods (NonlinearODEs, etc.). Extensive experiments are conducted to evaluate these methods based on simulation data and real data, respectively. Across such methods, we do not see an always-winner when comparing their performance based on the compositions of different data types and network scales. In order to achieve higher accuracy, we need to select appropriate inference methods based on known gene expression profiles and the characteristics of different methods.

From the analysis in previous sections, it can be found that each type of methods has a suitable application field. The model-based methods have advantages for inferring small real networks. The information theory-based methods prefer to solve the problems which are only provided steady-state data of genes. The machine learning-based methods tend to get better results than other categories on inferring large-scale networks.

For model-based methods, it is proposed to adopt nonlinear ODEs as the basic model which can capture the complex interactions among genes. And the nonlinear ODEs can be optimized by feature selection algorithm based on machine learning. For information theory-based methods, we recommend to calculate correlation depending on CMI or improved criteria, rather than directly considering MI as the base. For machine learning-based methods, we recommend to fine-tune the RF model to overcome its disadvantage when facing large-scale networks or to directly choose a more efficient machine learning algorithm. In addition, the implementation details of such methods are also important to the model quality.

Finally, for different data types provided by datasets, we recommend appropriate methods, respectively. If the dataset provides multiple types of input data, it is better to use multiple types of data as input when constructing small networks; however, it is better to take only one type of data that best reflects the interaction among genes as input when constructing large networks. When dealing with steady-state data, PCA-PMI is a good choice because of high accuracy and stability, while it is generally less effective on large scale networks. If only gene time-series data are known, information theory-based methods are not suitable because they are often only applicable to the steady-state data. In this case, we can use NonlinearODEs to

infer GRNs and it is necessary to specify parameters such as decay rate based on prior knowledge to get a more accurate result. NonlinearODEs can also be tried when the type of input data is unknown because it applies to both of them, and the results are generally good. In addition, we observe that the results on real data are generally lower than those on simulated data, which indicates that it is still a challenge to infer the GRNs of real organisms.

Availability of data and material

The datasets, codes and corresponding results are available at https://github.com/guofei-tju/Overview-of-GRN.

Funding

This work is supported by a grant from the National Natural Science Foundation of China (NSFC 61772362, 61902271 and 61972280), the National Key R&D Program of China (2020YFA0908401, 2020YFA0908402, 2020YFA0908400, 2018YFC0910405 and 2017YFC0908400).

Key Points

- We summarize three categories on the technologies of reverse engineering gene regulatory networks and make a detailed introduction respectively. We conduct a series of experiments on 12 representative methods and give thorough analysis to the comparison results.
- We conduct experiments on a wide range of datasets, which includes steady-state data and time-series data, simulated data and real data and small-scale networks and large-scale networks, so as to give comprehensive evaluation and reliable recommendations.
- According to the performance of various methods on real data, we recommend appropriate network inference technologies for different data types, network scales, etc., which provides a guidance to biologists and medical scientists for further research.

Supplementary Data

Supplementary data are available online at *Briefings in Bioinformatics*.

Acknowledgements

We would like to thank the DREAM Challenges and RegulonDB for providing extensive amounts of biological data for our research.

References

1. Wilczynski B, Furlong EEM. Challenges for modeling global gene regulatory networks during development: insights from Drosophila. *Dev Biol* 2010;**340**(2):161–9.
2. Huynh-Thu VA, Sanguinetti G. Gene regulatory network inference: an introductory survey. *Methods Mol Biol* 2019;**1883**:1–23.
3. Broeck LVD, Gordon M, Dirk Inzé, *et al*. Gene regulatory network inference: connecting plant biology and mathematical modeling. *Front Genet* 2020;**11**:457.
4. Che D, Wang Y, Bai W, *et al*. Dynamic and modular gene regulatory networks drive the development of gametogenesis. *Brief Bioinform* 2016;**18**(4):712–21.
5. Li A, Jia P, Mallik S, *et al*. Critical microRNAs and regulatory motifs in cleft palate identified by a conserved miRNA–TF–gene network approach in humans and mice. *Brief Bioinform* 2019;**21**(4):1–14.
6. Horton P. Next-generation bioinformatics: connecting bases to genes, networks and disease. *Brief Bioinform* 2014;**15**(2):137–7.
7. Iacono G, Massoni-Badosa R, Heyn H. Single-cell transcriptomics unveils gene regulatory network plasticity. *Genome Biol* 2019;**20**(1):110.
8. Oulas A, Minadakis G, Zachariou M, *et al*. Systems bioinformatics: increasing precision of computational diagnostics and therapeutics through network-based approaches. *Brief Bioinform* 2017;**20**(3):806–24.
9. Emmertstreib F, Dehmer M, Haibekains B. Gene regulatory networks and their applications: understanding biological and medical problems in terms of networks. *Front Cell Dev Biol* 2014;**2**:38.
10. Delgado-Chaves FM, Gómez-Vela F, Divina F, *et al*. Computational analysis of the global effects of Ly6E in the immune response to coronavirus infection using gene networks. *Genes* 2020;**11**(7):831.
11. Madhamshettiwar PB, Maetschke SR, Davis MJ, *et al*. Gene regulatory network inference: evaluation and application to ovarian cancer allows the prioritization of drug targets. *Genome Med* 2012;**4**(5):1–16.
12. Tong IL, Rinaldi NJ, Robert F, *et al*. Transcriptional regulatory networks in *Saccharomyces cerevisiae*. *Science* 2012;**298**(5594):799–804.
13. Yan W, Xue W, Chen J, *et al*. Biological networks for cancer candidate biomarkers discovery. *Cancer Inform* 2016;**15**:1–7.
14. Mercatelli D, Scalambra L, Triboli L, *et al*. Gene regulatory network inference resources: a practical overview. *Biochim Biophys Acta Gene Regul Mech* 2020;**1863**(6):194430.
15. Brown PO, Botstein DJNG. Exploring the new world of the genome with DNA microarrays. *Nat Genet* 1999;**21**:33–7.
16. Schena M, Shalon D, Davis RW, *et al*. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science* 1995;**270**(5235):467–70.
17. Buermans HPJ, den Dunnen JT. Next generation sequencing technology: advances and applications. *Biochim Biophys Acta* 2014;**1842**(10):1932–41.
18. Wang Z, Gerstein M, Snyder M. RNA-seq: a revolutionary tool for transcriptomics. *Nat Rev Genet* 2010;**10**(1):57–63.
19. Park PJ. ChIP-seq: advantages and challenges of a maturing technology. *Nat Rev Genet* 2009;**10**(10):669.
20. Barrett T, Wilhite SE, Ledoux P, *et al*. NCBI GEO: archive for functional genomics data sets—update. *Nucleic Acids Res* 2013;**41**(Database issue):D991–D995.
21. Brazma A, Parkinson H, Sarkans U, *et al*. ArrayExpress—a public repository for microarray gene expression data at the EBI. *Nucleic Acids Res* 2005;**33**(SI):D553–D555.
22. Gama-Castro S, Salgado H, Santos-Zavaleta A, *et al*. RegulonDB version 9.0: high-level integration of gene regulation, coexpression, motif clustering and beyond. *Nucleic Acids Res* 2015;**44**(D1):D133–43.
23. Blake JA, Chan J, Kishore R, *et al*. Gene Ontology Consortium: going forward. *Nucleic Acids Res* 2015;**43**(Database issue):1049–56.

24. Minoru K, Sato Y, Kawashima M, *et al*. KEGG as a reference resource for gene and protein annotation. *Nucleic Acids Res* 2015;**44**(D1):D457–62.
25. Feingold EA, Good PJ, Guyer MS, *et al*.. The ENCODE (encyclopedia of DNA elements) project. *Science* 2004; **306**(5696):636–640.
26. Lee W-P, Tzou W-S. Computational methods for discovering gene networks from expression data. *Brief Bioinform* 2009;**10**(4):408–23.
27. Maetschke SR, Madhamshettiwar PB, Davis MJ, *et al*. Supervised, semi-supervised and unsupervised inference of gene regulatory networks. *Brief Bioinform* 2014;**15**(2):195–211.
28. Dougherty ER. Validation of gene regulatory networks: scientific and inferential. *Brief Bioinform* 2011;**12**(3):245–52.
29. Muldoon JJ, Yu JS, Fassia M-K, *et al*. Network inference performance complexity: a consequence of topological, experimental and algorithmic determinants. *Bioinformatics* 2019;**35**(18):3421–32.
30. Villaverde AF, Banga JR. Reverse engineering and identification in systems biology: strategies, perspectives and challenges. *J R Soc Interface* 2014;**11**(91):20130505–5.
31. Saint-Antoine MM, Singh A. Network inference in systems biology: recent developments, challenges, and applications. *Curr Opin Biotechnol* 2020;**63**:89–98.
32. Chai LE, Loh SK, Low ST, *et al*. A review on the computational approaches for gene regulatory network construction. *Comput Biol Med* 2014;**48**:55–65.
33. Schlitt T, Brazma A. Current approaches to gene regulatory network modelling. *BMC Bioinform* 2007;**8**(6):1–22.
34. Marbach D, Costello JC, Küffner R, *et al*. Wisdom of crowds for robust gene network inference. *Nat Methods* 2012;**9**(8):796–804.
35. Thomas R. Boolean formalization of genetic control circuits. *J Theor Biol* 1973;**42**(3):563–585.
36. Pal R, Datta A, Dougherty ER. Optimal infinite-horizon control for probabilistic boolean networks. *IEEE Trans Signal Process* 2006;**54**(6):2375–2387.
37. Xiao Y. A tutorial on analysis and simulation of boolean gene regulatory network models. *Curr Genomics* 2009;**10**(7): 511–525.
38. Bruno-Edouard P, Ralaivola L, Mazurie A, *et al*. Gene networks inference using dynamic Bayesian networks. *Bioinformatics* 2003;**19**(Suppl_2):ii138–48.
39. Kim SY, Imoto S, Miyano S. Inferring gene networks from time series microarray data using dynamic Bayesian networks. *Brief Bioinform* 2003;**4**(3):228–35.
40. Sanchezcastillo M, Blanco D, Tienda-Luna IM, *et al*. A Bayesian framework for the inference of gene regulatory networks from time and pseudo-time series data. *Bioinformatics* 2018;**34**(6):964–70.
41. Liu F, Zhang SW, Guo WF, *et al*. Inference of gene regulatory network based on local Bayesian networks. *PLoS Comput Biol* 2016;**12**(8):e1005024.
42. Haury A-C, Mordelet F, Vera-Licona P, *et al*. TIGRESS: trustful inference of gene regulation using stability selection. *BMC Syst Biol* 2012;**6**(1):145.
43. Matsumoto H, Kiryu H, Furusawa C, *et al*. SCODE: an efficient regulatory network inference algorithm from single-cell RNA-seq during differentiation. *Bioinformatics* 2017;**33**(15):2314–21.
44. Fan A, Wang H, Xiang H, *et al*. Inferring large-scale gene regulatory networks using a randomized algorithm based on singular value decomposition. *IEEE/ACM Trans Comput Biol Bioinform* 2018;**16**(6):1997–2008.
45. Ma B, Fang M, Jiao X. Inference of gene regulatory networks based on nonlinear ordinary differential equations. *Bioinformatics* 2020;**36**(19):4885–4893.
46. Tsai MJ, Wang JR, Hoe SJ, *et al*. GREMA: modelling of emulated gene regulatory networks with confidence levels based on evolutionary intelligence to cope with the underdetermined problem. *Bioinformatics* 2020;**36**(12):3833–40.
47. Vohradsky J. Neural model of the genetic network. *J Biol Chem* 2001;**276**(39):36168–73.
48. Ressom HW, Zhang Y, Xuan J, *et al*. Inference of gene regulatory networks from time course gene expression data using neural networks and swarm intelligence. In: *IEEE Symposium on Computational Intelligence and Bioinformatics and Computational Biology*. IEEE, 2006;1–8.
49. Yang Y, Fang Q, Shen HB. Predicting gene regulatory interactions based on spatial gene expression data and deep learning. *PLoS Comput Biol* 2019;**15**(9):e1007324.
50. Yuan Y, Bar-Joseph Z. Deep learning for inferring gene relationships from single-cell expression data. *Proc Natl Acad Sci USA* 2019;**116**(52):27151–8.
51. Song L, Langfelder P, Horvath S. Comparison of co-expression measures: mutual information, correlation, and model based indices. *BMC Bioinform* 2012;**13**(1):328–8.
52. Zhang X, Liu K, Liu Z, *et al*. NARROMI: a noise and redundancy reduction technique improves accuracy of gene regulatory network inference. *Bioinformatics* 2013;**29**(1):106–13.
53. Basso K, Margolin AA, Stolovitzky G, *et al*. Reverse engineering of regulatory networks in human B cells. *Nat Genet* 2005;**37**(4):382–90.
54. Faith JJ, Hayete B, Thaden JT, *et al*. Large-scale mapping and validation of *Escherichia coli* transcriptional regulation from a compendium of expression profiles. *PLoS Biol* 2007;**5**(1):54–66.
55. Joshua MS, Segal E, Koller D, *et al*. A gene-coexpression network for global discovery of conserved genetic modules. *Science* 2003;**302**(5643):249–55.
56. Yang B, Xu Y, Maxwell A, *et al*. MICRAT: a novel algorithm for inferring gene regulatory networks using time series gene expression data. *BMC Syst Biol* 2018;**12**(S7):115.
57. Zhao J, Zhou Y, Zhang X, *et al*. Part mutual information for quantifying direct associations in networks. *Proc Natl Acad Sci USA* 2016;**113**(18):5130–5135.
58. Zhang X, Zhao J, Hao JK, *et al*. Conditional mutual inclusive information enables accurate quantification of associations in gene regulatory networks. *Nucleic Acids Res* 2015; **43**:e31.
59. Camacho DM, Collins KM, Powers RK, *et al*. Next-generation machine learning for biological networks. *Cell* 2018;**173**(7):1581–92.
60. Li Y, Wu F-X, Ngom A. A review on machine learning principles for multi-view biological data integration. *Brief Bioinform* 2016;**19**(2):325–40.
61. Magnusson R, Gustafsson M. LiPLike: towards gene regulatory network predictions of high certainty. *Bioinformatics* 2020;**36**(8):2522–9.
62. Huynh-Thu VA, Geurts P. dynGENIE3: dynamical GENIE3 for the inference of gene networks from time series expression data. *Sci Rep* 2018;**8**(1):3384.
63. Huynh-Thu VNA, Irrthum A, Wehenkel L, *et al*. Inferring regulatory networks from expression data using tree-based methods. *PLoS One* 2010;**5**(9):e12776.
64. Che D, Guo S, Jiang Q, *et al*. PFBNet: a priori-fused boosting method for gene regulatory network inference. *BMC Bioinform* 2020;**21**(1):308.

65. Aibar S, González-Blas CB, Moerman T, *et al*. SCENIC: single-cell regulatory network inference and clustering. *Nat Methods* 2017;**14**(11):1083–6.
66. Moerman T, Santos SA, González-Blas CB, *et al*. GRNBoost2 and Arboreto: efficient and scalable inference of gene regulatory networks. *Bioinformatics* 2018;**35**(12):2159–61.
67. Razaghi-Moghadam Z, Nikoloski Z. Supervised learning of gene-regulatory networks based on graph distance profiles of transcriptomics data. *NPJ Syst Biol Appl* 2020;**6**(1):21.
68. Zhang Y, Zhang XF, Lane AN, *et al*. Inferring gene regulatory networks of metabolic enzymes using gradient boosted trees. *IEEE J Biomed Health Inform* 2020;**24**(5):1528–36.
69. Zheng R, Li M, Chen X, *et al*. BiXGBoost: a scalable, flexible boosting based method for reconstructing gene regulatory networks. *Bioinformatics* 2019;**35**(11):1893–900.
70. Huynh-Thu VA, Guido S. Combining tree-based and dynamical systems for the inference of gene regulatory networks. *Bioinformatics* 2015;**31**(10):1614–22.
71. Mordelet F, Vert J-P. SIRENE: supervised inference of regulatory networks. *Bioinformatics* 2008;**24**(16):I76–82.
72. Marbach D, Prill RJ, Schaffter T, *et al*. Revealing strengths and weaknesses of methods for gene network inference. *Proc Natl Acad Sci USA* 2010;**107**(14):6286–91.
73. Daniel M, Schaffter T, Mattiussi C, *et al*. Generating realistic *in silico* gene networks for performance assessment of reverse engineering methods. *J Comput Biol* 2009;**2**(16):229–39.
74. Stolovitzky G, Monroe D, Califano A. Dialogue on reverse-engineering assessment and methods: the DREAM of high-throughput pathway inference. *Ann N Y Acad Sci* 2008;**1115**:1–22.
75. Stolovitzky G, Prill RJ, Califano A. Lessons from the DREAM2 challenges. *Ann N Y Acad Sci* 2009;**1158**:159–95.
76. Schaffter T, Marbach D, Floreano D. GeneNetWeaver: *in silico* benchmark generation and performance profiling of network inference methods. *Bioinformatics* 2011;**27**(16):2263–70.
77. Ronen M, Rosenberg R, Shraiman BI, *et al*. Assigning numbers to the arrows: parameterizing a gene regulation network by using accurate expression kinetics. *Proc Natl Acad Sci USA* 2002;**99**(16):10555–60.
78. Shen-Orr SS, Milo R, Mangan S, *et al*. Network motifs in the transcriptional regulation network of *Escherichia coli*. *Nat Genet* 2002;**31**(1):64–8.
79. Jozefczuk S, Klie S, Catchpole G, *et al*. Metabolomic and transcriptomic stress response of *Escherichia coli*. *Mol Syst Biol* 2010;**6**(1):364.
80. Bilmes J. A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models Technical Report ICSI-TR-97-021, University of Berkeley. Vol. 4. 1998.
81. Efron B, Hastie T, Johnstone I, *et al*. Least angle regression. *Ann Statist* 2004;**32**(2):407–51.
82. Meinshausen N, Bühlmann P. Stability selection. *J R Statist Soc* 2010;**72**(4):417–73.
83. Qian L, Wang H, Dougherty ER. Inference of noisy nonlinear differential equation models for gene regulatory networks using genetic programming and Kalman filtering. *IEEE Trans Signal Process* 2008;**56**(7):3327–39.
84. Zhang X, Zhao X-M, He K, *et al*. Inferring gene regulatory networks from gene expression data by path consistency algorithm based on conditional mutual information. *Bioinformatics* 2012;**28**:98–104.
85. Andrea O, Millar AJ, Guido S. Hybrid regulatory models: a statistically tractable approach to model regulatory network dynamics. *Bioinformatics* 2013;**29**(7):910–6.
86. Han J-DJ, Bertin N, Hao T, *et al*. Evidence for dynamically organized modularity in the yeast protein–protein interaction network. *Nature* 2004;**430**(6995):88–93.
87. Jeong H, Tombor B, Albert R, *et al*. The large-scale organization of metabolic networks. *Nature* 2000;**407**(6804):651–4.