Autoregressive models for gene regulatory network inference: Sparsity, stability and causality issues

George Michailidis, Florence d'Alché-Buc

Autoregressive models for gene regulatory network inference: Sparsity, stability and causality issues

George Michailidis [a], Florence d'Alché-Buc [b,c,*]

[a] *Department of Statistics, University of Michigan, Ann Arbor, MI 48109-1107, USA*
[b] *INRIA-Saclay, AMIB, TAO, LRI umr CNRS 8326, Orsay, France*
[c] *IBISC, EA 4526, Université d'Evry, Evry, France*

Reconstructing gene regulatory networks from high-throughput measurements represents a key problem in functional genomics. It also represents a canonical learning problem and thus has attracted a lot of attention in both the informatics and the statistical learning literature. Numerous approaches have been proposed, ranging from simple clustering to rather involved dynamic Bayesian network modeling, as well as hybrid ones that combine a number of modeling steps, such as employing ordinary differential equations coupled with genome annotation. These approaches are tailored to the type of data being employed. Available data sources include static steady state data and time course data obtained either for wild type phenotypes or from perturbation experiments.

This review focuses on the class of autoregressive models using time course data for inferring gene regulatory networks. The central themes of sparsity, stability and causality are discussed as well as the ability to integrate prior knowledge for successful use of these models for the learning task at hand.

1. Introduction

A number of technological advances, such as DNA microarrays, RNA-Seq [1], liquid chromatography tandem mass spectrometry [2], and similarly liquid or gaseous chromatography mass spectrometry [3], have enabled biomedical researchers to collect large amounts of transcriptomic, proteomic and metabolomic data. In addition, curated repositories containing both vast amounts of such data, as well as functional information, ontologies, gene and protein interactions, pathways, etc. are expanding at a fast pace (e.g. KEGG, IntegromeDB, BioGrid, GEO, NURSA, etc.).

The increasing availability of such high dimensional data and structured information have led to a number of novel learning problems, including that of *network inference*. Networks have become a key tool in computational biology due to their ability to capture at an appropriate level of abstraction biological processes. Overall, the study of biological networks including modeling, analysis, reconstruction and visualization aspects has become a key topic in bioinformatics and computational biology (for a review and recent trends see [4]).

A number of learning tasks have been studied in the literature, based on the type of biological network under consideration. For example, in metabolic reaction networks, the focus has been on learning enzyme kinetic parameters [5], stoichiometric analysis, as well as finding the operative modes of such networks subject to catalytic activity and steady state operational constraints. In protein interaction networks, predictions of interactions are based both on protein descriptors and labeled edges [6]. Information obtained from protein–protein interaction networks has proved useful in protein function prediction and in learning protein complexes [7], while predicting cellular responses using ontology information has been a key task involving signaling networks. In this review study, we focus on the problem of reconstructing (inferring) the structure of gene regulatory networks (GRN). Such networks involve interactions between DNA, RNA, proteins and other biomolecules, whose edges represent functional influences of one molecule on the other, rather than chemical interactions.

This learning task has become a central one in functional genomics, as the growing literature on the subject attests [8–10]. Two main types of data have been used to learn such networks: steady state data and time course data. steady state data are obtained from a long-term observation of gene expression, assuming the system reaches an equilibrium state. For instance, multiple biological replicates obtained at some late point in time provide such steady state data. Such data are usually obtained from microarray technologies, and provide a global view of the biological system under study in its natural state (wild type); however, their informational content for network reconstruction purposes is in general limited and accurate network inference usually requires a very

* Corresponding author at: IBISC, EA4526, 23, Bd de France, Université d'Evry et Genopole, 91037 cedex Evry, France. Tel.: +33 164853164.
E-mail address: florence.dalche@ibisc.fr (F. d'Alché-Buc).

large number of replicates [11]. On the other hand, time course data even for wild type measurements provide insights on the transitory behavior of the biological system which is induced by regulations, especially if the system is observed under different initial conditions due to perturbations, as discussed next.

A particularly informative source for the learning task at hand is data from perturbation experiments, involving specific gene knock-outs/downs or silencing. They may correspond to a single time observation point, selected so that the perturbation has manifested itself in the system, or could take the form of time series, as discussed above. The advantage of time course data obtained from perturbation experiments is that they contain significant information about the dynamics of the system and are shown to be a key component for network inference in the DREAM7 challenge on experimental design for parameter estimation in network models (more information regarding the DREAM challenge competition is provided in Section 9). The downside of perturbation data is that they are usually obtained from single gene knock-outs (downs). Hence, every replicate (time series or single time point observation) offers limited information about the overall system, especially when joint regulations are involved. Moreover, large scale perturbation experiments for most organisms are not readily available, due to technical complexities and cost considerations.

On the other hand, wild type time course data are still attractive for inferring relatively large scale GRNs, since they contain adequate information about regulatory interactions and are significantly less expensive to acquire compared to perturbation data.

For inferring GRNs, the majority of approaches in the literature belong to the class of *unsupervised methods*, although there has been work that assumes partial knowledge of the network which is either integrated as prior information in the model employed, or used in a *supervised approach* [12,13]. The class of unsupervised approaches can be divided in the following two categories: (i) *model-based* ones that aim to capture the dynamical behavior of the GRN by estimating the parameters of a chosen model [14–17,8,9,18–20], and (ii) *model-free* approaches that extract dependencies among state variables using information-theoretic criteria in the spirit of ARACNE [15,21,22].

This review primarily focuses on inferring GRNs from time course data and model-based approaches. Our goal is to emphasize the key elements that are common in the best off-the-shelf network inference algorithms and to outline the set of important features that such algorithms should possess to meet future challenges. A key feature is that of *sparsity*, due to the following facts. First, statistical analysis of known regulatory networks has shown that scale-free models are suitable to represent the topological structure of the network, thus reflecting their sparse nature. Second, most available data sets contain relatively few time points compared to the number of genes measured, thus making the use of sparse models obligatory. Another key element in network inference (and in learning complex structures in general) is that of *stability* of the algorithm. The concept of stability has been central for model selection in regularized regression [23] or as a construction principle in various randomized models, including bagging and random forests. Recent works explore the use of this concept in GRN inference [24,25]. Taking another angle, the ability to integrate prior knowledge into a model or in a learning method represents a valuable property in a field where partial knowledge coming from different sources may be available [26]. Finally a key question regarding network inference is the semantics associated with a direct edge in a regulation graph. Directed edges under certain conditions reflect causal relationships [27]. Even though estimating such relationships is known to be a very challenging task, *causality* nevertheless represents a central issue in network inference.

The remainder of the paper is organized as follows. Section 2 presents the problem of gene regulatory network inference from time course data and emphasizes desirable properties of a network structure inferred by a learning method. Section 3 gives an overview of the main Markov models used for network inference from time course data. In Sections 4 and 5, different works about Markov models and their associated network inference methods are reviewed and when it is possible, analyzed through the concepts of sparsity and causality. Section 4 focuses on linear autoregressive models for which sparse regression has been largely developed and from which Granger-causal networks can be inferred. Extensions of linear autoregressive models described in Section 5 consider generalized additive models and kernel-based methods. Section 6 gives a brief presentation of dynamic Bayesian networks that support, as a special case of autoregressive models, specific learning strategies. In Section 7, we highlight the notion of stability and describe how it has been recently used for model selection and to improve upon a base model. Section 8 addresses prior integration in the whole set of reviewed models, while Section 9 provides an overview of the performance of various methods in the DREAM computational challenges. Finally, Section 10 discusses recent trends and future challenges.

2. Gene regulatory network inference from time course data

In model-based approaches to network inference, a GRN is abstracted and considered as a dynamical system whose states correspond to different mRNA concentrations. The network structure is defined as a directed graph $\mathcal{G}$ whose nodes are associated to genes and whose directed edges represent the presence or the absence of regulations[1] from one regulating gene to a target gene. In the paper, $|\mathcal{G}| = p$ denotes the number of genes and A, a binary matrix of size $p \times p$, is the adjacency matrix of graph $\mathcal{G}$.

Assuming that we observe gene expression levels for wild type, we denote by $\mathbf{x}_T$ the p-dimensional vector of the gene expression levels measured at time T. Gene regulatory network inference consists in providing an estimate of A denoted by $\widehat{A}$, given the time course $\mathcal{S}_n = \{\mathbf{x}_0, \ldots, \mathbf{x}_{n-1}\}$ of length n measured at equidistant time points $t_0, \ldots, t_{n-1}$, with $t_i = t_{i-1} + \tau,\ i = 1, \ldots, n-1$. In the case the time points are not regularly spaced, which happens rather frequently in biological experiments, the observations are smoothed by a nonparametric regression which is re-sampled subsequently. The sampling rate τ is in this case an additional hyperparameter of any discrete-time modeling. This estimation task is by definition unsupervised unless partial knowledge about the graph is available. The main part of the paper is devoted to the case when no edges are known. However, integration of prior knowledge will be discussed in detail in Section 8.

Model-based approaches mainly proceed in two steps: first, given a model of the dynamical system $\mathcal{M}$, they estimate its parameters from observed time course and second, they extract from its parameters an estimate $\widehat{A}$ of the target matrix A. In some cases, like in Dynamical Bayesian Learning, the network structure is included in the parameter set and the second step is straightforward.

2.1. Desirable properties for the estimated network

Let us discuss the properties for the network structure estimated from model $\mathcal{M}$ and time course data $\mathcal{S}_n$. Beyond structure consistency, which will not be discussed here per se, other properties related to what biologists expect from the automated inference process can be targeted. They include network sparsity and stability of the algorithm employed. Next, we provide a high level

[1] For sake of simplicity, we will only consider transcriptional regulations.

overview of these key concepts, while a more detailed presentation is given in the next sections.

Sparse networks: A network is said to be sparse if the number of edges in a network is very small compared to the number of possible edges. Network inference methods search for sparse networks for different reasons: first, the unfavorable ratio between the complexity of models compared to the limited size of available time course data creates difficulties to any statistical inference method. So, only a rigorous control of the model's complexity can help to avoid overfitting. Second, when learning continuous estimates of matrix A, for instance within the autoregression framework, we wish to be able to clearly extract edges from continuous coefficients. This extraction process is similar to feature selection, so at the end of the learning process, the estimated matrix coefficients should be as close as possible to zero or to one in absolute value. Third, there is strong evidence that gene regulatory networks as well as more general biochemical networks are sparse [28,29]. This has motivated researchers to use techniques that encourage sparsity in the reconstructed network. A prime tool to achieve this objective is penalized regression, which has been extensively used with most of the models discussed in subsequent sections.

Causal networks: In order to better understand regulatory mechanisms, biologists aim at identifying causal influence of one component upon another. However, inferring causal relationships between two variables from data, as opposed to measuring associations or correlations, is one of the most challenging issues in empirical inference. Causality, which has attracted a lot of attention in all experimental sciences, supports many definitions; in this paper we retained two of them that have been effectively used for network inference: Granger causality [30], originally defined for linear autoregressive models in economics, and causality such as defined in graphical causal modeling by Pearl [27]. Applications to causal network inference will be discussed in Section 4 devoted to linear autoregressive models and in Section 6, which briefly describes Dynamic Bayesian Networks.

Stable networks: Stability of an estimation procedure refers to its ability to provide similar output (in an appropriate metric) when fed with closely related training data. In the specific problem of network estimation from time course data, this means that two adjacency matrices inferred from two appropriately bootstrapped samples are close enough. As shown in Section 7, stability has been recently used in network inference either as a model selection method or as a building principle using randomization.

3. Overview of Markov models for gene regulatory network inference

A large number of modeling approaches [31] have been developed to describe the behavior of a GRN. Formal approaches use discrete descriptions of the network using automata, Petri Nets and Computation Tree Logic that allow to analyze the discrete dynamics and perform model-checking [32]. There exist a few attempts to learn these models from data, although estimation generally involves combinatorial optimization, a challenging issue with high computational cost. Quantitative models, including ordinary differential equations (ODE), dynamical Bayesian networks, autoregressive and state-space models are more popular in the context of GRN inference. Characterized by a set of parameters, they can be estimated using continuous mathematical programming tools and other optimization tools devoted to linear or nonlinear modeling. Major achievements in network inference with linear ODEs were obtained by several groups [16], while nonlinear ODEs, whose parameters can be easily interpreted as in S-systems [33–35], have proven to be also appealing for network inference. The interested reader will find a complete review about the use of ODE based modeling in [9]. In this review, we focus on discrete-time probabilistic quantitative models, whose advantages are that they can take into account measurement noise, while estimation of their parameters does not involve computationally intensive integration steps. These models belong to the family of Markov models, where the future depends on the past only through the present state and possibly a few directly preceding states. We start by introducing some notation. We denote by $\mathbf{x}_T$ the p-dimensional state vector of the network under study at time T, where p corresponds to the number of genes in the system under consideration. Next, we give a quick overview of probabilistic models.

3.1. First order Markov models

A first order autoregressive model is defined by the following equation:

$$\forall T \geqslant 1, \quad \mathbf{x}_T = F(\mathbf{x}_{T-1}) + \boldsymbol{\epsilon}_T, \tag{1}$$

where F is a deterministic function from $\mathbb{R}^p$ to $\mathbb{R}^p$ and $\boldsymbol{\epsilon}_T$ is a noise term, usually assumed to be Gaussian: $\mathcal{N}(0, \sigma^2 I)$. Moreover, the $\boldsymbol{\epsilon}_T$'s are assumed to be independent.

Alternatively the same model can be described by the joint probability density of the state vectors over all different timepoints $T = 0, \ldots, n-1$, which in the Gaussian case is given by

$$p(\mathbf{x}_0, \ldots, \mathbf{x}_{n-1}) = p(\mathbf{x}_0) \prod_{T=1}^{n-1} p(\mathbf{x}_T | \mathbf{x}_{T-1}) \tag{2}$$

with $p(\mathbf{x}_T|\mathbf{x}_{T-1})$ being a multivariate Gaussian density on the vector: $\mathbf{x}_T - F(\mathbf{x}_{T-1})$.

Such a model can be decomposed into p one-dimensional models:

$$\forall i \in \{1, \ldots p\}, \quad x_T^i = f_i(x_{T-1}^1, \ldots, x_{T-1}^p) + \epsilon_T^i. \tag{3}$$

Dynamical Bayesian Networks also fit into a similar framework, but with the following restriction: each one-dimensional variable i only depends on a set of parent variables measured at time $(T-1)$:

$$\forall i \in \{1, \ldots p\}, \quad x_T^i = f_i(Pa(i)_{T-1}) + \epsilon_T^i. \tag{4}$$

A number of studies have focused on first order Markov models with a linear function F. Although GRNs do not exhibit linear behavior, especially due to saturation effects, the linearity assumption offers two main advantages: it enables looking at larger dimensional spaces than nonlinear models given the limited amount of data usually available and also the coefficients of the linear function admit a straightforward interpretation in terms of conditional independence. Then, the model can be written as:

$$\forall i \in \{1, \ldots p\}, \quad \mathbf{x}_T = A\mathbf{x}_{T-1} + \boldsymbol{\epsilon}_T, \tag{5}$$

where $\boldsymbol{\epsilon}_T$ is a p-dimensional Gaussian vector, A a $p \times p$ matrix, and consequently $\mathbf{x}_T$ is also a p-dimensional Gaussian vector. It can also be written as a collection of p one-dimensional autoregressive models of the form:

$$\forall i \in \{1, \ldots p\}, \quad x_T^i = \sum_{j=1}^{p} a_{ij} x_{T-1}^i + \epsilon_T^i. \tag{6}$$

3.2. Markov models of order m

Models of order 1, as those presented above are rather restrictive. Regulatory events may occur at different rates and higher order interactions can better capture the dynamics of the system under study. There has been some work in the literature

that considers the problem of GRN inference by employing higher order autoregressive models, especially in the linear case [36,37].

4. Linear autoregressive models for network inference

Linear models represent a popular class for learning GRNs from time course data, especially when coupled with the concept of Granger causality developed in the econometrics literature [30]. In this framework, interactions amongst variables (genes) are defined if past observations of one variable result in improved predictions of other variables. Specifically, let $\{x_T\}_{T=0}^n$ and $\{y_T\}_{T=0}^n$ be observations from two time series. Then, x is considered to be Granger-causal for y if the model

$$x_T = \sum_{t=1}^{q_1} a_t x_{T-t} + \sum_{t=1}^{q_2} b_t y_{T-t} + u_T, \quad q_1, q_2 \leqslant T, \tag{7}$$

significantly outperforms the model

$$x_T = \sum_{t=1}^{q_1} a_t x_{T-t} + e_T, \quad q_1 \leqslant T, \tag{8}$$

in terms of predictive ability as measured by the Predictive Mean Squared Error. Models as described in Eqs. (7) and (8) are based on a sequential strategy, where one keeps adding terms to the model and testing its overall significance through an F-test.

Graphical Granger causal models extend the notion of Granger causality to p variables. Specifically, define a vector time series $\mathbf{x}_t = (x_t^1, \ldots, x_t^p)$ and consider the d-th order corresponding vector auto-regressive (VAR) model [38, Chapter 2]:

$$\mathbf{x}_T = A_1 \mathbf{x}_{T-1} + \ldots A_d \mathbf{x}_{T-d} + \varepsilon_T, \tag{9}$$

where $A_t,\ t = 1, \ldots, d$, are $p \times p$ matrices whose coefficients represent the magnitude of interaction effects among variables at different time points. In this model formulation, variable x_{T-t}^j is considered Granger-causal for x_T^i, if the corresponding coefficient, $(A_t)_{ij} = a_{t,i,j}$ is statistically significant.

One of the first studies to use the notion of Granger causality for the estimation of a regulatory network has been [39]. However, due to lack of adequate samples and to overcome computational issues, instead of looking at the full model of Eq. (9), that study adopted a simplified strategy that examines pairs of genes one at a time for Granger causality. This results in testing $p(p-1)/2$ hypotheses and a strategy for addressing the multiple comparisons problem and thus controlling the familywise error rate of the test is used; [39] employed the False Discovery Rate procedure introduced by Benjamini and Hochberg [40].

It can be seen that in order to estimate the full model of Eq. (9), one must have $T > dp^2$ (more time points than the number of effects to be estimated); in the presence of n independent replicates of the time course data, the condition becomes $Tn > dp^2$. This is a strong requirement for any network containing $p = 30$ or more genes. Nevertheless, extensive research [41] indicates that GRNs are relatively sparse and hence it is appropriate to estimate a *sparse* VAR model. To do so, [17] focused on an order $d = 1$ VAR model and employed a Lasso penalty leading to the following loss function

$$\sum_{T=1}^{n} ||\mathbf{x}_T - A_1 \mathbf{x}_{T-1}||^2 + \Omega(A_1), \tag{10}$$

with $\Omega(A_1) = \lambda \sum_{i=1}^p \sum_{j=1}^p |a_{1,i,j}|$ and $\lambda > 0$ being a tuning parameter, selected either through cross-validation or through a grid search over a range of values followed by inspection of the structure of the resulting network. In an extension, and to overcome the impact of high correlations between the time series that leads to numerical instabilities, [42] used an *elastic net penalty* of the form $\Omega(A_1) = \lambda_1 \sum_{i=1}^p \sum_{j=1}^p |a_{1,ij}| + \lambda_2 \sum_{i=1}^p \sum_{j=1}^p (a_{1,i,j})^2$. A potential challenge for fitting this model is the need to search for two tuning parameters.

A shortcoming of these and related studies (e.g. [43]) is that they consider a *single time lag* in the VAR model. To rectify this, [37] proposed a different formulation of the model in Eq. (9). Specifically, it proposed fitting the following loss function

$$||\mathbf{x}_T - \sum_{t=1}^{T-1} A_t \mathbf{x}_{T-t}||^2 + \Omega(A), \tag{11}$$

where $\Omega(A) = \lambda \sum_{j=1, j\neq i}^p \sqrt{\sum_{t=1}^{T-1} (a_{t,i,j})^2}$ is the so-called group lasso penalty. The idea behind this penalty is that it forces simultaneous selection of all the members of the group, which in the present context implies including all the time lags of variable x^j. The shortcoming of this modeling approach lies in the fact that lags from the distant past may not be relevant in the estimation of Granger causal effects.

In a series of papers [36,19], *high-order sparse* VAR models were considered and estimated by employing different variable selection strategies. In this line of work, the problem is cast as that of estimating a Directed Acyclic Graph with a known ordering [18], which provides a general framework for addressing theoretical issues, such as prediction and selection consistency of the estimated Granger causal effects and subsequently of the resulting network structure.

In [36], a *truncating Lasso* penalty was employed, in an effort to identify the correct order of the VAR model. Specifically, the following loss function is optimized for $i = 1, \ldots, p$:

$$\arg\min_{\theta^t \in \mathbb{R}^p} n^{-1} \|x_T^i - \sum_{t=1}^{d} A_t \mathbf{x}_{T-t}\|_2^2 + \lambda \sum_{t=1}^{d} \Psi_t \sum_{j=1}^{p} |A_t^j| w_t^j, \tag{12}$$

$$\Psi_1 = 1, \quad \Psi_t = MI\{\|A_{(t-1)}\|_0 < p^2 \frac{\gamma}{(T-1)}\},\ t \geqslant 2,$$

where M is a large constant, and γ is the allowed false negative rate. In [36], a block-coordinate descent algorithm for solving this optimization problem is introduced and the following theoretical properties of the resulting estimated Granger effects established: (i) the proposed penalty gives a consistent estimate of the order of the underlying VAR model and (ii) its structure is consistently estimated under certain regulatory conditions, provided that the effects A_{ij}^t *decay* over time.

For the case where the latter decay assumption is violated, a thresholding strategy was adopted in [19], where the Granger causal effects obtained from a lasso penalized regression are subsequently thresholded. In this strategy, two tuning parameters need to be fixed: the penalty for the lasso regression step and the threshold. Theoretical work establishes consistent estimation of (i) the order of the VAR model and (ii) the structure of the underlying network under a *restricted eigenvalue* condition for the design matrix of the regression problem. The latter condition is a standard one in inference for high-dimensional regression, classification and learning Gaussian and Markov graphical model problems [44].

Finally, in [19] both the truncating lasso and the thresholding strategies are complemented with a group lasso term, in order to incorporate *externally given* pathway information in this learning problem.

5. Nonlinear autoregressive models

Living systems with feedback loops do not always exhibit linear dynamical behavior. This is the case in GRNs for which we observe nonlinearities in a large range of situations. Within the framework

of autoregressive models, it is possible to estimate a nonlinear model and subsequently extract its network structure without assumptions about the shape of the model. We present here two examples of nonlinear semiparametric autoregressive models where sparsity is imposed to models in order to extract networks.

5.1. Nonlinear semiparametric models with splines and kernels

In order to model nonlinear interactions, [45,46] have introduced a semiparametric nonlinear autoregressive model,[2] defined as follows:

$$\forall i \in \{1,\dots p\}, \quad x_T^i = \eta_T^i(\mathbf{x}_{T-1};\boldsymbol{\beta}^i) + \epsilon_T^i, \tag{13}$$

where ϵ_T^i is a Gaussian isotropic noise field and the function η_T^i can be written as a sum of B-spline basis functions using coefficients of matrix $\boldsymbol{\beta}^i$:

$$\forall i \in \{1,\dots p\}, \quad \eta_T^i(\mathbf{x}_{T-1};\boldsymbol{\beta}^i) = f_{i1}(x_T^1) + \cdots + f_{ip}(x_T^p) + \mu^i, \tag{14}$$

with μ^i being a gene-specific term, $f_{ij}(x_T^j) = \sum_{k=1}^M \beta_{jk}^i B_{jk}(x_T^j)$ and $\{B_{jk}\}$ spline basis functions. Each function f_{ij} allows to model the potential nonlinear regulation from gene j on gene i without any assumptions about the nature of the influence. In this model, the norm of β_j^i controls the influence of gene j on gene i. Morrissey et al. [46] developed a full Bayesian approach to identify the parameters β_j^i of this spline-based model.

Another approach called LOCKNI (Local Kernels for Network Inference) was recently proposed in [47]: it provides an even more direct approach to sparse nonlinear modeling of the expression of a given target gene i in the context of multiple kernel learning. In Eq. (13), each function η^i is now defined for each target gene i, as:

$$\eta_T^i(\mathbf{x}_{T-1};\mathbf{b}_i,\mathbf{w}_i) = \sum_{t=0}^{n-2} w_{it} K_i(\mathbf{x}_{T-1},\mathbf{x}_t), \tag{15}$$

where the components of vector $\mathbf{w}_i$ are weighting the observations at each time point and K_i is positive semi-definite kernel defined by a convex linear combination of the following component (or local) kernels:

$$K_i(\mathbf{x}_{T-1},\mathbf{x}_t) = \sum_{j=1}^p b_{ij}\kappa_j(\mathbf{x}_{T-1},\mathbf{x}_t) \tag{16}$$

with $\kappa_j(\mathbf{x}_{T-1},\mathbf{x}_t) = \kappa(x_{T-1}^j, x_t^j)$, defined as a positive semi-definite kernel applied to the jth projection of both entries $\mathbf{x}_{T-1}$ and $\mathbf{x}_t$. For instance, if we choose a Gaussian kernel, $\kappa_j(\mathbf{x}_{T-1},\mathbf{x}_t))$ writes as:

$$\kappa_j(\mathbf{x}_{T-1},\mathbf{x}_t) = exp(-\gamma(x_{T-1}^j - x_t^j)^2). \tag{17}$$

In the general case, Eq. (15) can be re-written in the following way to emphasize the different roles of $\mathbf{b}_i$ and $\mathbf{w}_i$:

$$\eta_T^i(\mathbf{x}_{T-1};\boldsymbol{\theta}^i) = \sum_{j=1}^p b_{ij} \quad \sum_{t=0}^{n-2} w_{it}\kappa_j(x_{T-1}^j, x_t^j)\Bigg). \tag{18}$$

This model can be seen explicitly as a weighted sum of local or component functions, each of them devoted to a single regulator candidate. The $\mathbf{b}_i$ parameters encode (after sparsification) the presence or absence of regulations, while the $\mathbf{w}_i$ parameters encode the dependence of the model on the training data. Network inference is clearly facilitated by controlling the sparsity of vector $\mathbf{b}_i$. In [47], vectors $\mathbf{w}_i$ and $\mathbf{b}_i$ are estimated for each target gene i using an alternate optimization scheme, typical of multiple kernel learning approaches described in [48,49].

[2] This model can also be presented as a Dynamic Bayesian Network discussed in Section 6 as shown in [45].

5.2. Nonlinear models with operator-valued kernels

Taking another angle, a general framework for network inference based on estimation of the Jacobian matrix of the model was recently introduced in [20]. Model F described in Eq. (1) is chosen to belong to a family of nonparametric nonlinear vector-valued functions $\mathcal{F}$. Once an estimate $\widehat{F}$ of F is obtained from data $(\mathbf{x}_0,\mathbf{x}_1),\dots,(\mathbf{x}_{n-2},\mathbf{x}_{n-1})$ in an autoregression scheme, the Jacobian matrix $\tilde{\nabla}\widehat{F}$ of $\widehat{F}$ is also estimated and thresholded to get an estimate $\widehat{A}$ of the incidence matrix of the directed graph describing the network. More precisely, the Jacobian matrix ∇F of the model F is empirically defined from observations as follows: for a given ordered pair $(i,j) \in \{1,\dots,p\}^2$,

$$\tilde{\nabla}F_{ij} = \frac{1}{n-1}\sum_{t=0}^{n-2} F_{ij}(t)\frac{\partial F(\mathbf{x}_t)_i}{\partial x_t^j}, \tag{19}$$

with $F_{ij}(t) = \frac{\partial F(\mathbf{x}_t)_i}{\partial x_t^j}$.

The Jacobian coefficient $F_{ij}(t)$ represents how much $F(\mathbf{x}_t)_i$ varies when x_t^j varies and therefore, reflects the influence of gene j on gene i. In the linear case, from Eq. (5), the Jacobian matrix corresponds exactly to the regression matrix $A : (\nabla F)_{ij} = a_{ij}$.

In order to cope with nonlinear interactions between genes, [20] proposed a new family of vector kernel-based autoregressive models, called Operator-valued Kernel Vector Auto-Regressive (OKVAR), and given by:

$$\forall T > 1, \quad F(\mathbf{x}_{T-1}) = \sum_{\ell=0}^{n-2} K(\mathbf{x}_{T-1},\mathbf{x}_\ell)\mathbf{c}_\ell \tag{20}$$

with $K(\mathbf{x}_{T-1},\mathbf{x}_\ell) \in \mathcal{L}(\mathbb{R}^p)$, the space of linear operators on p-dimensional vectors, i.e. the space of $p \times p$ real valued matrices and with K satisfying the properties of a matrix-valued kernel. A function K defined as the limit of a matrix-valued kernel $K_r,\ r \in \mathbb{N}$, was proposed for the OKVAR model:

$$K_r(\mathbf{x},\mathbf{y})_{ij} = b_{ij}exp(-\gamma_r||\mathbf{x}-\mathbf{y}||^2)exp(-\gamma(x^i - y^j)^2) \tag{21}$$

with $\lim_{r\to+\infty}\gamma_r = 0$ and B being a positive semi-definite matrix. In this model, a null value for b_{ij} codes for the absence of regulation from gene j to gene i, while the vectors $\mathbf{c}_\ell$ leverage the importance of data $\mathbf{x}_\ell$. Learning F requires to learn both the matrix C whose column vectors are $\mathbf{c}_0,\dots,\mathbf{c}_{n-2}$ and the B matrix, assuming that γ is determined using entropy maximization. Lim et al. [20] introduced a boosting algorithm to learn a linear combination of base models F_m, each of them devoted to a random subspace and learned using an elastic criterion, controlling the sparsity of vectors $\mathbf{c}$ as well as the Jacobian sparsity. The boosted model takes the following form:

$$\forall T > 1, \quad F(\mathbf{x}_{T-1}) = \sum_{m=1}^M \alpha_m F_m(\mathbf{x}_{T-1}) \tag{22}$$

with $\boldsymbol{\alpha}$ a M-dimensional weight vector. At each boosting iteration, the parameters (C_m, B_m) of the base model F_m are estimated through a two-stage procedure: B_m is determined using an independence test and C_m is the minimizer of the following loss function:

$$\mathcal{L}(C_m) = \frac{1}{2(n-1)}\sum_{t=1}^{n-1}||\mathbf{x}_t - F_m(\mathbf{x}_{t-1})||^2 + \lambda_1||C_m||_1 + \lambda_2||F_m||^2_{\mathcal{F}_m}. \tag{23}$$

As in linear models, the ℓ_1 constraint allows to control the sparsity of the model, which is necessary when working in a high dimensional space, with a few time-points. A model selection procedure is then required to select the values of hyperparameters λ_1 and λ_2.

6. Dynamic Bayesian networks

Dynamic Bayesian Networks (DBN) are a generalization of Bayesian networks in order to model random state variables at time t using random state variables at time $t-1$ as described in Eq. (4). Each variable x^j has a set of parent variables $Pa(i)$, on which it exclusively depends. In DBNs, cycles and feedback signals can be represented because dependences always occur from past to present. The network structure is unrolled through time and therefore, it does not need to be acyclic. DBNs can be used to represent either discrete variables or continuous ones. Gaussian DBNs correspond to first order linear autoregressive models, for which each one-dimensional state variable depends only on a subset of variables, its parents, meaning that the corresponding design matrix has true zeros. Nonlinear DBN models for continuous variables have also been derived [45] using spline regression on the parent variables similarly to the model of [46] previously described in Section 5.

DBNs are mostly used with discrete variables; in that case, they are completely identified by the set of parents of each variable and the conditional probability tables relatively to the parents of each variable. Learning methods developed at first for DBNs were inherited from those employed in Bayesian networks: namely, conditional independence-based and scoring methods. Conditional independence-based methods first estimate the dependence graph between variables at time $t-1$ and variables at time t using statistical tests. Once the dependence graph is learned, tables of conditional probabilities relatively to the graph structure can be estimated through maximum likelihood methods. In scoring methods, a function that measures the fit of a model given the data is defined. This scoring function is usually proportional to the logarithm of the *a posteriori* probability of the parameters of the model. Then, an exploration in the space of all possible graphs is undertaken. The latter step is computationally very intensive and evolutionary algorithms for instance have been developed to obtain local solution candidates. Alternative approaches work on a full Bayesian scheme with the use of Markov Chain Monte Carlo sampling to obtain the posterior probability. In recent work, [50] showed that contrary to Bayesian networks, learning a DBN that has only inter-time slice edges is possible in polynomial time. This theoretical result motivated Xuan et al. [51] to propose a global and scalable optimization algorithm. However, a drawback of DBNs as well as Bayesian networks is that they require large sample sizes to be correctly estimated, which is not always the case when it comes to GRNs. For this reason, special network topologies were considered in [52], that led to fast learning algorithms.

Addressing *Causality* in DBN follows the seminal works on causal networks built by Pearl [27,53] about Bayesian Networks. In a Bayesian Network (BN) a directed edge between A and B may be considered as causal if B is affected when A is removed by some intervention. As most of the methods we have explored so far, DBNs do not account for perturbation data, usually called intervention data in the BN and DBN framework. To tackle this issue, Hill et al. recently developed in [54] a modeling framework devoted to DBN that takes into account interventional designs and showed how to extract causal relationships with an illustration on signaling networks.

7. Using stability to improve network inference

Estimation of a high-dimensional network structure is a notoriously difficult task, especially in cases where data are limited. Controlling model sparsity is essential to network inference, but stability of the network learning algorithm is also a highly desirable property.

The learning algorithms presented in Sections 3 and 4 for inferring GRNs are fairly complex and involve a number of tuning parameters. Hence, even if the algorithm is provided with quite similar input data, the estimated network structure may exhibit quite large differences. The concept of stability selection can be used either as a model selection criterion or as a principle of model construction and has attracted a lot of attention recently in the literature [23,25,55] both from a theoretical standpoint and when applied to GRN inference. There exist two mechanisms that can induce changes in the training data: the first is based on *subsampling* (bootstrapping would exhibit a similar behavior) the training samples, while the second adds noise to the available measurements and hence acts as a local perturbation of the input.

7.1. Stability as a criterion for model selection

Meinshausen and Bühlhmann [23] proposed the concept of selection stability in the context of learning high-dimensional structures, such as in regularized regression models or in graphical modeling. It is based on subsampling the input to determine the amount of regularization, so that a certain family-wise type I error in multiple testing of whether a set of variables is part of the model can be conservatively controlled for finite sample size.

The mechanics of the procedure in the context of our network inference problem are as follows:

1. Subsample the input data.
2. Use a learning algorithm to estimate the network structure.
3. Repeat steps (1) and (2) S times.
4. Retain edges in the network that appear in the S replications more often than a prespecified threshold.

When the input data are in the form of time series, care must be taken when subsampling, so that their temporal correlation structure is preserved. For example, [20] used a block-bootstrap procedure that preserves temporal dependencies, together with a non-linear autoregressive model to infer gene regulatory networks. Haury et al. [25] employed these ideas to automatically select features (regulator candidates) in a LARS (Least Angle Regression) model. Randomization of the algorithm is performed by randomly modifying the gene expression matrix. A scoring function assesses the significance of a transcription factor in the model based on the number of times it is selected on the top features by the randomized LARS.

7.2. Stability by construction

As stability is a desirable property of a learning algorithm, several authors have proposed to build stable learning algorithms by randomizing base algorithms and aggregating their outcomes. Randomization can operate on samples via subsampling such as bagging (see for instance, [56]) or on variables via random subspaces, like random forests used in a regression framework by GENIE3 [24], the winner of DREAM 4 and 5 challenges [57].

8. Prior knowledge integration

8.1. Sources of prior knowledge

Considering the unfavorable ratio between the data size and the number of genes, incorporation of prior knowledge into the learning process represents a promising working direction. Prior knowledge used so far to improve performance of network inference algorithms includes general properties of degree distribution in the network, DNA binding sites and motifs, known pathways and

epigenetic information such as histone modification profiles, DNA methylation, interferences by micro-RNA. Several statistical studies [58] tend to show that the degree distribution of transcriptional regulatory networks is scale-free. This assumption can be encoded as prior knowledge during the learning process. Higher level network features have also been discussed in [26]. Prior knowledge can also be defined from additional experimental data like ChIP-Seq experiments [59]. ChIP-Seq experiments dedicated to a transcription factor (TF) allow to identify effective bindings of the given TF on target genes and therefore give a valuable source of knowledge about candidate target genes. When such experimental data are not available, it is still possible to find a description of transcription factors with their known DNA binding sites in databases like TRANSFAC [60] devoted to eukaryote organisms. A target gene can be associated to a transcription factor as soon as one of the transcription factor DNA binding sites can be found in the proximal promoter region of the target. DNA sequences can be obtained from the Ensembl project and processed by off-the-shelf tools like RSTAT as explained in [61]. Text-mining tools such as Pathway Studio or Ingenuity also provide an important source of knowledge about regulations associated with the bibliographic references these regulations have been introduced. Concerning epigenetic features, the work of Zheng et al. [62] assumes that genes involved in the same regulatory pathways have similar patterns of epigenetic features. Their study reveals that histone modification profiles of both regulators and regulees are correlated and that the corresponding correlation matrix can be used as a prior regarding the absence or presence of edges in the network. Furthermore, perturbation data like systematic single gene knock-out data also serve as prior knowledge [63] for network inference from time-series. Finally, all theses sources of information can provide prior knowledge under the form of an adjacency matrix corresponding to an initial guess of the network structure.

8.2. Methods to incorporate prior knowledge

Different methods have been developed to integrate such knowledge. Research in this area covers frequentist regularized regression [64,61], as well as Bayesian approaches [65,66,26,67]. Interestingly, all these works emphasize the importance of finding a good balance on weighing the experimental data and prior knowledge [64,61]. Within the framework of regularized regression, a recent approach described in [64] expresses prior knowlegde on regulations as a modifier on each ℓ_1 constraint applied to a single weight. This way, the ℓ_1 constraint is relaxed when applied to a putative regulation. Another closely related approach has also been proposed in Weber et al. [61] in the context of a system of linear ordinary differential equations. Regulations that are supposed to be known are associated with prior parameters. Parameters of the linear ODEs are learnt by minimizing the square loss penalized by the weighted distance between them and the prior parameters. The weights associated with each distance are defined as score values for the prior knowledge. If a regulation is known with great certainty, then the corresponding score is high and the prior knowledge becomes a model requirement. The majority of other approaches have been developed in the context of Bayesian estimation [65,66,26]. For DBNs, the works described in [65], as well as in [66,26], define a prior distribution on network structures (matrices) G as a Gibbs distribution:

$$P(G|\beta) = \frac{e^{-\beta E(G)}}{Z(\beta)} \tag{24}$$

with β a hyperparameter, $E(G)$ an energy function that can take different forms according to the nature of the prior, and $Z(\beta) = \sum_G e^{-\beta E(G)}$ the normalizing term. Denoting by B the matrix that codes for prior knowledge coming from one of the sources of information cited previously, Werhli et al. [66] express the energy function E as the ℓ_1 norm of the difference between G and B. Taking the logarithm of the prior distribution in Eq. (24) provides an expression similar to the one proposed by Weber et al., except that there is a single weight β for the whole term. Models are then learnt either by a maximum posterior approach [65] or by sampling the corresponding posterior distribution using Markov Chain Monte Carlo (MCMC) as in [66,26]. When several sources of information provide different initial guesses $B_i, i = 1, \ldots I$, [66] shows that the resulting energy function can be efficiently computed using the modularity of (Dynamic) Bayesian Networks. Muk et al. [26] also propose a list of more sophisticated priors for Bayesian Networks that also work for DBNs. Another approach involves hierarchical priors: in a Boolean Dynamic Bayesian network based on a sigmoid function composed with a linear model, [67] uses a hierarchical prior on the weight matrix. Specifically, at the first level, a Gaussian prior is employed on the parameters where all the prior distributions over weights emanating from the same gene j share the same variance σ_j^2. At the second level, σ_j^2 has a Gamma distribution. The use of this specific hierarchical prior has the following effects: genes that receive large weights exhibit also large variance and hence are good candidates for acting as hubs.

In [64], a method based on Bayesian regression with a modification of Zellner's g prior is developed for modeling the network behavior dimension by dimension. The prior on the regression coefficients takes the form of a multivariate Gaussian centered at an initial guess and the empirical covariance matrix re-scaled by a chosen factor g as covariance matrix. The choice of g depends on the belief about the initial guess.

9. Performance assessment of the network inference algorithms

Ideally, performance of network inference algorithms should be directly proportional to their real impact on biological discoveries. However it is not always possible to get an *in silico* prediction validated by biologists, especially in case of relatively large networks. In order to be able to compare several algorithms, a consortium of researchers initiated yearly challenges for computational biology tasks and especially network inference. The so-called DREAM project (Dialogue for Reverse Engineering Assessments and Methods) has as its main objective to enable and strengthen interactions between experiments and theory in the area of cellular network inference and quantitative model building in systems biology.

Towards that goal, it has held annual competitions from the year 2006 onwards. A number of these events included a gene regulatory network inference challenge: DREAM 3 [68], DREAM 4 [69] and DREAM 5 [57].

Next, we describe some of the features of the network inference DREAM 3 challenge. Subnetworks of the accepted *E. coli* and *S. cerevisiae* gene regulatory networks were extracted and gene expression measurements were generated using a thermodynamic model of coupled differential equations. A small amount of Gaussian noise was added to the generated trajectories to simulate measurement error. The obtained data reflected three types of experiments: (1) time course data of a wild type strain following an environmental perturbation (i.e., trajectories); (2) knock-down of a gene by deletion of one copy in a diploid organism (i.e., heterozygous mutants); (3) knock-out of a gene by deletion of both copies in a diploid organism (i.e., homozygous null mutants). The size of the challenge networks is 10, 50 and 100 and for each of them participants were asked to submit reconstructions for five variants, Ecoli1, Ecoli2, Yeast1, Yeast2, Yeast3, whose topology varied in terms of overall density and number of regulators for each gene.

The objective was to infer the underlying directed network, but not the sign of the edge (up- or down-regulated). Performance metrics included true and false positive and negative rates, summarized in Receiver Operator Curves and Precision-Recall ones.

Several teams submitted predictions for the 10, 50 and 100-size network challenges, and an overall assessment indicates that no single algorithm dominated all the challenges [68]. The submitted algorithms covered a wide spectrum of techniques, ranging from clustering ones, to DBNs, to employing ordinary differential equation modeling. The winning algorithms employed both time course data and perturbation data. However, in our recent work [20] employing nonlinear operator-valued kernel autoregressive models, we managed to match the performance of the best algorithms relying *only* on time course data. The take-home message is that sophisticated algorithms properly tuned can perform exceptionally well, even when relying only on time course data, which, as pointed out in the introductory section, are more readily available than data from exhaustive knock-out/down experiments.

The DREAM 4 challenge also focused on network inference, but for protein signaling networks from phosphoproteomics data. Although this learning task is similar in nature to that for GRNs and the competing teams employed similar algorithms as those discussed above, the main focus of the challenge was on how well predicted measurements from reconstructed networks matched the actual phosphoproteomics ones.

The DREAM 5 challenge revisited the problem of learning a GRN from gene expression data. The networks under consideration were from a prokaryotic model organism *(E. coli)* as in the DREAM 3 challenge, a eukaryotic model organism *(S. cerevisiae)*, a human pathogen *(S. aureus)* and an *in silico*benchmark. Similar type of data as in the DREAM 3 challenge were furnished to the competing teams that in turn employed a wide range of learning algorithms, like in the DREAM 3 challenge, including DBNs, sparse regression models, random forests based algorithms, as well as clustering ones.

The take-home message from this challenge is that no algorithm performed consistently across all networks, probably due to the differing features of their underlying topologies. It is interesting to note though that a consensus type strategy discussed in [57], where a network is built by integrating the inferred networks from different algorithms, clearly outperformed stand alone methods. This is akin to employing stability selection across a large model space, thus confirming the importance of this principle for GRN reconstruction.

10. Discussion

Learning methods devoted to GRN inference have now reached a certain degree of maturity, since most of them integrate some of the key ingredients of statistical learning, such as sparsity and stability. For a biomedical researcher contemplating which model to use for this learning task, our broad recommendation is to start with linear or kernel based vector autoregressive models, depending on whether or not the time course data exhibit strong nonlinear dynamics. The reason is that these two model classes share a number of similarities both at the conceptual, as well as the technical level (both boil down to fitting penalized regressions). Further, fast computational algorithms have been developed to train them and they can produce sparse and stable networks. Hence, they can be used effectively to learn the structure of GRNs. On the other hand, although conceptually powerful, DBNs are computationally expensive and therefore do not scale well for learning large scale GRNs. However, one can adopt a hybrid strategy, by first using (non)-linear autoregressive models to obtain an estimate of the GRN and then use it as a prior for a DBN model to further explore the network structure.

However, this first generation of tools needs to face new challenges. Let us focus on the following important issues that are routinely encountered by biologists: multiple data integration, knowledge integration, scaling to large networks and experimental design. Integration of multiple sources of data together with multiple methods has been recently addressed in [70] by formulating the learning problem as a multi-objective optimization task within an ensemble method framework. Combining multiple sources of data with prior knowledge is however still a challenge: while kernels have already shown to provide a powerful framework to data integration [71], other approaches combining logic-based approaches with probabilistic graphical models [13] may offer an interesting interface between biologists and modelers. In general, scaling algorithms to higher dimensional spaces often assumes a *modular* structure in the data. Interestingly, this fits well with observations about the apparent design features of the GRN, for instance, in development [72]. Principled modular learning approaches [73] based on mixture modeling, although not very well developed to date, may hold the key to large scale network inference. Finally, present practice focuses on network reconstruction *after* the data have been collected. Experimental design strategies that inform the biologists which are the most informative perturbation experiments for network identification purposes [74] will play a key role to improve GRN reconstruction.

Acknowledgments

GM was supported by NSF DMS-1228164 and NSA H98230-13-1-024 and FAB by the French National Research Agency [ANR-09-SYSC-009-01].

References

[1] D. Licatalosi, R.B. Darnell, Rna processing and its regulation: global insights into biological networks, Nature Reviews Genetics (2010) 75.
[2] R. Aebersold, M. Mann, Mass spectrometry-based proteomics, Nature (2003) 198.
[3] K. Dettmer, P. Aronov, B. Hammock, Mass spectrometry-based metabolomics, Mass Spectrom Reviews 26 (2007) 51.
[4] G. Michailidis, Statistical challenges in biological networks, Journal of Computational and Graphical Statistics 21 (2012) 840.
[5] E. Voit, Modelling metabolic networks using power-laws and s-systems, Essays Biochemistry (2008) 29.
[6] C. Brouard, F. d'Alché Buc, M. Szafranski, Semi-supervised penalized output kernel regression for link prediction, in: L. Getoor, T. Scheffer (Eds.), ICML, Omnipress, 2011, p. 593.
[7] Q.C. Zhang, D. Petrey, L. Deng, L. Qiang, Y. Shi, C. Thu, B. Bisikirska, C. Lefebvre, D. Accili, T. Hunter, et al., Structure-based prediction of protein-protein interactions on a genome-wide scale, Nature 490 (2012) 556.
[8] C. Sima, J. Hua, S. Jung, Inference of gene regulatory networks using time-series data: a survey, Current Genomics (2009) 416.
[9] I.C. Chou, E.O. Voit, Recent developments in parameter estimation and structure identification of biochemical and genomic systems, Mathematical Biosciences 219 (2009) 57.
[10] N. Lawrence, M. Girolami, M. Rattray, G. Sanguinetti, Learning and Inference in Computational Systems Biology, MIT Press, 2010.
[11] C. Auliac, V. Frouin, X. Gidrol, F. d'Alché Buc, Evolutionary approaches for the reverse-engineering of gene regulatory networks: a study on a biologically realistic dataset, BMC Bioinformatics 9 (2008) 91.
[12] F. Mordelet, J.-P. Vert, Sirene: supervised inference of regulatory networks, Bioinformatics 24 (2008) i76.
[13] C. Brouard, J. Dubois, C. Vrain, D. Castel, M.-A. Debily, F. d'Alché Buc, Learning a markov logic network for supervised inference of a gene regulatory network: application to the id2 regulatory network in human keratinocytes, BMC Bioinformatics, to appear, 2013.
[14] B.-E. Perrin, L. Ralaivola, A. Mazurie, S. Bottani, J. Mallet, F. d'Alché Buc, Gene networks inference using dynamic bayesian networks, Bioinformatics 19 (2003) 38.
[15] A. Hartemink, Reverse engineering gene regulatory networks, Nature Biotechnology 23 (2005) 554.
[16] M. Bansal, G. Della Gatta, D. di Bernardo, Inference of gene regulatory networks and compound mode of action from time course gene expression profiles, Bioinformatics 22 (2006) 815.

[17] A. Fujita, J. Sato, H. Garay-Malpartida, R. Yamaguchi, S. Miyano, M. Sogayar, C.E. Ferreira, Modeling gene expression regulatory networks with the sparse vector autoregressive model, BMC Systems Biology 1 (2007), Article 39.

[18] A. Shojaie, G. Michailidis, Penalized likelihood methods for estimation of sparse high dimensional directed acyclic graphs, Biometrika 97 (3) (2010) 519.

[19] S. Basu, A. Shojaie, G. Michailidis, Network granger causality with inherent grouping structure, 2012, 1. ArXiv:1210.3711v3.

[20] N. Lim, Y. Senbabaoglu, G. Michailidis, F. d'Alché Buc, Okvar-boost: a novel boosting algorithm to infer nonlinear dynamics and interactions in gene regulatory networks, Bioinformatics 29 (2013) 1416.

[21] A.A. Margolin, I. Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, R. Favera, A. Califano, Aracne: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context, BMC Bioinformatics 7 (2006) S7.

[22] P. Zoppoli, S. Morganella, M. Ceccarelli, Timedelay-aracne: reverse engineering of gene networks from time-course data by an information theoretic approach, BMC Bioinformatics 11 (2010) 154.

[23] N. Meinshausen, P. Bühlmann, Stability selection (with discussion), Journal of the Royal Statistical Society:Series B (2010) 417.

[24] V.A. Huynh-Thu, A. Irrthum, L. Wehenkel, P. Geurts, Inferring regulatory networks from expression data using tree-based methods, PLos ONE 5 (2010) e12776.

[25] A.-C. Haury, F. Mordelet, P. Vera-Licona, J.-P. Vert, Tigress: Trustful inference of gene regulation using stability selection, BMC Systems Biology 6 (2012), Article 145.

[26] S. Mukherjee, T. Speed, Network inference using informative priors, Proceedings of the National Academy of Sciences 105 (2008) 14313.

[27] J. Pearl, Causality: Models, Reasoning, and Inference, Cambridge Univ Press, 2000.

[28] T.S. Gardner, D. di Bernardo, D. Lorenz, J.J. Collins, Inferring genetic networks and identifying compound mode of action via expression profiling, Science 301 (2003) 102.

[29] J. Tegner, M.K. Yeung, J. Hasty, J.J. Collins, Reverse engineering gene networks: integrating genetic perturbations with dynamical modeling, Proceedings of the National Academy of Sciences USA 100 (2003) 5944.

[30] C.W.J. Granger, Investigating causal relations by econometric models and cross-spectral methods, Econometrica 37 (1969) 424.

[31] G. Karlebach, R. Shamir, Modelling and analysis of gene regulatory networks, Nature Reviews Molecular Cell Biology 9 (2008) 770.

[32] G. Batt, D. Ropers, H.D. Jong, J. Geiselmann, R. Mateescu, D. Schneider, Validation of qualitative models of genetic regulatory networks by model checking: analysis of the nutritional stress response in escherichia coli, Bioinformatics 21 (2005) 19.

[33] E. Voit, M. Savageau, Power-law approach to modeling biological systems; iii. methods of analysis, Journal of Fermentation Technology 60 (1982) 233.

[34] E. Voit, Computational Analysis of Biochemical Systems: A Practical Guide for Biochemists and Molecular Biologists, Cambridge University Press, Cambridge; New York, 2000.

[35] M. Vilela, I.-C. Chou, S. Vinga, A. Vasconcelos, E. Voit, J. Almeida, Parameter optimization in s-system models, BMC Systems Biology 2 (2008) 35.

[36] A. Shojaie, G. Michailidis, Discovering graphical granger causality using a truncating lasso penalty, Bioinformatics 26 (18) (2010) i517.

[37] A.C. Lozano, N. Abe, Y. Liu, S. Rosset, Grouped graphical granger modeling for gene expression regulatory networks discovery, Bioinformatics 25 (2009) i110.

[38] H. Lütkepohl, New Introduction to Multiple Time Series Analysis, Springer, 2005.

[39] N. Mukhopadhyay, S. Chatterjee, Causality and pathway search in microarray time series experiment, Bioinformatics 23 (2007) 442.

[40] Y. Benjamini, Y. Hochberg, Controlling the false discovery rate: a practical and powerful approach to multiple testing, Journal of the Royal Statistical Society Series B Methodological 57 (1995) 289.

[41] A.-L. Barabasi, R. Albert, Emergence of scaling in random networks, Science 286 (1999) 11.

[42] T. Shimamura, S. Imoto, R. Yamaguchi, A. Fujita, M. Nagasaki, S. Miyano, Recursive regularization for inferring gene networks from time-course gene expression profiles, BMC Systems Biology, 2009.

[43] R. Opgen-Rhein, K. Strimmer, Learning causal networks from systems biology time course data: an effective model selection procedure for the vector autoregressive process, BMC Bioinformatics 8 (2007) S3.

[44] P. Buhlmann, S. van de Geer, Statistics for High-Dimensional Data: Methods, Theory and Applications, Springer, 2011.

[45] S. Kim, S. Imoto, S. Miyano, Dynamic bayesian network and nonparametric regression for nonlinear modeling of gene networks from time series gene expression data, Biosystems 75 (2004) 57.

[46] E.R. Morrissey, M.A. Jurez, K.J. Denby, N.J. Burroughs, Inferring the time-invariant topology of a nonlinear sparse gene regulatory network using fully bayesian spline autoregression, Biostatistics 12 (2011) 682.

[47] A. Fouchet, J.-M. Delosme, F. d'Alché Buc, Gene regulatory network inference using ensemble of multiple local kernel models, Programme of Seventh International Workshop on Machine Learning in Systems Biology, satellite meeting of ISMB'2013, Uwe Owler and Jean-Philippe Vert, July 19–20, 2013.

[48] M. Gonen, E. Alpaydyn, Multiple kernel learning algorithms, JMLR 12 (2011) 2211.

[49] A. Rakotomamonjy, F. Bach, S. Canu, Y. Grandvalet, Simplemkl, Journal of Machine Learning Research 9 (2008) 2491.

[50] N. Dojer, Learning bayesian networks does not have to be np-hard, Proceedings of International Symposium on Mathematical Foundations of Computer Science (2006) 305.

[51] N. Xuan, M. Chetty, R. Coppel, P. Wangikar, Gene regulatory network modeling via global optimization of high-order dynamic bayesian network, BMC Bioinformatics 13 (2012) 131.

[52] A. Tresch, F. Markowetz, Structure learning in nested effects models, Statistical Applications in Genetics and Molecular Biology 7 (2008) 9.

[53] D. Eaton, K.P. Murphy, Exact bayesian structure learning from uncertain interventions, Journal of Machine Learning Research – Proceedings Track 2 (2007) 107.

[54] S. Spencer, S. Hill, S. Mukherjee, Dynamic Bayesian networks for interventional data, Technical Report, Warwick University, UK, 2012.

[55] J.C. Rajapakse, P.A. Mundra, Stability of building gene regulatory networks with sparse autoregressive models, BMC Bioinformatics 12 (2011) S17.

[56] R. de Matos Simoes, F. Emmert-Streib, Bagging statistical network inference from large-scale gene expression data, PLoS One 7 (2012) e33624.

[57] D. Marbach, J.C. Costello, R. Kuffner, N.M. Vega, R.J. Prill, D.M. Camacho, K.R. Allison, T.D. Consortium, M. Kellis, J.J. Collins, et al., Wisdom of crowds for robust gene network inference, Nature Methods 9 (2012) 796.

[58] R. Albert, Scale-free networks in cell biology, Journal of Cell Science 118 (2005) 4947.

[59] A. Valouev, D. Johnson, A. Sundquist, C. Medina, E. Anton, S. Batzoglou, R. Myers, A. Sidow, Genome-wide analysis of transcription factor binding sites based on chip-seq data, Nature Methods 5 (2008) 829.

[60] E. Wingender, The transfac project as an example of framework technology that supports the analysis of genomic regulation, Briefings in Bioinformatics 9 (2008) 326.

[61] M. Weber, S. Henkel, S. Vlaic, R. Guthke, E. van Zoelen, D. Driesch, Inference of dynamical gene-regulatory networks based on time-resolved multi-stimuli multi-experiment data applying netgenerator v2.0, BMC Systems Biology 7 (2013) 1.

[62] J. Zheng, I. Chaturvedi, J.C. Rajapakse, Integration of epigenetic data in bayesian network modeling of gene regulatory network, in: M. Loog, L.F.A. Wessels, M.J.T. Reinders, D. de Ridder (Eds.), PRIB, Lecture Notes in Computer Science, 7036, Springer, 2011, p. 87.

[63] A. Pinna, N. Soranzo, A. de la Fuente, From knockouts to networks: establishing direct cause-effect relationships through graph analysis, PLoS ONE 5 (2010) e12912.

[64] A. Greenfield, C. Hafemeister, R. Bonneau, Robust data-driven incorporation of prior knowledge into the inference of dynamic regulatory networks, Bioinformatics 29 (2013) 1060.

[65] S. Imoto, T. Higuchi, T. Goto, K. Tashiro, S. Kuhara, S. Miyano, Combining microarrays and biological knowledge for estimating gene networks via bayesian networks, in: Proceedings of the IEEE Computer Society Bioinformatics Conference (CSB 03), IEEE, 2003, p. 104.

[66] A. Werhli, D. Husmeier, Reconstructing gene regulatory networks with bayesian networks by combining expression data with multiple sources of prior knowledge, Statistical Applications in Genetics and Molecular Biology 6 (2007), Article 15.

[67] M. Bock, S. Ogishima, H. Tanaka, S. Kramer, L. Kaderali, Hub-centered gene network reconstruction using automatic relevance determination, PLoS ONE 7 (2012) e35077.

[68] D. Marbach, R.J. Prill, T. Schaffter, C. Mattiussi, D. Floreano, G. Stolovitzky, Revealing strengths and weaknesses of methods for gene network inference, Proceedings of the National Academy of Sciences USA 107 (14) (2010) 6286.

[69] R.J. Prill, J. Saez-Rodriguez, L.G. Alexopoulos, P.K. Sorger, G. Stolovitzky, Crowdsourcing network inference: the dream predictive signaling network challenge, Science Signaling 4 (2011) mr7.

[70] R. Gupta, A. Stincone, P. Antczak, S. Durant, R. Bicknell, A. Bikfalvi, F. Falciani, A computational framework for gene regulatory network inference that combines multiple methods and datasets, BMC Systems Biology 5 (2011) 52.

[71] B. Scholkopf, T. Tsuda, J.-P. Vert, Kernel Methods In Computational Biology, The MIT press, 2004.

[72] E.H. Davidson, M.S. Levine, Properties of developmental gene regulatory networks, Proceedings of the National Academy of Sciences 105 (2008) 20063.

[73] E. Segal, M. Shapira, A. Regev, D. Pe'er, D. Botstein, D. Koller, N. Friedman, Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data, Nature Genetics 34 (2) (2003) 166.

[74] B. Steiert, A. Raue, J. Timmer, C. Kreutz, Experimental design for parameter estimation of gene regulatory networks, PLoS ONE 7 (2012) e40052.