BIOINFORMATICS ORIGINAL PAPER Vol. 25 no. 22 2009, pages 2937–2944
doi:10.1093/bioinformatics/btp511

Systems biology

Learning gene regulatory networks from gene expression measurements using non-parametric molecular kinetics

Tarmo Äijö[1] and Harri Lähdesmäki[1,2,*]

[1]Department of Signal Processing, Tampere University of Technology, Tampere and
[2]Department of Information and Computer Science, Helsinki University of Technology, Helsinki, Finland

Received on April 21, 2009; revised on July 30, 2009; accepted on August 17, 2009

Advance Access publication August 25, 2009

Associate Editor:Olga Troyanskaya

ABSTRACT

Motivation: Regulation of gene expression is fundamental to the operation of a cell. Revealing the structure and dynamics of a gene regulatory network (GRN) is of great interest and represents a considerably challenging computational problem. The GRN estimation problem is complicated by the fact that the number of gene expression measurements is typically extremely small when compared with the dimension of the biological system. Further, because the gene regulation process is intrinsically complex, commonly used parametric models can provide too simple description of the underlying phenomena and, thus, can be unreliable. In this article, we propose a novel methodology for the inference of GRNs from time-series and steady-state gene expression measurements. The presented framework is based on the use of Bayesian analysis with ordinary differential equations (ODEs) and non-parametric Gaussian process modeling for the transcriptional-level regulation.

Results: The performance of the proposed structure inference method is evaluated using a recently published *in vivo* dataset. By comparing the obtained results with those of existing ODE- and Bayesian-based inference methods we demonstrate that the proposed method provides more accurate network structure learning. The predictive capabilities of the method are examined by splitting the dataset into a training set and a test set and by predicting the test set based on the training set.

Availability: A MATLAB implementation of the method will be available from http://www.cs.tut.fi/~aijo2/gp upon publication.

Contact: harri.lahdesmaki@tut.fi

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

An important problem in molecular biology is to understand the regulatory mechanisms that control gene expression. Although high-throughput technologies have recently witnessed tremendous progress, complex molecular control mechanisms cannot be deciphered using experimental methods alone. Hence, computational modeling has an important role in revealing genome-wide regulatory mechanisms.

Models of transcriptional regulation are commonly depicted in the form of a network (or graph), where directed connections between nodes represent the regulatory interactions. Genome-wide transcriptional regulatory mechanisms are largely unknown and, thus, a central goal is to infer the structure of the gene regulatory network (GRN) from the experimental data. Biological samples are commonly profiled using gene expression microarrays and the measured mRNA levels provide a valuable and quantitative information source for understanding molecular control mechanisms at systems level. Inference of GRNs from experimental data, such as gene expression measurements, represents a considerably challenging problem.

In recent years, researchers have proposed several different computational approaches to reconstruct GRNs, e.g. see reviews by Bansal *et al.* (2007) and Markowetz and Spang (2007). These include, among others, approaches that rely on linear models (D'haeseleer *et al.*, 1999), information theory (ARACNE) (Margolin *et al.*, 2006), static and dynamic Bayesian networks (BANJO; Yu *et al.*, 2004) and Boolean networks and their probabilistic extensions (Shmulevich *et al.*, 2002). While these methods have been found useful in a number of applications, they primarily model the data, not the underlying biological process. On the other hand, GRNs could be modeled in great detail with chemical reaction network models. However, there are major difficulties in inference with this modeling approach, e.g. lack of measurements from single cells and computational problems in inferring the model parameters and structure from data (Wilkinson, 2006). The exact models are commonly approximated by ordinary differential equations (ODE), which can be obtained as the expectation of the chemical master equation under certain assumptions, and are often coupled with linear, mass action, sigmoidal, Hill or Michaelis–Menten kinetics. A number of different modeling approaches using ODEs have been proposed, including, among others, estimation of model parameters (Cao and Zhao, 2008), inference for unknown transcription factor (TF) levels (Gao *et al.*, 2008), coupling ODE models with protein complexes (Wang *et al.*, 2007) and model structure inference (NIR, TSNI and Inferelator; Bansal *et al.*, 2006; Bonneau *et al.*, 2006; Gardner *et al.*, 2003). Other related methods that combine aspects from ODEs and Bayesian modeling have been proposed, e.g. in Imoto *et al.* (2002), Perrin *et al.* (2003), Nachman *et al.* (2004) and Zou and Conzen (2005).

Overall, the gene expression process consists of several steps, including transcription, splicing and translation. The transcription

*To whom correspondence should be addressed.

step has a central role in controlling gene expression and, hence, most of the network modeling approaches focus on that level of regulation. Different kinetic models used to capture transcriptional dynamics in ODEs are well motivated and widely used, but they are derived based on simplified assumptions. For example, it is known that the transcription alone consists of at least four main steps: TF binding, initiation, elongation and termination (Greive and von Hippel, 2005). In addition to these difficulties, in the case of more than one TF there remains a question about protein dimerizations and their cooperative effect on regulation, such as AND, OR or XOR types of logic. Consequently, commonly applied parametric kinetic models may be unreliable and may have too simple view on the underlying phenomena.

Additional difficulty stems from the fact that it is currently difficult to measure protein concentrations in a high-throughput manner. Although methods exist for estimating unknown protein concentrations for a given GRN structure, such an approach leads to an increased computational complexity. Furthermore, no computational method has been proposed so far to simultaneously estimate protein levels and GRNs that includes combinatorial regulatory interactions. Thus, the inference is usually done based on mRNA measurements. In addition, usually there are only a small number of gene expression measurements available and the measurements are contaminated by noise.

Here, we propose an approach that is similar to the aforementioned ODE-based methods but has a couple of important differences. The main differences are non-parametric modeling of molecular kinetics and Bayesian analysis. Given the complexity of the transcription step and the overall gene expression process, non-parametric modeling allows us to infer the shape of a regulatory function from the data without making any drastic assumptions beforehand. Bayesian approach is particularly well-suited for 'small n large p' problems where the measurements additionally contain a considerable amount of uncertainly. In particular, under the Bayesian approach, we may assign probabilities to different models, whereas in the traditional frequentist view, we are only able to accept or reject different models, which might be too harsh an action given the limitations discussed above. Additionally, uncertainty in measurements is taken into consideration in the model by assuming normally distributed noise and learning its characteristics from measurements. The proposed reverse engineering method is also able to use both steady-state and time-series data. Having the aforementioned aspects (non-parametric, ODE and Bayesian modeling) in mind, we discriminate our approach from previously proposed GRN structure learning methods in the following fashion. All ODE-based methods are essentially parametric, such as those proposed in Gardner *et al.* (2003), Perrin *et al.* (2003), Nachman *et al.* (2004), Bansal *et al.* (2006) and Bonneau *et al.* (2006). The work of Gao *et al.* (2008), however, shows a departure from standard parametric approaches in that latent protein activities are modeled using Gaussian processes, although the regulation function has a parametric form. Previously proposed non-parametric approaches, on the other hand, are essentially not based on differential equation-type modeling, such as those in Imoto *et al.* (2002), Yu *et al.* (2004) and Zou and Conzen (2005). Finally, most of the ODE-based approaches make use of frequentist inference (Bansal *et al.*, 2006; Bonneau *et al.*, 2006; Gardner *et al.*, 2003), which as such might have, e.g. the aforementioned problems of making hard decisions (although resampling methods can alleviate that problem).

Performance of the proposed computational method is assessed using a recently published *in vivo* reverse-engineering and modeling assessment (IRMA) network (Cantone *et al.*, 2009), which provides an excellent framework for validation. Results demonstrate that our novel modeling approach provides more accurate network structure predictions than other commonly used ODE and Bayesian methods and non-parametric modeling allows us to identify molecular kinetics that best explain experimental data. In the next section, we will introduce our modeling framework. After that, the performance of the presented method is assessed by comparing it with other methods using real data. Finally, we conclude this study in Section 4.

2 METHODS

2.1 Gene regulation model

We base our modeling approach on the commonly used first-order ODE model which, given the lack of protein concentration measurements, uses amounts of mRNA as a proxy for protein concentrations. Let $x_i(t)$ denote the expression of gene i at time t and vector $\hat{\mathbf{x}}_i(t)$ denote the expressions of genes that regulate gene i. The general ODE model can be expressed as (Barenco *et al.*, 2006)

$$\frac{\mathrm{d}x_i(t)}{\mathrm{d}t} = \alpha_i + f_i(\hat{\mathbf{x}}_i(t)) - \lambda_i x_i(t), \tag{1}$$

where α_i is the basal transcription rate, f_i is an unknown regulation function and λ_i is the decay rate of the mRNA. We also consider the possibility that a gene x_i is not regulated by other genes via regulatory function f_i. In that case, the model in Equation (1) reduces to the following form

$$\frac{\mathrm{d}x_i(t)}{\mathrm{d}t} = \alpha_i - \lambda_i x_i(t). \tag{2}$$

Since we only have measurements of gene expressions, we approximate the rates of gene expression with a first-order approximation

$$\frac{\mathrm{d}x_i(t_k)}{\mathrm{d}t} \simeq \Delta x_i(t_k) = \frac{x_i(t_{k+1}) - x_i(t_k)}{t_{k+1} - t_k} \tag{3}$$

for a given set of measurement time points. If one wants to infer regulatory interactions from steady-state measurements, then the rate of expression is set to zero

$$\frac{\mathrm{d}x_i(t)}{\mathrm{d}t} \simeq \Delta x_i(t) = 0. \tag{4}$$

One of the key ideas behind our method is to use Gaussian processes to learn the unknown regulation function $f_i(\cdot)$ from the data.

2.2 Gaussian processes

Gaussian processes provide non-parametric prior distributions over functions and can be thought of as a generalization of multinomial Gaussian distributions. A Gaussian process is defined to be a collection of random variables such that any finite subset of the random variables have a joint Gaussian distribution. We use the following notation to represent that values of a function $f(\mathbf{x})$ are modeled by a Gaussian process (Rasmussen and Williams, 2005)

$$f(\mathbf{x}) \sim \mathcal{GP}\big(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')\big), \tag{5}$$

where $m(\mathbf{x})$ is a mean function and $k(\mathbf{x}, \mathbf{x}')$ is a covariance function. From now on, this section, we assume that the mean function is identically zero for notational convenience.

The covariance matrix is constructed by using covariance functions, i.e. $K_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)$. In order to control mean square differentiability of the process, we choose to use the Matérn covariance function with $\nu = 3/2$ for both time-series and steady-state data

$$k(\mathbf{x}, \mathbf{x}') = \sigma(1 + \sqrt{3}\sqrt{\mathbf{u}^{\mathrm{T}} P^{-1} \mathbf{u}}) \exp\left(-\sqrt{3}\sqrt{\mathbf{u}^{\mathrm{T}} P^{-1} \mathbf{u}}\right),$$

where $\mathbf{u} = \mathbf{x} - \mathbf{x}'$, $P = \mathrm{diag}(l^2)$, l is a length-scale hyperparameter and σ is an additional-scale hyperparameter.

By assuming normal i.i.d. additive noise on measurements, we may write the combined covariance function as $k_c(\mathbf{x}_i,\mathbf{x}_j)=k(\mathbf{x}_i,\mathbf{x}_j)+\sigma_n^2\delta_{ij}$, where δ_{ij} is the Kronecker delta and σ_n^2 is a hyperparameter additional to the hyperparameters of the Matérn covariance function. In that case, we may write the joint distribution of the training samples $(\mathbf{y},X)$ and test samples $(\mathbf{f}_*,X_*)$ as

$$\begin{bmatrix}\mathbf{y}\\ \mathbf{f}_*\end{bmatrix}\sim\mathcal{N}\left(0,\begin{bmatrix}K(X,X)+\sigma_n^2 I & K(X,X_*)\\ K(X_*,X) & K(X_*,X_*)\end{bmatrix}\right), \tag{6}$$

where matrix X contains the explanatory variables $\mathbf{x}$ as columns and vector $\mathbf{y}$ contains the responses (similarly for $\mathbf{f}_*$ and X_*).

Predictions by a Gaussian process are done with the mean in the following way

$$\mathbf{f}_*|\mathbf{y},X,X_*\sim\mathcal{N}(\bar{\mathbf{f}}_*,\mathrm{Cov}(\mathbf{f}_*)),\text{ where} \tag{7a}$$

$$\bar{\mathbf{f}}_*=K_*^{\mathrm{T}}K_n^{-1}\mathbf{y}, \tag{7b}$$

$$\mathrm{Cov}(\mathbf{f}_*)=K(X_*,X_*)-K_*^{\mathrm{T}}K_n^{-1}K_*, \tag{7c}$$

where $K_n=K(X,X)+\sigma_n^2 I$ and $K_*=K(X,X_*)$. As usual, extrapolation results should be used carefully, however, interpolation results are usually more reliable as we will see. Given the above specification for a Gaussian processes, an important result is that the marginal likelihood $p(\mathbf{y}|X,\theta)$ can be computed analytically (Rasmussen and Williams, 2005)

$$p(\mathbf{y}|X,\theta)=(2\pi)^{-n/2}|K_n|^{-1/2}\exp\left(-\frac{1}{2}\mathbf{y}^{\mathrm{T}}K_n^{-1}\mathbf{y}\right), \tag{8}$$

where θ denotes the hyperparameters.

2.3 Non-parametric molecular kinetics

The unknown regulation function f_i in Equation (1) is modeled with a zero-mean Gaussian process as it is difficult to justify any specific mean function for f_i, which, in general, can be activating or repressing or often implements a more complex combinatorial function. Since in practice the exact values of basal rate α_i and decay rate λ_i are also unknown, we assign prior probability distributions to them and integrate them out to get the complete marginal likelihood. Assignment of a prior can be done easily in the Gaussian process framework by introducing to the model a set of fixed basis functions $\mathbf{h}(\mathbf{x})$

$$g(\mathbf{x})=f(\mathbf{x})+\mathbf{h}(\mathbf{x})^{\mathrm{T}}\beta, \tag{9}$$

where $f(\mathbf{x})\sim\mathcal{GP}\left(0,k(\mathbf{x},\mathbf{x}')\right)$ and β is the coefficient vector of the linear regression model.

For the gene regulation model shown in Equation (1), an obvious choice of fixed basis functions and coefficient vector for gene i are $\mathbf{h}(x_i(t))^{\mathrm{T}}=(1,-x_i(t))$ and $\beta=(\alpha_i,\lambda_i)^{\mathrm{T}}$, respectively. That is, with this choice, by combining Equations (1) and (9), we end up with the same form as in Equation (1). In order to be able to derive analytical expression for the marginal likelihood, we assign a Gaussian prior to the coefficient vector $\beta\sim\mathcal{N}(\mathbf{b},B)$, where $\mathbf{b}$ is a given mean and B is a covariance matrix. This choice of prior, however, makes it possible that parameters α_i,λ_i have negative values. We specify the prior rather conservatively so that all plausible values of β have sufficiently high probability. Such an approach provides us with a stable estimation where results are relatively insensitive to small fluctuations in the prior.

Combining the above model specifications together for gene i, we may write the transformation of $\mathcal{GP}\left(0,k(\mathbf{x}(t),\mathbf{x}(t'))\right)$ (Rasmussen and Williams, 2005)

$$\begin{aligned}&g_i\left(x_i(t),\hat{\mathbf{x}}_i(t)\right)\sim\\ &\mathcal{GP}\left(\mathbf{h}\left(x_i(t)\right)^{\mathrm{T}}\mathbf{b},k\left(\hat{\mathbf{x}}_i(t),\hat{\mathbf{x}}_i(t')\right)+\mathbf{h}\left(x_i(t)\right)^{\mathrm{T}}B\mathbf{h}\left(x_i(t')\right)\right),\end{aligned} \tag{10}$$

where $g_i\left(x_i(t),\hat{\mathbf{x}}_i(t)\right)$ represents the right-hand side of Equations (1) and (3), specifically,

$$\frac{\mathrm{d}x_i(t)}{\mathrm{d}t}\simeq\Delta x_i(t)=g_i\left(x_i(t),\hat{\mathbf{x}}_i(t)\right). \tag{11}$$

Intuitively speaking, this can be seen as a regression problem, where gene expressions $x_i(t)$ and $\hat{\mathbf{x}}_i(t)$ are used to model response variable $\Delta x_i(t)$. With Equation (11), we connect the core ODE model from Section 2.1 with the general theory of Gaussian processes from Section 2.2. After incorporating the given fixed basis functions, the mean function of the Gaussian process is not zero but is explicitly defined by the linear part of the model, which corresponds to the basal transcription rate and degradation terms. The non-linear part, i.e. the Gaussian process, corresponds to the (unknown) regulation function. Note that the predictive quantities as well as the marginal likelihood for noisy observations can still be computed similarly as in Equations (7) and (8)

$$\mathbf{g}_*|\Delta\mathbf{x}_i,X,X_*\sim\mathcal{N}(\bar{\mathbf{g}}_*,\mathrm{Cov}(\mathbf{g}_*)),\text{ where} \tag{12a}$$

$$\bar{\mathbf{g}}_*=H_*^{\mathrm{T}}\bar{\beta}+K_*^{\mathrm{T}}K_n^{-1}(\Delta\mathbf{x}_i-H^{\mathrm{T}}\bar{\beta})=\bar{\mathbf{f}}(X_*)+R^{\mathrm{T}}\bar{\beta} \tag{12b}$$

$$\mathrm{Cov}(\mathbf{g}_*)=\mathrm{Cov}(\mathbf{f}_*)+R^{\mathrm{T}}(B^{-1}+HK_n^{-1}H^{\mathrm{T}})^{-1}R, \tag{12c}$$

where $\bar{\beta}=(B^{-1}+HK_n^{-1}H^{\mathrm{T}})^{-1}(HK_n^{-1}\Delta\mathbf{x}_i+B^{-1}\mathbf{b})$ and $R=H_*-HK_n^{-1}K_*$, the vector $\Delta\mathbf{x}_i$ contains all the response variables $\Delta x_i(t)$ (both time-series and steady-state), and the matrices H and H_* contain all the training and test vectors $\mathbf{h}(x_i(t))$, respectively. Equation (12b) in scalar form can be utilized in the prediction of gene expression profile with the Euler method, i.e. given the expressions $x_i(t_n)$ and $\hat{\mathbf{x}}_i(t_n)$, approximate the rate of expression change $\Delta x_i(t_n)$ by $\bar{g}_i\left(x_i(t_n),\hat{\mathbf{x}}_i(t_n)\right)$. Similarly, the log-marginal likelihood of $\Delta\mathbf{x}_i$ can be written as

$$\log p(\Delta\mathbf{x}_i|X,\mathbf{b},B,\theta)=-\frac{1}{2}\mathbf{v}^{\mathrm{T}}W^{-1}\mathbf{v}-\frac{1}{2}\log|W|-\frac{1}{2}\log 2\pi, \tag{13}$$

where $\mathbf{v}=H^{\mathrm{T}}\mathbf{b}-\Delta\mathbf{x}_i$ and $W=K_n+H^{\mathrm{T}}BH$.

In the case of no explanatory variables [Equation (2)], it should be clear after incorporating the fixed basis functions that this model is based on Bayesian linear regression. To keep the two approaches [Equations (1) and (2)] consistent, we use the same Gaussian process framework for the case of no explanatory variables as well by incorporating another covariance function

$$k_{\mathrm{noise}}(\mathbf{x}_i,\mathbf{x}_j)=\sigma_n^2\delta_{ij}, \tag{14}$$

where σ_n^2 is again the hyperparameter representing the noise variance. Therefore, in this case, the Gaussian process is merely used to model the noise in the measurements, i.e. the covariance matrix K is diagonal.

Instead of full Bayesian treatment, we resort to an empirical Bayesian approach in model fitting where the hyperparameters, i.e. l, σ and σ_n^2 are optimized by maximizing the (log)-marginal likelihood with the Polack–Ribiere conjugate gradient method (Rasmussen and Williams, 2005). By taking a logarithm, one may see how the balancing between goodness-of-fit and complexity is carried out. Consequently, danger of overfitting the model is reduced in a natural way without introducing any statistical criteria from outside, such as Bayesian information criteria or Akaike information criteria.

2.4 Inference

As noted above, we have two goals: estimation of the non-parametric kinetic models and inference of the network structure. For a given model structure, regulatory functions can be estimated as shown in Equation (12a) and by subtracting the linear part. Bayesian model structure selection, where the goal is to choose explanatory variables $\hat{\mathbf{x}}_i$ for each gene i, can be obtained via the marginal likelihood shown in Equation (13). Let $\mathcal{M}_j$ denote a network structure. The posterior probability of a given model $\mathcal{M}_j$ can be obtained by applying Bayes' theorem

$$p(\mathcal{M}_j|\Delta\mathbf{x}_i,X)=\frac{p(\mathcal{M}_j)p(\Delta\mathbf{x}_i|X,\mathcal{M}_j)}{\sum_j p(\Delta\mathbf{x}_i|X,\mathcal{M}_j)p(\mathcal{M}_j)}, \tag{15}$$

where terms $p(\Delta\mathbf{x}_i|X,\mathcal{M}_j)$ are obtained by evaluating Equation (13) for each gene i (corresponding to the explanatory variables specified by $\mathcal{M}_j$) and $p(\mathcal{M}_j)$ is the prior probability of the network structure $\mathcal{M}_j$. For the purposes of this study, we use a uniform prior over networks. We use directly the posterior probabilities [Equation (15)] for ranking different models, i.e. which TFs regulate a given gene. Because the model selection relies on the marginal likelihood, the variable selection automatically favors models

that are explanatory but at the same time not too complex. Note that the computation of Bayesian posterior model probabilities in Equation (15) can be extended to compare different covariance functions as well.

The actual inference procedure is done separately for each gene in the network. That is, for each gene, we fit the model in the Equation (1) with different combinations of explanatory variables $\hat{\mathbf{x}}$ and compute the posterior probabilities using Equation (15). We summarize the posterior probabilities of network models using a square connection matrix, where the (i,j) element represents the posterior probability that gene j is regulated by gene i. Each element of the connection matrix can be computed by summing posterior probabilities of all networks that contain a directed connection from x_i to x_j.

2.5 Scalability

For a given network structure, the most time-consuming step is the computation of the marginal likelihood, which, for each gene, involves matrix multiplications, inversion of the covariance matrix (size n-by-n, where n is the number of measurements) and computation of the matrix determinant, an $\mathcal{O}(n^3)$ operation. Computational complexity is also increased by the iterative optimization of the hyperparameters. For moderately sized networks, we can perform an exhaustive search for model structures, resulting in $\mathcal{O}(N2^N)$ complexity, where N is the number of genes. An alternative strategy could be to implement a Markov chain Monte Carlo algorithm to sample network structures from the posterior. For larger networks, we may need to set an upper bound for the number of explanatory variables. However, because the explanatory variables can be searched for each gene separately, it is trivial to make use of distributed computing and, thus, to be able to infer GRNs with thousands of genes.

3 RESULTS

Validation of GRN network inference methods has traditionally been done using *in silico* networks. However, depending on how realistic the choice of an *in silico* model is, this kind of validation approach has obvious limitations. To overcome these problems, we use the IRMA network (Cantone *et al.*, 2009) to compare the performance of different GRN inference methods. The IRMA network is a synthetically constructed GRN in the *Saccharomyces cerevisiae* genome, which is designed to be maximally independent in such a way that genes in the network are not regulated by genes outside of the network (i.e. by other yeast genes). However, genes in the IRMA network may regulate other genes in the genome. The network consists of five genes and there are positive and negative feedback loops and one protein to protein interaction. For details on the construction of the network and experimental procedures, we refer to Cantone *et al.* (2009). One of the purposes of the IRMA network is to provide a realistic benchmark set for computational studies by providing mRNA-level measurements from a known GRN. To our knowledge, the IRMA network and dataset are the first of a kind that are meant for validation purposes. Although the IRMA network contains only five genes, there are about 33.6 million different networks structures. Further, there are studies suggesting that the performance on smaller networks typically generalize to larger networks (Bansal *et al.*, 2007; Stolovitzky *et al.*, 2007).

We use both switch-off and switch-on experiments (Cantone *et al.*, 2009), which refer to experiments where yeast cells were shifted from galactose to glucose and glucose to galactose, respectively. Galactose affects the network by activating the genes whereas glucose has an opposite effect. The time-series measurements were taken in switch-off and switch-on conditions, resulting in time series with a length of 20 for the switch-off and 15 for the switch-on. The steady-state data were measured in five different conditions where each of the genes was overexpressed in turn and this procedure was carried out for cells growing in both galactose and glucose media. In inference, we used averaged mRNA profiles and discarded the possibility of self-loops to keep the results comparable with those reported in Cantone *et al.* (2009). The hyperparameters of the basal transcription rate and the decay rate ($\mathbf{b}$) and the corresponding covariance matrix ($B=\sigma^2 I$) are the only fixed parameters of our approach and were set as specified in Supplementary Table 2. We use different variance hyperparameters for time-series and steady-state data to reflect the fact that the sample variance of the data varies by several orders of magnitude. Note that variance hyperparameters are small because the experimental data has remarkably small dynamic range.

Cantone *et al.* used the IRMA network to compare different commonly used network modeling approaches. In particular, they compared the performance of an ODE model (TSNI) and dynamic Bayesian networks (Banjo) on time-series measurements (switch-on and switch-off), and another ODE model (NIR), static Bayesian networks (Banjo) and an information theoretic method (ARACNE) on steady-state measurements (galactose and glucose). ODE-based methods TSNI and NIR were found to be top performers on time-series and steady-state data, respectively. In addition to comparing our results with those of TSNI, NIR and Bayesian networks, we also provide a comparison with the Bayesian method proposed in Zou and Conzen (2005).

The following metrics are used to assess the performance of each of the methods: precision–recall operating characteristic (P-ROC) curve, where positive predictive value [PPV = TP/(TP+FP)] is plotted against true positive rate [TPR = TP/(TP+FN)]] and receiver operating characteristic (ROC) curve, where TPR is plotted against false positive rate [FPR = FP/(FP+TN)].[1]

Supplementary Figure 1 shows the P-ROC and ROC curves for the proposed method when network structure is learned from the switch-on data. First of all, we notice that our method correctly identified five out of eight interactions without any false interactions. If one wants to find all true connections in the IRMA network, it will come with a cost of a few false connections. Second, the precision and recall values of the other methods are beneath our P-ROC curve.

Figure 2 shows the P-ROC and ROC curves related to the switch-off data. From this figure, we can conclude that our method works better as in the case of the switch-on data, which is surprising given the nature of the switch-off experiment, but this improvement might be due to greater number of timepoints. In this case also we find five connections out of eight without false positives, but we are able to find seven true positives with a smaller number of false positives than with the switch-on data. As in the case of switch-on dataset, the proposed method outperforms the other methods. The topology of the inferred network obtained with a threshold of $P=0.5$ on individual connections is shown in Figure 1. The two missing connections are regulatory interactions of CBF1 on GAL4 and SWI5 on CBF1 and the two extra connections are a regulatory effect of ASH1 on SWI5 (regulatory effect of SWI5 on ASH1 was found correctly) and a of SWI5 on CBF1. Even by looking at the expression profiles of SWI5 and ASH1 it is

[1]Full P-ROC and ROC curves can be obtained for the TSNI and NIR methods as well. However, we were not able to reproduce exactly the same results as in Cantone *et al.* (2009) and, hence, we only report the point estimates from Cantone *et al.* (2009).

difficult to say anything about the direction of regulation. Further, in steady-state experiments where each of the genes was overexpressed separately, overexpression of CBF1 did not have a significant impact on expressions GAL4 (Cantone *et al.*, 2009). It should also be noted that there is a significant delay in the activation of CBF1 (Cantone *et al.*, 2009), which might be the reason why the regulatory effect of SWI5 on CBF1 was not found.

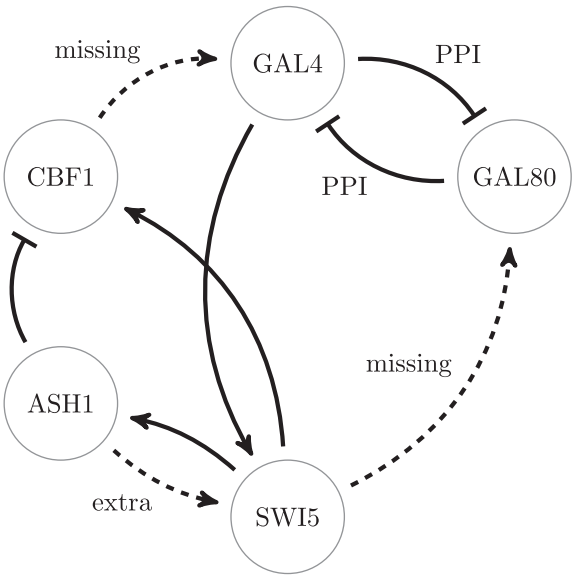

Fig. 1. The topology of the inferred IRMA network showing the correct (solid lines), missing and extra connections (dashed lines). The interaction between GAL4 and GAL80 is a protein to protein interaction and is considered to be bi-directional as in Cantone *et al.* (2009). Lines with arrow heads indicate activation and lines with blunt ends indicate inhibition.

One might expect that using all possible data together in the inference process should yield better results. To address this question, we applied the proposed method to the combination of switch-on and switch-off time-series datasets. Obtained results are shown in Figure 3. The results are better than the ones obtained with the individual datasets in a sense that we find all true connections with a smaller number of false positives. On the other hand, some true connections have contradicting evidence in the two datasets and those are ranked lower. Moreover, we also considered the GRN inference by allowing self-loops and it turned out that with individual datasets our method did not perform as well as without self-loops (but still better than random), i.e. self-loops had high probabilities. However, with the switch-on and switch-off datasets together the method performed well (Fig. 3). This observation suggests that the inference needs more data to distinguish between self-regulatory and non-self-regulatory interactions. This observation is also supported by the fact that genes in individual experiments exhibit only little dynamic variability (see, e.g. Fig. 4 and Supplementary Fig. 4).

Results for galactose and glucose steady-state datasets are shown in Supplementary Figures 2 and 3, respectively. We may see that in both cases, inference results obtained with steady-state data are worse than those obtained with the time-series data, despite the different perturbations (overexpressions) that were carried out. However, the obtained results are better than those reported in Cantone *et al.* (2009) with NIR. From the above remarks, we can conclude that time-series data contains more information about the network dynamics and that the proposed method is able to infer the network structure both from steady-state and time-series data.

In order to investigate the sensitivity of the fixed hyperparameters (**b** and B), we rerun the complete GRN inference simulations 1000 times with different hyperparameter values for $B=\sigma^2 I$. For each run, hyperparameters were chosen uniformly randomly

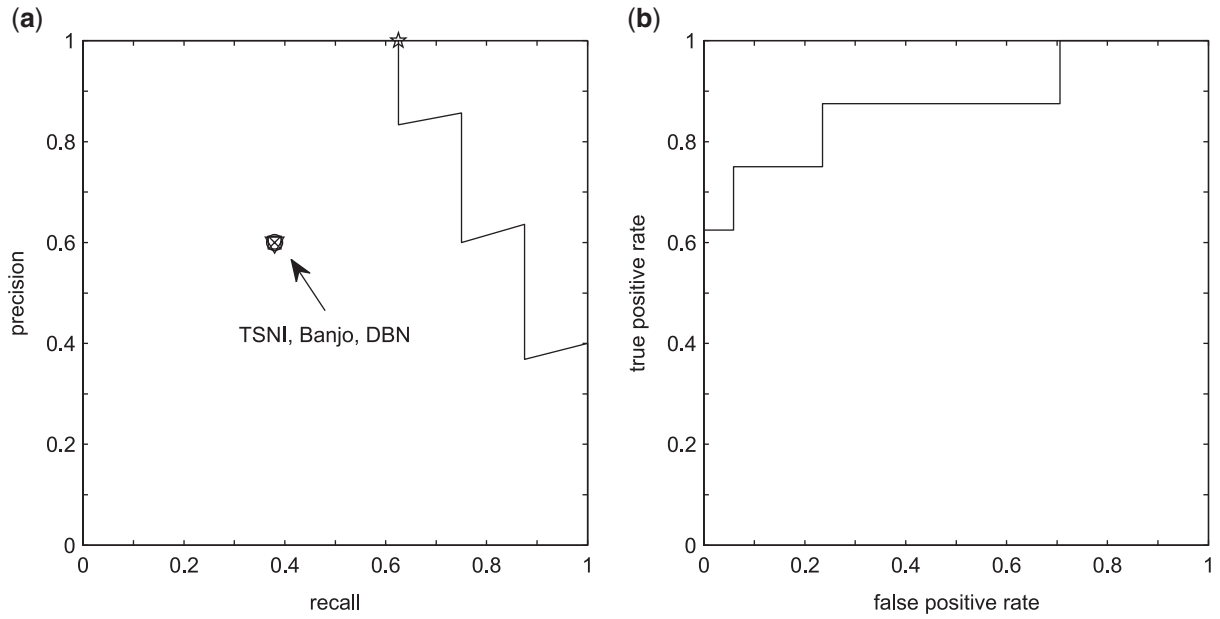

Fig. 2. Inference results on the switch-off dataset. Obtained P-ROC curve is shown in (**a**) and ROC curve is shown in (**b**). The cross in circle marks the results obtained with the TSNI method, the square marks the results obtained with the methods from Zou and Conzen (2005), the triangle marks the results obtained with Banjo and the star marks our results if we take five of the most probable interactions into consideration.

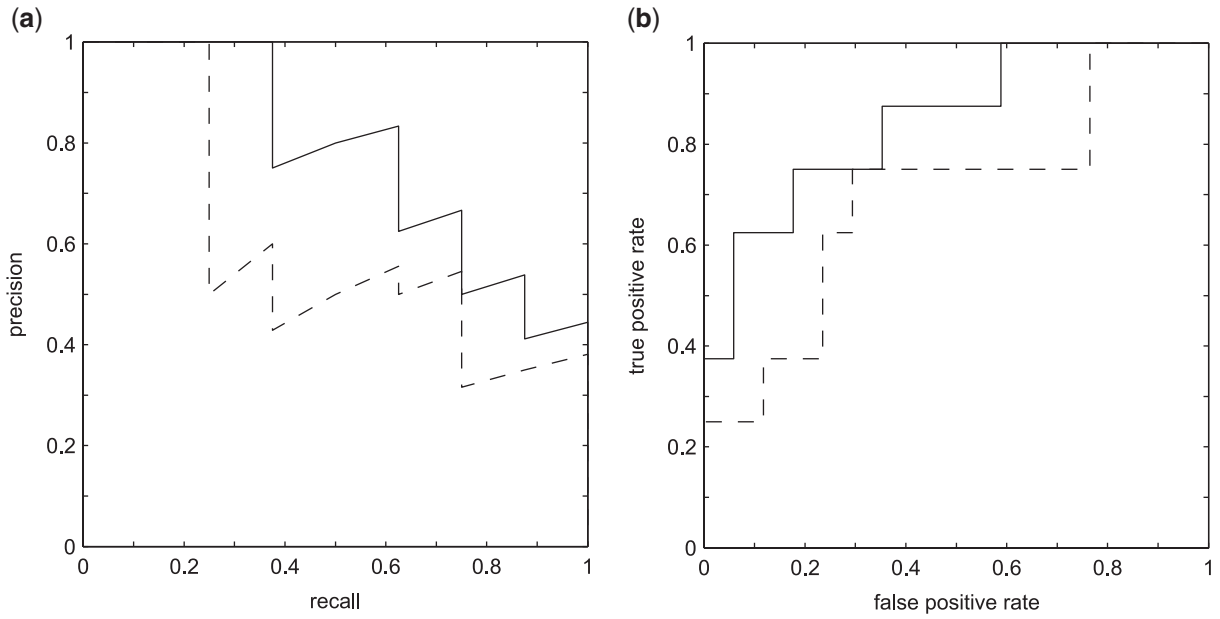

Fig. 3. Inference results on the combination of the switch-on and switch-off datasets. Obtained P-ROC curves are shown in (**a**) and ROC curves are shown in (**b**). The solid lines and dashed lines represent the results obtained without and with self-loops, respectively.

from the interval determined by increasing and decreasing the original hyperparameter value by a factor of two. Supplementary Figure 6 shows the box plots of the area under the ROC (AUROC) curves for different datasets. For time-series datasets, the hyperparameters have only a small effect on the structure inference results (AUROC numbers for the original hyperparameters are reported in Supplementary Table 1). Inference from steady-state data, however, is moderately sensitive to the value of the variance hyperparameter. This is most probably due to the very small sample size (only five steady-state measurements) and not an inherent problem of our computational method itself.

Part of the same data was also used in the second Dialogue for Reverse Engineering Assessments and Methods (DREAM2; Stolovitzky *et al.*, 2007), where the best inference results were obtained with an ODE method with Hill-type dynamics (Marbach *et al.*, 2009a, b). Although our method provides significantly better P-ROC curves, results may not be directly comparable because the amount of gene expression data in DREAM2 was slightly smaller and the network predictions for the DREAM2 were done in a completely blind fashion.

One of the goals of modeling GRNs is to obtain a predictive model. To assess the prediction capabilities of our method, we used the switch-off time-series dataset as a training set for both the model structure selection and for learning the dynamics of the model (i.e. non-parametric regulatory function and hyperparameters). We tried to predict the expression profiles in the switch-on experiment, which was not used to train the model, hence representing an independent validation dataset. We show the results for gene GAL4 as an example case. First, our method found out correctly that gene GAL4 is regulated by genes CBF1 and GAL80, i.e. this combination of regulatory genes had highest posterior probability. We assumed that expression of GAL4 at the first timepoint and expression profiles of regulatory (i.e. explanatory) genes at all time points are known. Figure 4 shows the measured expression profiles of genes GAL4, CBF1 and GAL80 and the predicted expression profile of gene GAL4 on the switch-off dataset. It can be seen that the predicted expression profile closely follows the measured one, which demonstrates that the proposed method also has predictive capabilities.

To address the question of validity of the proposed non-parametric ODE model more extensively, we look at the estimated regulatory function. In Figure 5a, the estimated regulatory function for gene GAL4 is shown along with the variance of the estimate in Figure5b. As above, gene GAL4 is regulated by genes CBF1 and GAL80 and the regulatory function is estimated from the switch-off dataset. From this figure, it can be seen that the method found out correctly that gene CBF1 is an activator and gene GAL80 is a repressor of gene GAL4. This observation demonstrates that the proposed non-parametric model is able to learn the regulatory role of different explanatory variables even in the case of combinatorial regulation.

In addition to the previous example, we predicted the expression profile of CBF1 on the switch-on dataset (see Supplementary Fig. 4). As in the previous example, the predicted expression profile resembles the measured one. Supplementary Figure 5 shows the estimated regulatory function on the test set (switch-on). The method again correctly infers the repressive and activating roles of ASH1 and SWI5, respectively.

In order to investigate the scalability of our method, we use an *in silico* dataset (networks with 100 genes) from the DREAM3 challenges. This dataset is computationally intensive due to the number of genes and the amount of data, i.e. 46 time series with 21 samples in each and null-mutants and heterozygenous steady-state measurements. Yet, we were able to get results with distributed computing in a few days for all five networks. However, given the

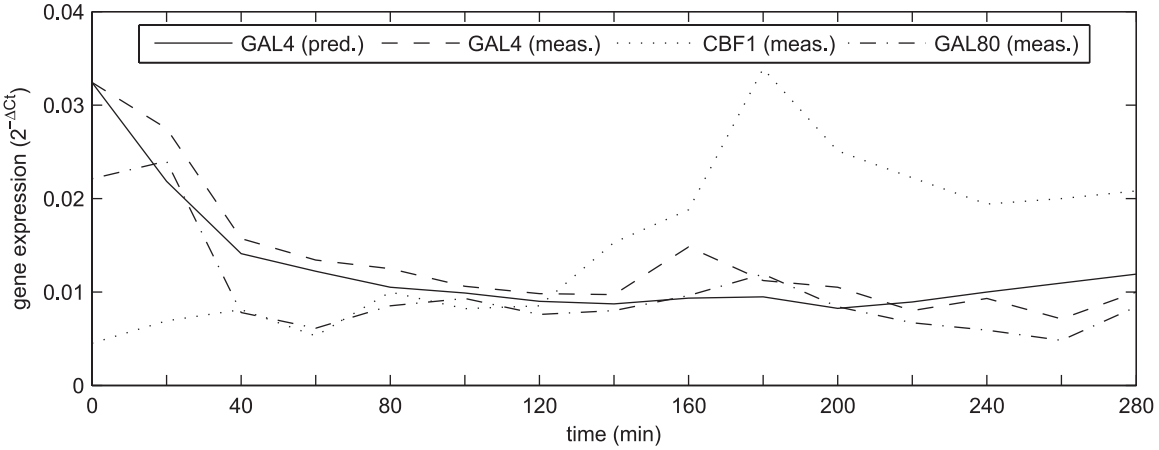

Fig. 4. Predicted expression profile of gene GAL4 on the switch-on dataset. Gene GAL4 is regulated by genes CBF1 and GAL80.

combinatorial explosion of the search space, which is independent of our model, we had to constrain the maximum in-degree in the models to two. Based on the results of the challenge, which are available from the DREAM project web site, we may conclude that the method presented in this article would have been placed second in the challenge with a score of 53.220. Obtained AUROC and P-ROC measures, the corresponding *P*-values as well as the ROC and P-ROC curves are shown in Supplementary Tables 3–4 and Supplementary Figures 7–11, respectively. In the two networks where the performance is weaker there are genes whose in-degree is high, and thus the weaker performance might be due to constraint on the maximum in-degree. In the three other cases, it should be noted that a number of the first predictions are correct.

4 DISCUSSION

We presented a novel ODE-based approach for inferring GRNs from steady-state or time-series measurements. The presented method does not rely on the assumption that regulatory function has a predefined shape, but the shape of the regulatory function is learned from the data. This provides additional flexibility in modeling but, at the same time, also allows modeling the standard parametric regulatory functions via the universal approximation property of non-parametric methods. In addition, the method is able to take uncertainty, e.g. noise in the measurements, into account in a well-defined manner. Bayesian analysis provides a principled way of comparing different network structures via the posterior probabilities. The proposed approach is able to make use of several time-series and steady-state datasets to improve the inference of transcriptional regulatory networks. Our method outperforms the TSNI and Zou and Conzen's (2005) methods on the time-series data, NIR (and ARACNE) on the steady-state data and dynamic and static Bayesian networks on time-series and steady-state data. Even though any ODE method is a simplification of the underlying biochemical system, the prediction results (Fig. 4 and Supplementary Fig. 4) and the estimated regulatory functions (Fig. 5 and Supplementary Fig. 5) demonstrate that the model is also able to capture the dynamics of the system. Possible future work includes extending the model in such a way that gene expressions are modeled in a continuous manner without approximating the derivatives and developing an adaptive methodology to extend the models in order to tackle the difficulties arising from constraining the maximum in-degree for large networks.

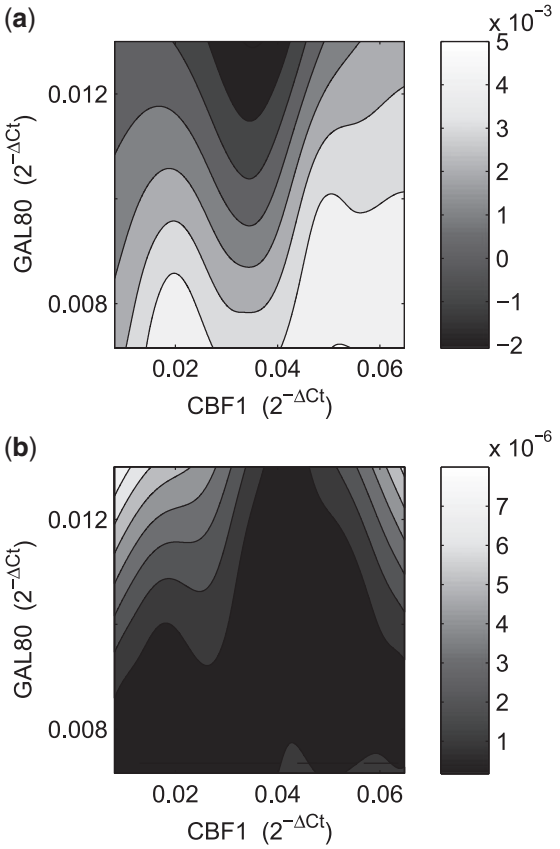

Fig. 5. Estimated regulatory function of GAL4 as a function of CBF1 and GAL80 evaluated on the training set (switch-off) is shown in (**a**) and the variance of the estimate in (**b**).

ACKNOWLEDGEMENTS

The authors are very grateful to Diego di Bernardo and his lab members (Telethon Institute of Genetics and Medicine) for providing the unique data and the reviewers for their constructive comments that helped to improve the manuscript. The authors wish to thank Timo Erkkilä, Antti Larjo and Monica Li for their careful reading and suggestions on the manuscript. This work benefited from the Tampere Center for Scientific Computing (TCSC) and Techila Technologies's grid computing solution.

Funding: This work was supported by the Academy of Finland, project no. 213462 (Finnish Programme for Centres of Excellence in Research 2006–2011); and the Finnish Foundation for Technology Promotion.

Conflict of Interest: none declared.

REFERENCES

Bansal,M. *et al.* (2006) Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. *Bioinformatics*, **22**, 815–822.

Bansal,M. *et al.* (2007) How to infer gene networks from expression profiles. *Mol. Syst. Biol.*, **3**, 78.

Barenco,M. *et al.* (2006) Ranked prediction of p53 targets using hidden variable dynamic modeling. *Genome Biol.*, **7**, R25.

Bonneau,R. *et al.* (2006) The Inferelator: an algorithm for learning parsimonious regulatory networks from systems-biology data sets de novo. *Genome Biol.*, **7**, R36.

Cantone,I. *et al.* (2009) A yeast synthetic network for in vivo assessment of reverse-engineering and modeling approaches. *Cell*, **137**, 172–181.

Cao,J. and Zhao,H. (2008) Estimating dynamic models for gene regulation networks. *Bioinformatics*, **24**, 1619–1624.

D'haeseleer,P. *et al.* (1999) Linear modeling of mRNA expression levels during CNS development and injury. In *Proceedings of Pacific Symposium on Biocomputing (PSB 99).* World Scientific, Singapore, pp. 41–52.

Gao,P. *et al.* (2008) Gaussian process modelling of latent chemical species: applications to inferring transcription factor activities. *Bioinformatics*, **24**, i70–i75.

Gardner,T.S. *et al.* (2003) Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, **301**, 102–105.

Greive,S.J. and von Hippel,P.H.V. (2005) Thinking quantitatively about transcriptional regulation. *Nat. Rev. Mol. Cell Biol.*, **6**, 221–232.

Imoto,S. *et al.* (2002) Estimation of genetic networks and functional structures between genes by using Bayesian networks and nonparametric regression. *Pac. Symp. Biocomput.*, **7**, 175–186.

Marbach,D. *et al.* (2009a) Combining multiple results of a reverse-engineering algorithm: application to the DREAM five-gene network challenge. *Ann. N. Y. Acad. Sci.*, **1158**, 102–113.

Marbach,D. *et al.* (2009b) Replaying the evolutionary tape: biomimetic reverse engineering of gene networks. *Ann. N. Y. Acad. Sci.*, **1158**, 234–245.

Margolin,A.A. *et al.* (2006) ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, **7** (Suppl. 1), S7.

Markowetz,F. and Spang,R. (2007) Inferring cellular networks–a review. *BMC Bioinformatics*, **8** (Suppl. 6), S5.

Nachman,I. *et al.* (2004) Inferring quantitative models of regulatory networks from expression data. *Bioinformatics*, **20** (Suppl. 1), i248–i256.

Perrin,B.-E. *et al.* (2003) Gene networks inference using dynamic Bayesian networks. *Bioinformatics*, **19** (Suppl. 2), ii138–ii148.

Rasmussen,C.E. and Williams,C.K.I. (2005) *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press.

Shmulevich,I. *et al.* (2002) Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks. *Bioinformatics*, **18**, 261–274.

Stolovitzky,G. *et al.* (2007) Dialogue on reverse-engineering assessment and methods: the DREAM of high-throughput pathway inference. *Ann. N. Y. Acad. Sci.*, **1115**, 1–22.

Wang,R.-S. *et al.* (2007) Inferring transcriptional regulatory networks from high-throughput data. *Bioinformatics*, **23**, 3056–3064.

Wilkinson,D.J. (2006) *Stochastic Modelling for Systems Biology*, 1st edn. Chapman & Hall/CRC.

Yu,J. *et al.* (2004) Advances to Bayesian network inference for generating causal networks from observational biological data. *Bioinformatics*, **20**, 3594–3603.

Zou,M. and Conzen,S.D. (2005) A new dynamic Bayesian network (DBN) approach for identifying gene regulatory networks from time course microarray data. *Bioinformatics*, **21**, 71–79.