British Journal of Applied Science & Technology
17(4): 1-18, 2016, Article no.BJAST.28154
ISSN: 2231-0843, NLM ID: 101664541

SCIENCEDOMAIN *international*
www.sciencedomain.org

State-space Modelling of Replicated Dynamic Genetic Networks

Anani Lotsi[1*] and Ernst Wit[2]

[1] *Department of Statistics, University of Ghana, Legon P.O.Box 115 LG, Ghana.*
[2] *Department of Statistics and Probability, Johann Bernoulli Institute, University of Groningen, Nijenborgh 9, 9747 AG Groningen, The Netherlands.*

Authors' contributions

This work was carried out in collaboration between both authors. Author EW conceived the idea, and author AL performed the statistical analysis, wrote the protocol, the first draft of the manuscript and managed literature searches. Both authors managed the analyses of the study, read and approved the final manuscript.

Article Information

DOI: 10.9734/BJAST/2016/28154
Editor(s):
(1) Qing-Wen Wang, Department of Mathematics, Shanghai University, P.R. China.
Reviewers:
(1) Chun-Liang Lin, National Chung Hsing University, Taiwan.
(2) Yagvalkya Sharma, Jaipur National University, India.
(3) Anonymous, The Institute of Medical Science, The University of Tokyo, Japan.
Complete Peer review History: http://www.sciencedomain.org/review-history/16311

Original Research Article

Received: 4th July 2016
Accepted: 25th August 2016
Published: 23rd September 2016

ABSTRACT

The genomic reality is highly complex and dynamic. Recent developments of high-throughput technologies have enabled researchers to measure the RNA abundance of thousands of genes simultaneously. The challenge is to unravel from such measurements genomic interactions and key biological features of cellular systems. Two common problems are the high-dimensionality of the system and the spurious correlations induced by unmeasured intermediate substrates. Furthermore most currently available models cannot deal with biological replicates. Our goal is to devise a method for inferring large transcriptional or gene regulatory networks from high-throughput data sources such as gene expression microarrays with potentially hidden states, such as unmeasured transcription factors (TFs).

**Corresponding author: E-mail: alotsi@ug.edu.gh*

Methodology: We propose a dynamic state space representation to account for the effects of such hidden states. Our inference method is based on a Kalman smoothing algorithm incorporated in the E-step of an EM algorithm. We employ bootstrap confidence intervals for inferring sparse networks, combined with an AIC criterion for determining the size of the latent space. The proposed method is applied to time course microarray data obtained from a well established T-cell experiment.

Keywords: Genomic interactions; microarray experiments; dynamic networks; state space representation; EM algorithm.

1 INTRODUCTION

Since the turn of the century a new scientific field has emerged: system biology has started to view biological processes as interrelated events, which ought to be understood in its entirety to make progress within the life sciences [1]. It is a biology-based, but interdisciplinary field that focuses on the systematic study of complex interactions in biological systems. The aim of this holistic approach is to discover new emergent properties that may arise from a systemic view, which are inaccessible to reductionist approaches. The concept of gene networks is central in system biology. A network is an abstract representation of a system, where the substrates or genes are seen as the nodes and the links as some kind of relationship, such as binding or some chemical reaction between them. It is an abstract representation of the stability and interconnectedness of molecular reactions. The challenge is to give this a precise statistical interpretation in order to allow one to be able to infer the network from quantitative observations on the nodes. Nowadays, expression levels of many genes can be measured simultaneously through many techniques including DNA hybridization arrays [2, 3] or RNA-seq methods [4]. A major challenge in system biology is to uncover, from such measurements, gene-protein interactions and key biological features of cellular systems.

The inverse problem of system biology requires a flexible statistical method that in a computationally efficient manner infers the complexity, the dependence structure of the network topology and the functional relationship between the genes. A lot of the statistical system biology literature only consider static networks [5, 6, 7]. In this paper, we will focus on the well-known linear state space models (SSM) [8, 9], which consider dynamic interactions across observed variables and non-observed states. Several authors have used Kalman filtering of SSM on gene expression data to reverse engineer transcriptional networks. [10] used a two-step approach. In the first step, factor analysis was employed to estimate the state vector and the design matrix; this resulted in the choice of the dimension of the state vector by means of BIC. In the second step, the matrix representing protein-protein translation was estimated using least squares regression. [11, 12, 13] have applied SSM to T-cell activation data, in which a bootstrap procedure was used to derive a classical confidence interval for parameters representing gene-gene interaction through a re-sampling technique. [14] approached the problem of inferring the model structures of the SSM using variational approximations in the Bayesian context. They used a Variational Bayes method to make the method computationally tractable and identify the dimension of the latent state. [15] also applied SSM to infer the topology of Gene regulatory networks. They introduce an empirical Bayes estimation procedure for a feedback state space model in a hierarchical Bayesian framework that is complementary to the method developed by [14]

Recently, [16] used SSMs to rank observed genes in gene expression time series experiments according to their degree of regulation in a biological process. Their technique is based on Kalman smoothing and maximum likelihood estimation to obtain estimates of the model parameters; however, little attention was paid to the dimension of the hidden state. [17] also presented a novel approach based on the state space model to identify the transcriptional modules and module-based gene networks simultaneously using SSM.

The common problem of all current statistical implementations of SSMs has been that they ignore the existence of biological replicates. [17, 18, 15] however used technical replicates of gene expression profiles, which are often measured in duplicate or triplicate. In the presence of biological replicates, time-series are typically averaged out within each time point and the SSM is applied to the average profiles. It is well-known however that replicated genomic time-series typically undergo a gradual phase-shift. These diffusion-like shifts are typically stochastic and not under any genomic control. Averaging out time-series will blur the genomic control and reduce the ability of correct inference. It is our aim to build a dynamic model of replicated dynamic RNA transcripts and unobserved quantities that represent (linear combinations of) commonly unmeasured protein regulators. We infer the model structure as a biological network by estimating model interaction parameters through the EM algorithm [19] combined with the Kalman smoothing algorithm [20, 21] in the context of maximum likelihood estimation. We use a bootstrap approach [22] to infer the complex transcriptional response of the network and to reveal interactions between components.

Choosing SSM to model network kinetics has a number of advantages. Most importantly, it allows the inclusion of hidden regulators, which can either be unobserved gene expression values or transcription factors (TFs). It can be used to model gene-gene and gene-protein interactions. The parameter estimates obtained through the EM algorithm and the state estimates from the Kalman filter have been shown to be consistent and asymptotically normal under some general conditions [23, 24]. In this paper, we demonstrate how the EM algorithm with the Kalman smoothing algorithm are used in the maximum likelihood set-up to reverse engineer transcriptional networks from gene expression profiling data. By so doing, we are able to add some useful interpretations to the model. The EM algorithm itself guarantees at least a monotonically increasing likelihood. Model selection or determining a suitable dimension of the hidden state is an additional complication. [12] approached the problem of deciding on a suitable dimension of the hidden state through cross-validation. In their approach, they continuously increased the dimension of the hidden states and monitored the predictive likelihood using the test data. One major drawback of this approach is that it is very slow and that it cannot be applied in an exploratory analysis of the data. As a result, we focus on faster information-based criteria.

The rest of the paper is organized as follows. In section 2, we introduce the model and give it a precise mathematical and biological interpretation. Crucially, we will focus on replicated genomic time-series that will undergo stochastic time-shifts. Section 3 describes the inference method including a model selection procedure for the regulating, but unobserved substrates. Identifiability issues of the model are also discussed and resolved through a minimal number of assumptions. In section 4 we assess via simulation the performance of our method extensively in terms of the F1-score, true positive and false positive rates under various scenarios. Section 5 consists of the application of our model to a well-studied T-cell data set through a bootstrap procedure where we identify the network kinetics, by identifying genetic regulatory networks. Importantly, our analysis makes explicitly use of the replicated time-series, which has been ignored thus far. We summarize the results, analyze their statistical significance and their biological plausibility. We conclude with a discussion of the method, possible extensions and a summary of related work in section 6.

2 GENOMIC STATE SPACE MODEL

Linear Gaussian state space models, also known as linear dynamical systems [25, 26], are a class of dynamic Bayesian networks that relate p temporal observations $y_t \in \mathbb{R}^p$ to k temporal hidden state variable $\theta_t \in \mathbb{R}^k$. We consider a sequence $(y_1, ..., y_T)$ of p-dimensional real-valued observation vectors through time, which we shall simply denote by $y_{1:T}$, representing a gene expression data matrix with p rows and T columns, where p and T are the number of genes and the measuring time points, respectively. The model assumes that the evolution of the hidden variables θ_t is governed by the state dynamics, which follows a first-order Markov

process and is further corrupted by a Gaussian intrinsic biological noise η_t. However, these hidden variable are not directly accessible to the experimenter but rather can be noticed through their effect on the observed data vector, y_t, the quantity of mRNA for each of the p genes at time t. The observation y_t is a linear transformation of a k-dimensional real-valued θ_t with observational Gaussian noise ξ_t. It is assumed that the entire experiment is replicated n_R times, resulting in the following model formulation:

$$\begin{cases} \theta_{tr} = F\theta_{t-1,r} + Ay_{t-1,r} + \eta_{tr} \\ y_{tr} = Z\theta_{t,r} + By_{t-1,r} + \xi_{tr} \end{cases} \tag{2.1}$$

where $r = \{1, 2, ..., n_R\}$, F, A, Z and B represent the model interactions parameters of dimensions compatible with the matrix operations required in (2.1). The terms η_t and ξ_t are zero-mean independent system noise and measurement noise, respectively with

$$E(\eta_t \eta_t^{'}) = Q, \qquad E(\xi_t \xi_t^{'}) = R \tag{2.2}$$

Both Q and R are assumed to be diagonal in many practical applications. The initial state θ_0 is independently Gaussian distributed with mean $a_0 = 0$ and covariance Q. This model is more complex and represents an extension of the standard SSM described in [27, 28, 10] as it includes various forms of feedback and can also be extended to include additional covariates.

The novelty of our approach as compared to other method such as [14] and [12] is the fact that we take biological replication into account. This is a crucial difference as can be seen in Fig. 1. In this simple 2 gene system with a single latent state, the expression of the same gene, i.e. 1, varies dramatically between replicates, although the underlying kinetics, given by

$$F = (0.9)\,,\ A = (-0.5, 0.5)\,,\ Z = \begin{pmatrix} 0.5 \\ -0.5 \end{pmatrix},$$

$$B = \begin{pmatrix} 1.0 & 0.5 \\ -0.3 & 1.0 \end{pmatrix},$$

are exactly the same, just as the covariance structure and initial states, given by

$$\theta_1 = 1,\ y_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix},\ Q = (0.02)\,,$$

$$R = \begin{pmatrix} 0.2 & -0.03 \\ -0.03 & 0.2 \end{pmatrix}.$$

It is clear that averaging such profiles would lose all biological meaning. Nevertheless, this is what most methods do currently. Our method takes the inter-replicate variability explicitly into account.

It must be noted that the method proposed in [15] is also capable of dealing with replicated time-series gene expression data. Their approach is based on an iterative empirical Bayesian procedure with the introduction of hyperparameters that estimate the posterior distributions of network parameters. We proposed a complementary method. The novelty in our paper is that we used a Maximum Likelihood inference approach which is a direct inference of the parameters and do not have to worry much about convergence problems . We also used AICc which is a data driven technique in estimating the dimension of the hidden state k.

A mathematical representation of the model is depicted in Fig. 2 indicating the latent and observed dynamics across 3 consecutive time points, where we assumed $k = p = 2$. The model in Fig. 2 assumes RNA-protein translation at two consecutive time points through the matrix A, and instantaneous protein-RNA transcription through Z. From a biological point of view, the model describes two fundamental stages in gene regulation which are in conformity with the central dogma which states information flows from DNA via RNA to proteins through transcription and translation. The observation-to-state matrix A, is of dimension $k \times p$, and models the influence or the effects of the gene expression values from previous time steps on the hidden states. Matrix B is the $p \times p$ matrix indicating the direct gene-gene interactions. The state dynamic matrix F describes the temporal development of the regulators or the evolution of the transcription factors from previous time step $t-1$ to the current time step t and is of dimension $k \times k$. It describes the influences of the hidden regulators on each other. The $p \times k$ observation dynamics matrix Z relates the transcription factors to the RNAs

at a given time point. We collect the model interaction parameters into a single parameter φ i.e $\varphi = \{G, Q, R\}$ where $G = \begin{bmatrix} B & Z \\ A & F \end{bmatrix}$ represents our genomic network of interactions including the hidden states. We must point out that [12] focused on the matrix $CB + D$ which is just direct gene-gene interactions. The corresponding $CB + D$ in our case is $ZA + B$.

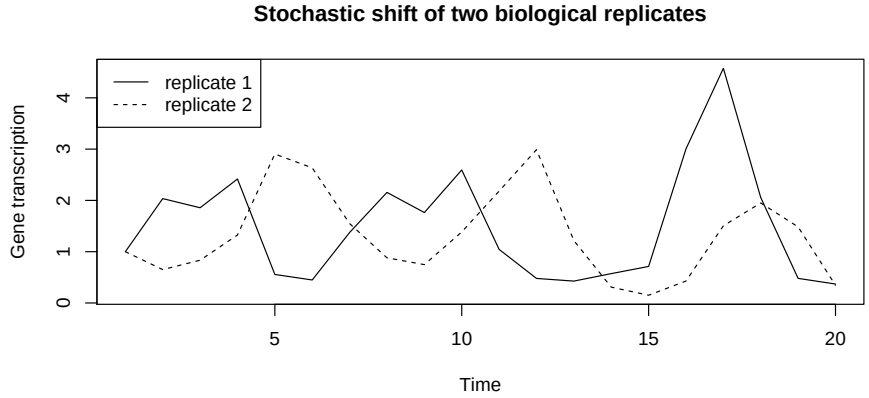

Fig. 1. The expression of two replicates of the same gene, simulated according to (2.1), whereby the underlying dynamics and initial values are the same, but the resulting profile gradually diverges. Averaging these profiles would lead to disastrous loss of information

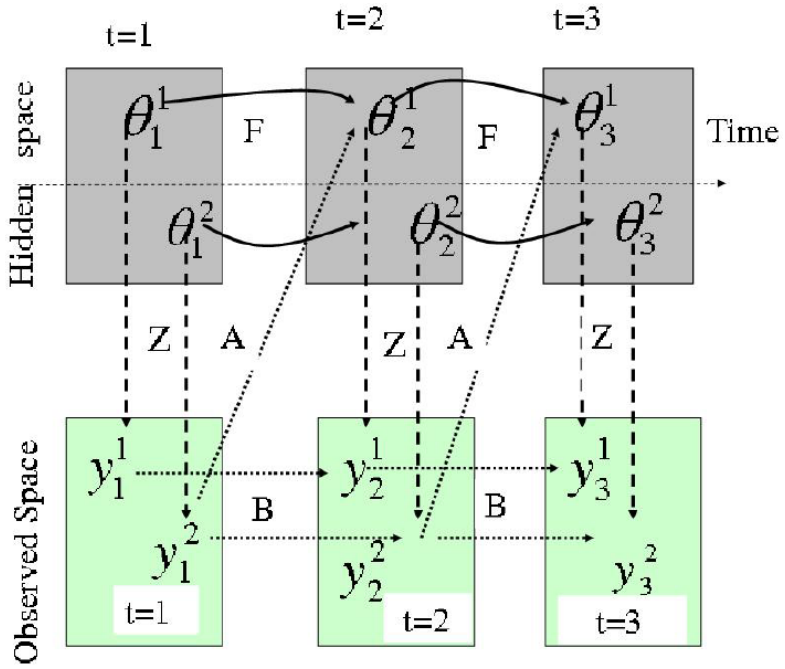

Fig. 2. A 2 gene network representing an input-dependent SSM for Gene regulation with the vector of observed gene expression (y_t) and the hidden regulators of gene expression (θ_t) at 3 different time points, where F, A, Z, and B correspond to the matrices in Equation (2.1)

2.1 Identifiability Issues

A parameter of a dynamic system is said to be identifiable if given some data only one value of this parameter maximizes the observed likelihood. The identifiability property is important because it guarantees that the model parameter can be determined uniquely and with a unique interpretation from the available data. Identifiability issues of the SSM stems from the fact that given the original model (Equation 2.1), and with the linear transformation of the state vector $\theta_t^* = T\theta_t$, where T is a non-singular square matrix, we can find a different set of parameter vectors

$$\hat{\varphi}^* = \left\{\hat{G}^*, \hat{Q}^*, \hat{R}^*\right\}$$

that give rise to the same observation sequence $\{y_t, t = 1, 2, ..., T\}$ having the same likelihood as the one generated by the parameter vector φ. Hence, if we place no constraints on F, A, Z, B and possibly Q and R, there exists an infinite space of equivalent solutions $\hat{\varphi}$ all with the same likelihood value. To overcome such identifiability issues, further restrictions have to be imposed on the model. In our work, we assume Q to be an identity matrix and R is set to be diagonal matrix. Subjecting Q to be identity only affects the scale of θ and matrices A and Z.

We further assume that the errors $\{\eta_t, t = 1, ..., T\}$ and $\{\xi_t, t = 1, ..., T\}$ are jointly normal and uncorrelated. Also the number of time points or biological replicates in microarray data are typically much smaller than the number of genes. This fundamental problem of high-dimensional statistical modelling of micro array data demands additional care in the estimation of the model parameters in the state space model. This problem is avoided by requiring that the number of observations exceed the total number of parameters to be estimated, i.e.,

$$pTn_R > p^2 + 2kp + k^2. \tag{2.3}$$

Recall that p is the number of genes, T is the measuring time points, n_R is the number of replicates hence we have pTn_R as total number of observations. Next, according to our model, we have B, Z, F, and A as parameters to estimate but B is a matrix of dimension $(p * p)$ i.e p^2, Z is of dimension $(p * k)$, A is of dimension $(k * p)$, F is of dimension $(k * k)$ i.e K^2. This gives the total number of parameters to estimate as $p^2+2kp+k^2$. Clearly we will have wished that we have enough data points to enable us estimate the parameters in our model, hence Eq(2.3) which can be seen as a quadratic equation in k. Solving equation Eq(2.3) for k, the number of latent states puts the following bound on the dimension of the hidden states,

$$0 \leq k < -p + \sqrt{pTn_R}, \tag{2.4}$$

for which the system is still identifiable especially for large number of replicates.

2.2 The Likelihood Function

With the identifiability constraints from the previous section, we can now write the model parameters as φ =$\{G, R\}$. As can be seen from Fig. 2, the observations at time t, y_{tr} are conditioned on the past observations, $y_{(t-1)r}$ and on the regulators θ_{tr}. The latent state θ_{tr} depends on $\theta_{(t-1)r}$ and $y_{(t-1)r}$. To that effect, we can assume

$$\begin{aligned}
\theta_{0r} &\sim N_k(0, I) \\
y_{0r} &\sim N_p(0, R) \\
\theta_{tr}|\theta_{(t-1)r}, y_{(t-1)r} &\sim N_k(\tilde{\theta}_{tr}, I) \\
y_{tr}|\theta_{tr}, y_{(t-1)r} &\sim N_p(\tilde{y}_{tr}, R),
\end{aligned}$$

where

$$\begin{aligned}
\tilde{\theta}_{tr} &= F\theta_{(t-1)r} + Ay_{(t-1)r}, \\
\tilde{y}_{tr} &= Z\theta_{tr} + By_{(t-1)r},
\end{aligned}$$

and $N_d(\mu, \Sigma)$ is the d-dimensional normal distribution with mean μ and covariance matrix Σ. We write the marginal likelihood function $L_y^r(\varphi)$ of the data is given by

$$
\begin{aligned}
L_y^m(G,R) &= \prod_{r=1}^{n_R}\int\prod_{t=1}^{T} P(\theta_{tr}|F,A,\theta_{(t-1)r},y_{(t-1)r})P(y_{tr}|B,Z,\theta_{tr},y_{(t-1)r})d\theta \\
&= \prod_{r=1}^{n_R}\int\prod_{t=1}^{T} \phi(\theta_{tr}|\tilde{\theta}_{tr},\sigma_\eta^2 I)\phi(y_{tr}|\tilde{y}_{tr},\sigma_\xi^2 I)d\theta. \qquad (2.5)
\end{aligned}
$$

Maximizing (2.5) across the parameters is extremely challenging, as it involves an integral across the latent state. Therefore, we first consider the complete log-likelihood function of the augmented data (y_{tr},θ_{tr}), given as

$$
l_{y,\theta}(G,R) = \sum_{r=1}^{n_R} l^r_{y_r\theta_r}(F,A,Z,B), \qquad (2.6)
$$

where the complete log-likelihood of the r^{th} replicate $l^r_{y_r\theta_r}(F,A,Z,B)$ is given by

$$
\begin{aligned}
l^r_{y_r\theta_r}(F,A,Z,B) &= \sum_{t=1}^{T} l_{y_t|\theta_t,y_{(t-1)}}(Z,B) + \sum_{t=1}^{T} l_{\theta_t|\theta_{(t-1)},y_{(t-1)}}(F,A) \\
&= -\frac{1}{2\sigma_\xi^2}\sum_{t=1}^{T}(y_t-\tilde{y_t})'(y_t-\tilde{y_t}) - \frac{T}{2}\log(\sigma_\xi^2) \\
&- \frac{1}{2\sigma_\eta^2}\sum_{t=1}^{T}\left(\theta_t-\tilde{\theta_t}\right)'\left(\theta_t-\tilde{\theta_t}\right) - \frac{T-1}{2}\log(\sigma_\eta^2) \qquad (2.7)
\end{aligned}
$$

ignoring constant term.

2.3 Joint Parameter Estimation via EM Algorithm

The EM algorithm used in [12] is developed for a single replicate. Extension of this to multiple replicates is non-trivial and non-automatic. There are various ways to extend the SSM to multiple replicates. In our case we assume that each replicate has its own associated latent space governed by the same global parameters (i.e. F and A). The reason is that biological replicates have their own internal clock, governed by universal biological constraints. Our aim is to estimate the model parameters G, which represents the underlying directed genomic network, by maximizing the marginal likelihood function $l_y^m(\varphi)$ given in Equation 2.5. Due to the intractability of the integral, we resort to using the EM algorithm [29, 30] to learn the parameters of the model.

2.3.1 The EM-algorithm

The main method of inference used in this paper is the Expectation- Maximization (EM) algorithm. Frequentist estimation by and large relies on maximum likelihood (ML) estimators. It consists of maximizing the likelihood across the parameter space. Special cases of the EM algorithm were developed before it was formally introduced by [19]. The EM algorithm has become a popular method of inference in statistical estimation problems involving incomplete data, i.e, data with some missing or latent or hidden observations or problems that can be posed in a similar form, such as mixture models.

The EM algorithm is an iterative tool to compute the maximum likelihood estimate in data characterized by the presence of missing, or hidden or latent observations. This optimization can be difficult especially if the data consist of

missing or latent parts. The intuition behind ML is to estimate the parameter(s) for which the observed sample is most likely. It possesses some optimality properties as discussed in [31]. Each iteration of the EM algorithm consists of an expectation step (E-step) followed by a maximization step (M-step). In the E-step, the hidden variables are "estimated" as conditional expectations given the observed data and current estimates of the model parameters. In our SSM, the Kalman filtering algorithm is precisely the E-step. The later is achieved by computing the conditional expectation of the (log) likelihood of the "complete" data. The M-step maximizes the complete likelihood function across the parameter space given the estimate of the missing data from the E-step.

To this effect the algorithm requires the computation of the conditional expectation of the log-likelihood given the complete data. The algorithm is a two-stage procedure, which alternates by calculating the Kalman smoother in the E-step and updating the model parameters in the M-step. The algorithm alternates recursively between an expectation and maximization steps until convergence is obtained.

The procedure to obtain the maximum likelihood estimator of the parameter vector φ via the EM-algorithm is summarized below:

1. Select initial values of $(\hat{\varphi}_0)$ that is, start with initial guess for the parameters $\hat{\varphi}_0$
2. At the k^{th} step, calculate the conditional expectation of the log likelihood (E-step)
3. Determine the next iterative estimated parameters $(\hat{\varphi}_{k+1})$ that maximizes conditional expectation of the log likelihood. (M-step) and compute the corresponding log likelihood.
4. Iterate step 2 and 3 until the log likelihood is converged.

2.3.2 The expected log-likelihood function: The E-step

The E-step step of the EM algorithm involves the calculation of the first two moments of the hidden states θ_t. Let $\mathbf{Q}$ denote the expected log-likelihood. Then from Equation 2.6, $\mathbf{Q}$ becomes

$$\begin{aligned}\mathbf{Q}(\varphi|\varphi^*) &= E_\theta\left[l_{y,\theta}(\varphi)|y,\varphi^*\right] \\ &= \sum_{r=1}^{n_R} E_\theta\left[l^r_{y_r,\theta_r}(Z,B)|y,\varphi^*\right] + \\ &\quad \sum_{r=1}^{n_R} E_\theta\left[l^r_{y_r,\theta_r}(F,A)|\varphi^*,y\right] \\ &= Qb_1(Z,B) + \mathbf{Q}_2(A,F) \qquad (2.8)\end{aligned}$$

where $\varphi^* = (Z^*, B^*, F^*, A^*)$ is the estimate obtained from the previous M-step

The calculation of $\mathbf{Q}(\varphi|\varphi^*)$ in Equation 2.8 involves finding $E(\theta)$ and $E(\theta^{'}\theta)$ for each replicate r. These forms are readily found: for each replicate we run the Kalman smoothing algorithm to obtain the expected hidden states and their variance-covariance components and these are joined together to get $\mathbf{Q}(\varphi|\varphi^*)$.

The Kalman-Filtering Algorithm

The Kalman filter has been considered as one of the optimal solutions to many data prediction, filtering and smoothing problems. In this context, it is used to estimate the hidden or latent states in the E-step of the EM algorithm. We describe here the basic concepts that one needs to know to design and implement a Kalman filter algorithm. Given our original model from equation (2.1)

The predictive step equations are given by:

$$\tilde{\theta}_t = F\hat{\theta}_{t-1} + Ay_{t-1} \qquad (2.9)$$

$$\tilde{P}_t = F\hat{P}_{t-1}F^{'} + \eta_t \qquad (2.10)$$

where $\tilde{P}_t$ represents the corresponding predicted or prior state estimate error covariance.

Then the observation prediction equation step becomes:

$$\tilde{y}_t = Z\tilde{\theta}_t - By_{t-1} \qquad (2.11)$$

$$\Sigma_t = Z\tilde{P}_t Z^{'} + R \qquad (2.12)$$

where Σ_t represent the observation prediction covariance.

with

$$v_t = y_t - \tilde{y}_t \qquad (2.13)$$

denoting the measurement innovation or the residual and reflects the discrepancy between

the predicted measurement $\tilde{y}_t$ and the actual observation y_t

The filtered equations are also given by

$$\hat{\theta}_t = \tilde{\theta}_t + K_t v_t \tag{2.14}$$

$$\hat{P}_t = \tilde{P}_t - K_t \Sigma_t K_t^{'} \tag{2.15}$$

and

$$K_t = \tilde{P}_t Z^{'} \Sigma_t^{-1} \tag{2.16}$$

is the Kalman gain matrix and is chosen to be the gain or blending factor that minimizes the posterior error covariance in Equation (2.15).

Finally the smoothing step is given by

$$\hat{\theta}_t^T = \hat{\theta}_t + H_t \left(\hat{\theta}_{t+1}^T - \tilde{\theta}_t^T \right) \tag{2.17}$$

where H is the Kalman smoothing matrix. More information on K and H can be obtained from [20, 21] Equation (2.16) represents the expected hidden states needed at the E-step.

2.3.3 The update equations: The M-step

A new parameter set φ^* is computed by estimating the parameters that maximize the two quadratic forms in (2.8). We solve $\frac{\partial}{\partial \varphi}\mathbf{Q}_1 = 0$ and $\frac{\partial}{\partial \varphi}\mathbf{Q}_2 = 0$ to obtain estimates for Z, B and F, A, respectively. This can be solved in closed form. For a full derivation, see the appendix. The entire EM algorithm can be regarded as alternating between Kalman smoothing and least squares minimization given by the update equations.

2.4 Choice of Hidden State Dimension: AIC_c

Model selection or the determination of the optimum dimension of the hidden state k is important in the application of SSM to network reconstruction. Popular model selection criteria include Akaike's Information Criterion (AIC) [32] and the Bayesian Information Criterion (BIC) [33]. We apply a corrected Akaike's Information Criterion (AICc) method in our scenario. The AICc has good model estimation properties, especially for small sample time-series data [34, 35]. Furthermore, the CV approach used in [12] tends to be slow and unstable for small number of replicates.

The AICc is aimed at finding the best approximating model to the unknown data generating process via minimizing the estimated expected Kulback-Leibler divergence. Given the log-likelihood function l, the AIC for a model with k-dimensional state vector is given by:

$$AIC(k) = -2l(y_t|\hat{\varphi}_k) + 2P \tag{2.18}$$

with P the number of estimated parameters, and $l(y_t|\hat{\varphi}_k)$ the log-likelihood of the observed data. [36] recommends the use of the corrected AIC, which corrrects for small sample size bias. In the framework of normal linear regression models, the penalty term of AICc provides an exact expression for the bias adjustment. The AIC_c is given by

$$AIC_c(k) = -2l(y_t|\hat{\varphi}_k) + 2P\left[\frac{N}{N-P-1}\right] \tag{2.19}$$

where $N = pTn_R$ represents total number of observations and $P = p^2 + 2kp + k^2$ is the total number of estimated parameters. We select the hidden state dimension that has the minimum AIC_c, i.e we find k such that

$$k = \arg\min_k \left\{AIC_c(k)\right\}. \tag{2.20}$$

We successively increase the number of hidden states and monitor the behavior of AICc as a function of k.

2.5 Network Reconstruction by Bootstrapping

We use a bootstrap approach to find confidence intervals for the parameters defined in our model. By so doing we compute the bootstrap distribution of the estimator of φ. Let $\hat{\varphi}$ denote the MLE of the parameters defined in our model that come from using the EM algorithm described in previous section. In the following we will use the notation $y_r \in \mathbb{R}^{P\times T}$ with $r \in \{1, ..., n_R\}$ to represent each of the biological time series. The bootstrap procedure adopted is outlined below:

1. Obtain the Kalman filter model fit $\tilde{y}_1, ...\tilde{y}_{n_R}$.

2. Calculate, for all r in $\{1, ..., n_R\}$, the innovation errors $\xi_r = y_r - \tilde{y}_r$.
3. Sample with replacement from $\{\xi_r\}$ to obtain ξ_r^*
4. For all r in $\{1, ..., n_R\}$, generate a bootstrap sample y_r^* through $y_r^* = \tilde{y} + \xi_r^*$

Whereas we bootstrap the residuals in our method, [12] bootstrapped the original observations. As our approach is non-parametric, the two methods result in the same bootstrapping procedure. Given each new data we estimate, among other things, the bootstrap set of parameters $\left\{\hat{\varphi_b^*}; b = 1, ..., N_b\right\}$ through the EM algorithm. Stated differently, for each bootstrap sample the parameters that maximize the likelihood of the bootstrap data are found. We then obtain the sampling distributions of the estimators of the elements of φ. The results of the bootstrapping are the distribution of the parameters and we proceed to make statistical inferences about those underlying parameters by computing confidence interval for each of them [37, 20].

3 SIMULATION STUDIES

In order to illustrate the performance of our method for analyzing gene expression data, we simulate artificial data and applied our proposed method to these data according to the model described in Equation 2.1 with $T = 10$ time points, $p = 3$ genes and $k = 2$ latent states. The true newtork is depicted on the left in Fig. 3.

In the initialization step of the EM-algorithm, Z and F are assumed to be identity matrices, whereas A to set to zero. For B we perform a simple linear regression where we regress all observed genes on the previous time point. The diagonal variance matrix R is given the usual variance estimate coming from the regression. We apply the bootstrap procedure to the data and identify the significant and non-significant parameters through the use of bootstrap confidence intervals on the element G_{ij} of the network. For this decision problem where we formulate two hypotheses, namely,

$$H_0 : \quad G_{ij} = 0$$
$$H_1 : \quad G_{ij} \neq 0$$

where rejecting H_0 indicates the presence of a connection among the gene i to j. With k equals 2, we obtained network shown on the right in Fig. 3.

Table 1 depicts the performance of our method as the size of the network increases. Comparison is also made to the method of [12], in which no replicates are considered. In order to deal with the replicates, an average profile across 50 biological replicates is calculated. When p increases, our method is able to detect the network more accurately. The reason is that the number of latent states $k = 2$ for larger network in terms of p becauses relatively speaking smaller, which makes network detection easier. Ignoring the inter-replicate variability, however, means that, unlike it is the case for our method, there is no gradual improvement in network detection for the other SSM method.

4 APPLICATION

For this study, to demonstrate the application of our network inference method, we used publicly available data. Two separate experiments investigated the expression response of human T-cells to PMA and ionomicin treatment. The entire data set is a combination of the datea from these two experiments. The first data set (tcell.34) contains the temporal expression levels of 58 genes for 10 unequally spaced time points. At each time point there are 34 separate measurements. The second data set (tcell.10) comes from a related experiment considering the same genes and identical time points, and contains 10 further measurements per time point. Therefore, at each time point there are 44 separate measurements or replicates. Corresponding to each gene expression y_{tr}, we also assumed the existence of generative replicates for the hidden variables θ_{tr}. With $p = 58$ genes, $R = 44$ as replicates and $T = 10$, the constraint represented by Equation (2.3) is satisfied, indicating that we have enough data to estimate our parameters. The dimension of the hidden variables was determined using AICc as explained in section 2.4. Table 2 shows the behavior of AICc with corresponding k's. It turns out that $k = 4$ is the optimum number of the hidden states. This is fewer than [12] and [14]

who obtained 9, 14 respectively under different criteria.

In essence, we treated the data as a time series measurement data y_{t_r}, $t = 1, 2, ..., 10$ and $r = 1, 2, ..., 44$. For each replicate, y_t and θ_t consist of 58 genes and 4 transcriptions factors respectively, each, measured at 10 different time points, i.e for each replicate r, y_t and θ_t are of dimension (58×10), (4×10) respectively. Some of these genes include RB1, CCNG1, TRAF5, CLU.... The parameters Q were fixed.

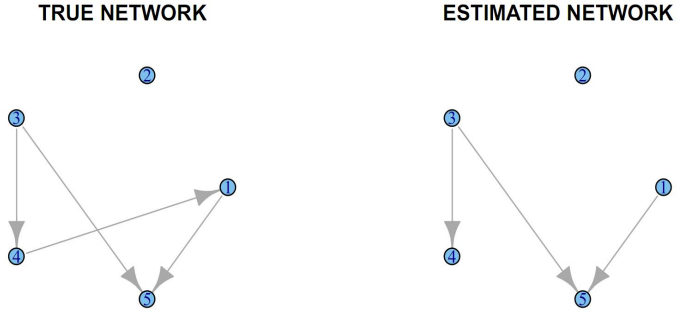

Fig. 3. The true network G (left) and recovered network (right) $\hat{G}$

Table 1. Simulation results showing the average scores for true positive rates (TPR), false positive rates (FPR) and F_1-scores as the number of nodes p increases. We compare the performance of our method and the method of [12], whereby for the latter it is necessary to average across the replicates. The TPR, FPR and F_1 are average scores across 50 simulations. The numbers in parentheses represent the standard deviations. In each simulation $T = 10$ is the number of time points, $n_R = 50$ the number of replicates and $k = 2$ the number of hidden states

p	10		20		60	
method	Our	Rangel	Our	Rangel	Our	Rangel
TPR	0.558	0.15	0.935	0.213	0.961	0.123
	(0.483)	(0.000)	(0.072)	(0.369)	(0.001)	(0.410)
FPR	0.098	0.000	0.017	0.003	0.006	0.002
	(0.040)	(0.000)	(0.013)	(0.003)	(0.0007)	(0.002)
F_1-score	0.410	0.208	0.827	0.250	0.816	(0.350)
	(0.358)	(0.360)	(0.115)	(0.433)	(0.005)	(0.391)

Table 2. For the selection of the dimension of the latent states in the state space model, we calculate the AIC_c as a function of the number latent states k

k	2	3	4	5	6	8	10
AICc	3,386,201	2,537,048	2,524,402	2,849,645	2,800,490	2,884,533	3,137,672

Based on 95% confidence intervals to detect significant interactions, we plot the connectivity matrix of the directed genomic network $\hat{G}$. The output is a directed graph showing connections from one gene expression variable at a given time point t to another gene expression variable whose expression it influences at the next time point, $t+1$. The arrows indicates the direction of the regulation. The entire directed graph $\hat{G}$ gives 350 genomic interactions. Fig. 4 represents a portion of the interaction network $\hat{\varphi}$ where we indicate genes that have at least 3 outwards connections. These genes include the FYN-binding protein gene FYB, the JUND proto-oncogene, the CD69 antigen p60, early T-cell activation antigen to mention but a few. Fig. 5 is the sub-network produced at 95% confidence level and it represents the interaction between, two Jun proteins family namely JUNB and JUND and various genes involved in programmed cell death. The results of our method in Fig. 5 support the hypothesis of the anti-proliferation and anti-apoptotic role of JUND.

According to our method, the following genes were mostly seen as regulatory genes. These genes include the JUND proto-oncogene, the CLU gene, the cell division cycle 2 CDC2, the FYN-binding protein gene FYB, TRAF5, the CD69 and the GATA-binding protein 3. The latent variables were also seen to regulate the expression level of most genes as can be seen in Fig. 4.

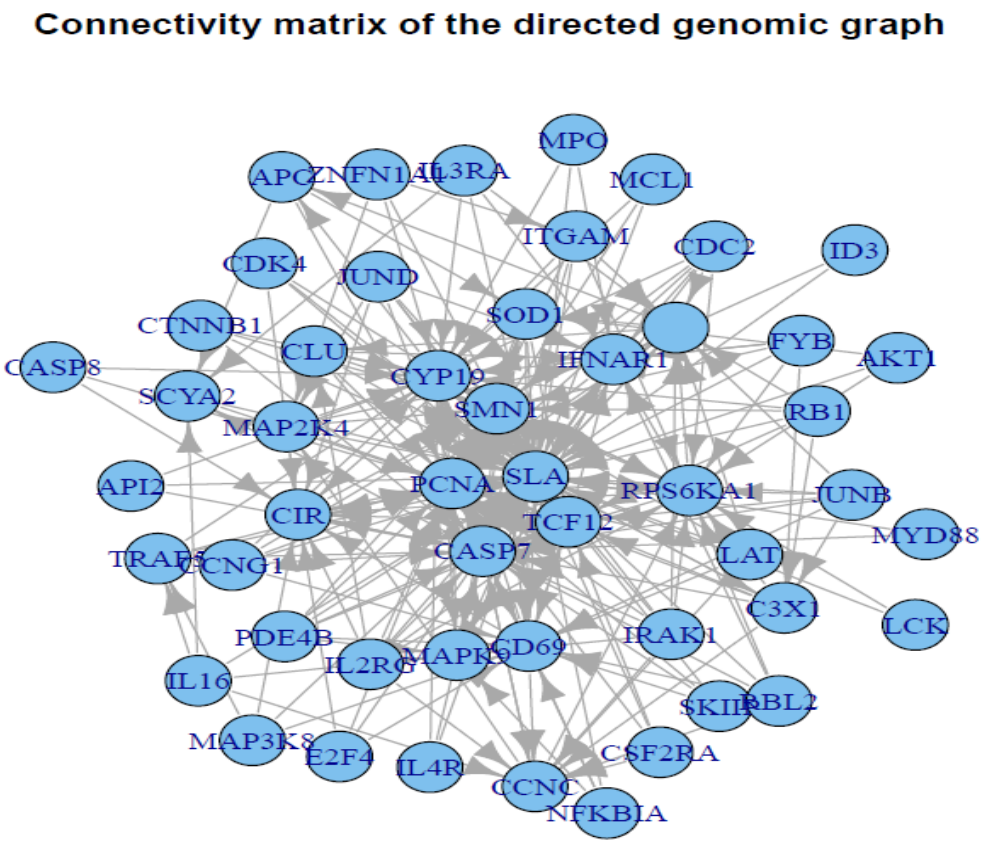

Fig. 4. Sub-network found representing the genomic interactions $\hat{G}$, of genes with at least 3 outwards connections, nodes refer to gene expression in the form of proteins or RNAs; empty nodes refer to latent variables

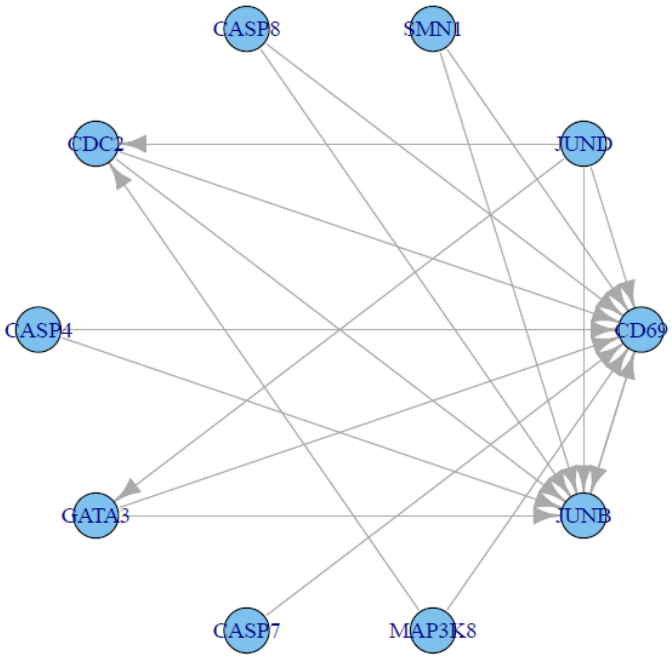

Fig. 5. Sub-network found representing the interactions between Jun proteins family and apotostic genes

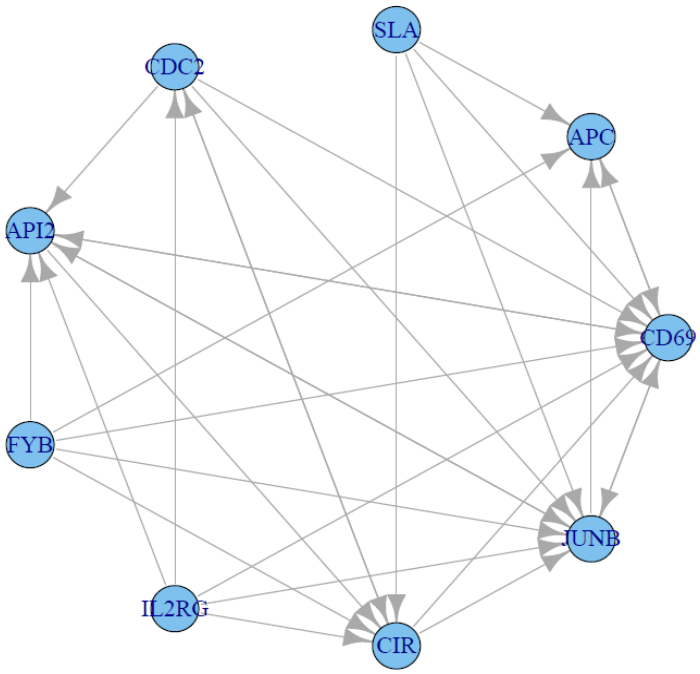

Fig. 6. Sub-network found representing the topology of gene FYB in connection with other genes

Our approach has revealed interesting features in the family of Jun genes. The network in Fig. 5 provides support for several hypotheses that were also confirmed in [12] and [14]. However, we also found new connections. Our results support the interaction between the proto-oncogene JUNB, the apoptosis-related cysteine protease genes CASP4 and CASP8. The implication is that JUNB is clearly modelled as a pro-apoptotic gene by activating CASP4 and CASP8. This interaction was also was recovered by [14]. We, however, found no evidence for interaction between JUNB and MAP3K8. Also Fig. 5 reveals that the proto-oncogene JUND activates the GATA-binding protein 3, but represses the expression level of the cell division cycle 2 (CDC2). This further supports the anti-proliferative JUND. Furthermore, in our model, the survival of motor neuron 1 gene SMN1 and the cell division cycle CDC2 influence the expression level of JUNB and MAPK8 respectively. JUNB activates the expression level of CDC2. A critical comparison of our Fig. 5 to that of similar sub-networks found in the work of [15] and [14] shows that in all the 3 sub-networks, JUND regulates the expression level of CDC2. JUNB activates CASP8 in the sub-network found by [14] and indirectly regulates CASP8 through CASP4 in the sub-network found by [15]. However we found interaction between JUNB and both CASP8 and CASP4.

The gene FYN-binding protein FYB has been found to occupy one of the most crucial positions in the network recovered by [12] also it has a high degree of connectivity in our work. Fig. 6 reveals some crucial genes that are found to be directly connected to FYB. Most importantly, in our model FYB influences the expression level of genes such as the early T-cell activation marker CD69, the JUNB proto-oncogen. FYB is also seen to be connected to genes such as APC, API2, and CIR. Clearly, these results support the fact that FYB mRNA levels are predictive of the expression level of a number of genes. The hidden state dimensionality was found to be 4, a result similar to the work of [15] in which they developed an iterative empirical Bayesian procedure with a Kalman filter to estimate the posterior distributions of network parameters. [12] found the dimension of the hidden state to be 9 through cross validation, while [14] obtained the value of 14 through a variational Bayesian approach. At a 95% confidence level, we found no significant interactions among the hidden variables or transcription factors. However their role in the transcription process can not be ignored as the inferred matrix Z representing instantaneous protein-RNA transcription was not sparse signifying that the transcription factors regulate the expression level of most mRNAs.

5 CONCLUSION

In this paper, we have developed a state space model with biological replication and applied it to the T-cell data. We used the EM algorithm combined with the bootstrap to infer the structure of the underlying genomic network. The proposed method offers significant advantages over other methods that have recently appeared in the literature. For example, [14] used a variational Bayesian methodology which is an approximation of the posterior distribution of the parameters, whereas we obtained much faster results through direct inference of the parameters. [12] used cross validation as a model selection technique which is quite slow as compared to AIC. [16] used an *ad hoc* method for selecting the hidden state dimensionality k, while our method uses a data-driven approach. Also our model allows for dynamic correlation over time, as each observation and hidden state depend explicitly on some function of previous observations as opposed to the model described by [27, 28, 10]. Their model does not allow for RNA-protein translation and RNA-RNA interactions through the matrix A and B respectively in our model.

One fundamental assumption in our proposed model is the first-order linear dynamics in the state and observation equations of the SSM. This assumption can only be an approximation to the true nature of a complex biological system since more realistic models of gene regulatory interactions surely include complex interactions or nonlinear relationships. Our linear dynamics assumption is a stepping stone upon which a future model with non-linear dynamics will be explored. With application to the t-cell data, we have discovered new interactions that have not

yet been reported in the current literature; as as part of our ongoing work we are investigating these interactions further.

Furthermore the AIC approach is prone to over-fitting, especially in high-dimensional data. A natural way to avoid this over-fitting is through regularization.

Given that Most of existing time-series gene expression data have much fewer replicates, e.g., 3, 2, 1 or no replicates our bound condition (2.4) may be broken down making our system non-identifiable. We can overcome this situation by imposing a penalty on the parameters. In future work, we plan to employ a penalized maximum likelihood strategy in the context of the EM algorithm in the state space model.

COMPETING INTERESTS

Authors have declared that no competing interests exist.

References

[1] Palsson Bernhard. Systems biology simulation of dynamics network states; 2011.

[2] Wen X, Fuhrman S, Michaels S. George, Carr DB, Smith S, Barker JL, Somogyi R. Large-scale temporal gene expression mapping of central nervous system development. Proceedings of the National Academy of Sciences. 1998;95(1):334-339.

[3] Derisi JL, Iyer VR, Brown PO. Exploring the metabolic and genetic control of gene expression on a genomic scale. Science. 1997;278:680.

[4] Wang Z, Gerstein M, Snyder M. Rna-seq: A revolutionary tool for transcriptomics. Nature Reviews Genetics. 2009;10(1):57-63.

[5] Friedman N, Linial M, Nachman I, Pe'er D. Using bayesian networks to analyze expression data. Journal of Computational Biology. 2000;7:601-620.

[6] Friedman J, Hastie T, Tibshirani R. Sparse inverse covariance estimation with the graphical lasso. Biostatistics. 2008;9(3):432-441.

[7] Maucher M, Kracher B, Kühl M, Kestler HA. Inferring boolean network structure via correlation. Bioinformatics. 2011;27(11):1529-1536.

[8] Fahrmeir L, Kunstler R. Penalized likelihood smoothing in robust state space models. Biometrika (1999). 2009;49:173-191.

[9] Fahrmeir L, Wagenpfeil S. Penalized likelihood estimation and iterative kalman smoothing for non-gaussian dynamic regression models. Computational Statistics & Data analysis. 1997;24:295-320.

[10] Fang-Xiang W, Wen-Jun Z, Anthony JK. Modelling gene expression from microarray expression data with state-space equations. Biocomputing. 2004;9:588-592.

[11] Ghahramani Z. Introduction to hidden markov models and bayesian networks. International Journal of Pattern Recognition and Artificial Intelligence. 2001;15(1):9-42.

[12] Rangel C, Angus J, Ghahramani Z, Lioumi M, Sotheran E, Gaiba A, David LW, Falciani F. Modeling t-cell activation using gene expression profiling and state-space models. Bioinformatics. 2004;20(9):1361-1372.

[13] Yamaguchi R, Yoshida R, Imoto S, Higuchi T, Miyano S. Finding module-based gene networks with state-space models. IEEE Signal Processing Magazine. 2007;37.

[14] Beal MJ, Falciani F, Ghahramani Z, Rangel C, Wild DL. A bayesian approach to reconstructing genetic regulatory networks with hidden factors. Bioinformatics. 2005;21:349-356.

[15] Rau A, Foulley JL, Jaffrzic F, Doerge W Rebecca. An empirical bayesian method for estimating biological networks from temporal microarray data. Statistical Applications in Genetics and Molecular Biology. 2010;9(1):2010.

[16] Bremer M, Doerge RW. The km-algorithm identifies regulated genes in time series expression data. Advances in Bioinformatics (in press); 2009.

[17] Hirose O, Yoshida R, Imoto S, Yamaguchi R, Higuchi T, Charnock-Jones DS, Print C,

Miyano S. Statistical inference of transcriptional module-based gene networks from time course gene expression profiles by using state space models. Bioinformatics. 2008;24(7):932-942. Available: http://bioinformatics.oxfordjournals.org/content/24/7/932.full.pdf+html; http://bioinformatics.oxfordjournals.org/content/24/7/932.abstract

[18] Hirose O, Yoshida R, Imoto S, Higuchi T, Miyano S. Analyzing time course gene expression data with biological and technical replicates to estimate gene networks by state space models. 2008;940-946.

[19] Dempster AP, Laird NM, Rubin DB. Maximum likelihood from incomplete data via the em algorithm. Journal of the Royal Statistical Society Series B (Methodological). 1977;39(1)1-38. Available: http://www.jstor.org/stable/2984875

[20] Shumway RH, Stoffer DS. Time series analysis and its applications with r examples. second edition; 2005.

[21] Meinhold RJ, Singpurwalla ND. Understanding the kalman filter. The Americam Statistician. 1983;37(2):123-127.

[22] Efron B. Bootstrap methods: Another look at the jacknife. The Annals of Statistics. 1979;7:1-26.

[23] Ljung L, Caines P. Asymptotic normality of prediction error estimators for approximate systems models. Stochastics. 1979;3:29-46.

[24] Dent W, Min A. A monte carlo study of autoregressive integrated-moving average processes. Journal of Econometrics. 1978;7:23-55.

[25] Brown RG, Hwang PY. Introduction to random signals and applied kalman filtering. John Willey and Sons, New York; 1997.

[26] Dewey TG, Galas DJ. Generalized dynamical models of gene expression and gene classification. Funt Int Genomics. 2000;1:269-278.

[27] Yamaguchi R, Higuchi T. State-space approach with the maximum likelihood principle to identify the system generating time-course gene expression data of yeast. Int J Data Mining and Bioinformatics. 2006;1(1):77-87.

[28] Perrin B, Ralaivola L, Mazurie A, Bottani S, Mallet J, d'Alche Buc F. Gene networks inference using dynamic bayesian networks. Bioinformatics. 2003;19(Suppl2):138-148.

[29] Shumway R, Stoffer D. An approach to time series smoothing and forecasting using the em algorithm. JTime series Analysis. 1982;3:253-264.

[30] Shumway R. Dynamic mixed models for irregularly observed time series. Resenhas-Reviews of the Institute of Mathematics and Statistics, University of Sao Paulo,USP Press, Brazil. 2000;4(4):433-456.

[31] George C, Berger RL. Statistical inference; 1996.

[32] Akaike H. A new look at the statistical model identification. Automatic Control, IEEE Transactions. 1974;19(6):716 -723.

[33] Schwarz G. Estimating the dimension of a model. The Annals of Statistics. 1978;6(2):461-464.

[34] Hurvich CM, Tsai CL. Regression and time series model selection in small samples. Biometrika. 1989;76(2):297-307.

[35] Hurvich CM, Tsai CL. A corrected akaike information criterion for vector autoregressive model selection. Journal of Time Series Analysis. 1993;14(3):271-279.

[36] Burnham KP, Anderson DR. Model selection and multi-model inference. 2002;2.

[37] Wild DL, Rangel C, Angus J, Ghahramani Z. Modeling genetic regulatory networks using gene expression profiling and state space models. Probabilistic Modelling in Bioinformatics and Medical informatics Springer-Verlag (in press); 2004.

APPENDIX

We outline here the derivations of the update equations. We write $\mathbf{Q}(Z,B)$ as

$$\mathbf{Q}(Z,B) = \sum_{r=1}^{n_R} E_{\theta,\varphi^*}\left[l^r_{y_r,\theta_r}(Z,B)\right] \tag{5.1}$$

then

$$\begin{aligned}\mathbf{Q}(Z,B) \quad = \quad & -\sum_{r=1}^{n_R}\sum_{t=1}^{T} y_{tr}^{'} y_{tr} + 2\sum_{r=1}^{n_R}\sum_{t} E(\theta_{tr}^{'} Z y_{tr}) \\ & +2\sum_{r=1}^{n_R}\sum_{t} y_{(t-1)r}^{'} B^{'} y_{tr} - \sum_{r=1}^{n_R}\sum_{t}^{T} ZE(\theta_{tr}^{'}\theta_{tr} Z^{'}) \\ & -2\sum_{r=1}^{n_R}\sum_{t}^{T} E(\theta_{tr}^{'} Z^{'} B y_{(t-1)r}) - \sum_{r=1}^{n_R}\sum_{t}^{T} B^{'} y_{(t-1)r}^{'} y_{(t-1)r} B\end{aligned}$$

Setting $\frac{\partial}{\partial Z}\mathbf{Q}(Z,B)$ and $\frac{\partial}{\partial B}\mathbf{Q}(Z,B)$ equal 0 result in two linear system of equations in the form:

$$\begin{aligned} 0 \quad = \quad & -\frac{1}{2\sigma^2_{\xi_{tr}}}\sum_{r=1}^{n_R}\sum_{t=1}^{T}[-2y_{tr}E(\theta_{tr}^{'}) \\ & +2ZE(\theta_{tr}\theta_{tr}^{'}) + 2By_{(t-1)r}E(\theta_{tr}^{'})]\end{aligned} \tag{5.2}$$

and

$$\begin{aligned} 0 = \quad - \quad & \frac{1}{2\sigma^2_{\xi_{tr}}}\sum_{r=1}^{n_R}\sum_{t=1}^{T}[-2y_{(t-1)r}y_{tr}^{'} + 2y_{(t-1)r}E(\theta_{tr}^{'})Z^{'} \\ & +2(y_{(t-1)r}y_{(t-1)r}^{'}B^{'})]\end{aligned} \tag{5.3}$$

Equations 5.2 and 5.3 could also be re-written as

$$-M_{y\theta} + ZM_{\theta\theta} + BM_{L(y)\theta} = 0 \tag{5.4}$$

$$-M_{L(y)y} + M_{L(y)\theta}Z^{'} + M_{L(y)L(y)}B^{'} = 0 \tag{5.5}$$

where

$$M_{y\theta} = \sum_{rt} y_{tr}E(\theta_{tr}^{'}), \quad M_{\theta\theta} = \sum_{rt} E(\theta_{tr}\theta_{tr}^{'}), \quad M_{L(y)L(y)} = \sum_{rt} y_{(t-1)r}y_{(t-1)r}^{'}$$

$$M_{L(y)\theta} = \sum_{rt} y_{(t-1)r}E(\theta_{tr}^{'}), \quad M_{L(y)y} = \sum_{rt} y_{(t-1)r}y_{tr}^{'}$$

and $L(y)$ in Equations 5.4 and 5.5 is the shift operator on matrix y.
From Equation 5.4,

$$Z = M_{y\theta}M_{\theta\theta}^{-1} - BM_{L(y)\theta}M_{\theta\theta}^{-1} \tag{5.6}$$

Substitute Equation 5.6 into Equation 5.5, gives

$$BM_{L(y)L(y)} = M_{yL(y)} - M_{y\theta}M_{\theta\theta}^{-1}M_{\theta L(y)} + BM_{L(y)\theta}M_{\theta\theta}^{-1}M_{\theta L(y)} \tag{5.7}$$

Therefore

$$B = \left[M_{yL(y)} - M_{y\theta}M_{\theta\theta}^{-1}M_{\theta L(y)}\right]\left[M_{L(y)L(y)} - M_{L(y)\theta}M_{\theta\theta}^{-1}M_{\theta L(y)}\right]^{-1} \tag{5.8}$$

Also, from Equation 5.5

$$B = M_{yL(y)}M_{yL(y)}^{-1} - ZM_{\theta L(y)}M_{yL(y)}^{-1} \tag{5.9}$$

Substitute Equation 5.9 into Equation 5.4, gives

$$ZM_{\theta\theta} = M_{y\theta} - M_{yL(y)}M_{L(y)L(y)}^{-1}M_{L(y)\theta} + ZM_{\theta L(y)}M_{L(y)L(y)}^{-1}M_{L(y)\theta} \tag{5.10}$$

Rearranging Equation 5.10, we have

$$Z = \left[M_{y\theta} - M_{yL(y)}M_{L(y)L(y)}^{-1}M_{L(y)\theta}\right]\left[M_{\theta\theta} - M_{\theta L(y)}M_{L(y)L(y)}^{-1}M_{L(y)\theta}\right]^{-1} \tag{5.11}$$

Equations 5.8 and 5.11 are the update equations in the maximization step used to infer the parameters in the observation dynamics.

In the same manner we derive the updates equations for A and F for the model interaction parameters in the state dynamics model.

Peer-review history:
The peer review history for this paper can be accessed here:
http://sciencedomain.org/review-history/16311