

RESEARCH

Open Access



Intrinsically Bayesian robust classifier for single-cell gene expression trajectories in gene regulatory networks

Alireza Karbalayghareh^{1*}, Ulisses Braga-Neto^{1,2} and Edward R. Dougherty^{1,2}

From The Fourth International Workshop on Computational Network Biology: Modeling, Analysis, and Control (CNB-MAC 2017) Boston, MA, USA. 20 August 2017

Abstract

Background: Expression-based phenotype classification using either microarray or RNA-Seq measurements suffers from a lack of specificity because pathway timing is not revealed and expressions are averaged across groups of cells. This paper studies expression-based classification under the assumption that single-cell measurements are sampled at a sufficient rate to detect regulatory timing. Thus, observations are expression trajectories. In effect, classification is performed on data generated by an underlying gene regulatory network.

Results: Network regulation is modeled via a Boolean network with perturbation, regulation not fully determined owing to inherent biological randomness. The binary assumption is not critical because the resulting Markov chain characterizes expression trajectories. We assume a partially known Gaussian observation model belonging to an uncertainty class of models. We derive the intrinsically Bayesian robust classifier to discriminate between wild-type and mutated networks based on expression trajectories. The classifier minimizes the expected error across the uncertainty class relative to the prior distribution. We test it using a mammalian cell-cycle model, discriminating between the normal network and one in which gene p27 is mutated, thereby producing a cancerous phenotype. Tests examine all model aspects, including trajectory length, perturbation probability, and the hyperparameters governing the prior distribution over the uncertainty class.

Conclusions: Simulations show the rates at which the expected error is diminished by smaller perturbation probability, longer trajectories, and hyperparameters that tighten the prior distribution relative to the unknown true network. For average-expression measurement, methods have been proposed to obtain prior distributions. These should be extended to the more mathematically difficult, but more informative, expression trajectories.

Keywords: Intrinsically Bayesian robust classifier, Optimal Bayesian classifier, Bayesian trajectory classifier, Single-cell expression trajectory, Gene regulatory network, Boolean network, Bayesian partially observed Boolean dynamical system

Background

Phenotype classification is a salient issue for translational genomics, for instance, classification of normal versus cancerous cells, of different stages of tumor development, or different prospective drug response. Both expression

microarray and RNA-Seq measurements have received much interest; however, they suffer from a lack of specificity. First, pathway timing is not reflected in the data and, second, expressions are averaged across groups of cells, so that individual cell responses are not detectable. New technologies are being developed for profiling individual cells using RNA-Seq or quantitative PCR [1]. Individual cells can be captured via standard methods, such as flow cytometry, glass capillaries, or laser [2], and be measured at various time points.

*Correspondence: alireza.kg@tamu.edu

¹Department of Electrical and Computer Engineering, Texas A&M University, 77843 College Station, TX, USA

Full list of author information is available at the end of the article

Genes have interactions with each other, which can determine how they are behaving over time and define the dynamics of gene regulatory networks (GRNs). One way of showing the dynamics of GRNs over discrete time points is Boolean networks with perturbation (BNp) [3]. A BNp is a Markov chain, in which the state of a gene (0 for off and 1 for on) at the current time is a function of the states of its predictor genes at the previous time plus a small random Boolean noise. Network inference could be done using RNA-Seq time series measurements [4].

Suppose we have single-cell measurements sampled with a sufficient rate to detect regulatory timing. In effect, this would mean that classification would be done on data reflecting an underlying gene regulatory network. In [5] we proposed a classifier to classify the state trajectories of the two classes: wild-type and mutated, each having its own BNp. We derived the Bayes classifier and computed the Bayes error. We analyzed the effects of the length of the trajectories, perturbation probability, and different mutations on the Bayes error.

In [6] and [7] we assumed an observation model on the state dynamics of the BNps, from which the expression values of the genes are obtained. As the parameters of the model were all unknown and the network functions were partially known, we proposed an expectation-maximization (EM)-based algorithm to estimate these parameters and functions, and then plugged them into the Bayes classifier. In [6] we assumed that the expression trajectory data come from single-cell measurements and compared that with a multiple-cell scenario, in which instead of trajectories we have the averaged expressions of all genes over all cells, which translates to an average over all states.

In this paper we extend the single-cell trajectory classification to the Bayesian framework. We propose the intrinsically Bayesian robust (IBR) classifier for the trajectories. The IBR classifier is a specific type of the optimal Bayesian classifier (OBC), first introduced in [8, 9] for the classification of static data. In fact, the difference between the OBC and IBR classifiers is that in the OBC the expectation of the class-conditional densities is taken over the posteriors of the parameters to obtain the effective class-conditional densities, whereas in the IBR classifier the expectation is taken over the priors. The IBR/OBC concept has been applied to linear and morphological filtering [10, 11], and IBR Kalman filtering [12]. Regarding the prior distributions, prior construction methods using the pathway knowledge have been studied in the literature, such as [13, 14]. Bayesian methods for finding differentially expressed genes can be used for the aim of classification [15].

Here we apply the IBR classifier to the classification of trajectories. As opposed to [6, 7], where we estimate the parameters, here we assume that the parameters belong to an uncertainty class governed by a prior distribution. We

assume that there are two classes: wild-type ($S = 0$) and mutated ($S = 1$). We introduce a Bayesian version of the partially observed Boolean dynamical system (POBDS), proposed in [16], as the observation model. We use a beta prior distribution for the prior probability of the class $S = 0$ and also for the gene perturbation probability. Since the observation model given the states is Gaussian, we employ the normal-gamma distribution as the prior distribution of the mean and precision (inverse of variance) of the Gaussian model.

In the simulation part, we employ a mammalian cell-cycle gene regulatory network [17] consisting of 10 genes as the BNp for the class $S = 0$ and its mutated version as the BNp for the class $S = 1$. We analyze the effects of all the hyperparameters of the model and the length of the observed trajectories on the classification error. The proposed classifier is computationally efficient because we use the sum-product method to reduce the complexity to $m \times 2^n$, where m is the length of the observed trajectory and n is the number of genes in the network. Being linearly dependent on time points, m , makes the classifier very fast even for longer trajectories.

Methods

In a Boolean network (BN) for n genes, each gene value $x_i \in \{0, 1\}$, for $i = 1, \dots, n$, at time $k + 1$ is determined by the values of some predictor genes at time k via a Boolean function $f_i : \{0, 1\}^n \rightarrow \{0, 1\}$. In practice, f_i is a function of a small number of genes, K_i , called the *in-degree* of the gene x_i in the network. The in-degree of the network is $K = \max_{i=1, \dots, n} K_i$. A gene network can be represented as a graph with vertices representing genes and edges representing regulations. There is a state diagram of 2^n states corresponding to the truth table of the BN, representing the dynamics of the network. Given an initial state, a BN will eventually reach a set of states, called an *attractor cycle*, through which it will cycle endlessly. Each initial state corresponds to a unique attractor cycle, and the set of initial states leading to a specific attractor cycle is known as the *basin of attraction* (BOA) of the attractor cycle.

State model

We allow stochasticity in our state model by using Boolean networks with perturbation (BNps) instead of deterministic BNs. For BNps, perturbation is introduced with a probability p_k by which the state of a gene in the network can be randomly flipped at time k . We assume that there is an independent identically distributed (i.i.d.) random perturbation vector at each time k , denoted by $\mathbf{n}_k \in \{0, 1\}^n$, such that the i -th gene flips at time k if the i -th component of \mathbf{n}_k is equal to 1. Therefore, the dynamical model can be expressed as

$$\mathbf{X}_{k+1} = \mathbf{f}(\mathbf{X}_k) \oplus \mathbf{n}_{k+1}, \quad k = 0, 1, 2, \dots, \quad (1)$$

where $\mathbf{X}_k = [x_1(k), x_2(k), \dots, x_n(k)]^T$ is a binary state vector, called a *gene activity profile* (GAP), at time k , in which $x_i(k)$ indicates the expression level of the i -th gene at time k (either 0 or 1); $\mathbf{f} = [f_1, f_2, \dots, f_n]^T : \{0, 1\}^n \rightarrow \{0, 1\}^n$ is the vector of the network functions, in which f_i shows the expression level of the i -th gene at time $k + 1$ when the system lies in the state \mathbf{X}_k at time k ; $\mathbf{n}_k = [n_1(k), n_2(k), \dots, n_n(k)]^T$ is the perturbation vector at time k , in which $n_1(k), n_2(k), \dots, n_n(k)$ are i.i.d. Bernoulli random variables for every k with the parameter $p_k = P(n_i(k) = 1)$ for $i = 1, \dots, n$; and \oplus is component-wise modulo 2 addition.

The existence of perturbation makes the corresponding Markov chain of a BNp irreducible. Hence, the network possesses a steady-state distribution π describing its long-run behavior. If p_k is sufficiently small, π will reflect the attractor structure within the original network. We can derive the transition probability matrix (TPM) if we know the truth table and the perturbation probability of a BNp. As a result, the steady-state distribution π can be computed as well.

Prior for state parameter

In this paper, we assume that we know the underlying Boolean networks for both the wild-type and mutated classes, and the only uncertain parameter at the state level is the perturbation probability p_k . Since $0 < p_k < 1$, we can employ the beta prior for p_k , for all $k = 1, 2, \dots$, as

$$g(p_k) \sim \text{Beta}(a, b) = \frac{\Gamma(a + b)}{\Gamma(a)\Gamma(b)} p_k^{a-1} (1 - p_k)^{b-1}, \quad (2)$$

where a and b are known parameters. Since in reality p_k is close to zero, we can choose a and b in such a way that this fact is satisfied. To this end, we can use the mean and variance of p_k :

$$E[p_k] = \frac{a}{a + b}, \quad \text{Var}[p_k] = \frac{ab}{(a + b)^2(a + b + 1)}. \quad (3)$$

Observation model

We define a Bayesian partially-observed Boolean dynamical system (BPOBDS) as the model for the gene expression data. In this model, we assume that the gene expressions come from Gaussian distributions whose parameters are governed by prior distributions whose parameters (the hyperparameters of the observations) are a function of the hidden Boolean states. If $y_j(k)$ is the expression value of the j -th gene at time k , then the observation model is

$$p(y_j(k) | \theta_j(k), \lambda_j(k)) \sim \mathcal{N}(\theta_j(k), \lambda_j(k)^{-1}), \quad (4)$$

for $j = 1, 2, \dots, n$ and $k = 1, 2, \dots$, where $\theta_j(k)$ and $\lambda_j(k)$ denote the mean and precision, respectively, of the Gaussian distribution.

Priors for observation parameters

We employ the well-known normal-gamma prior distribution for $\theta_j(k)$ and $\lambda_j(k)$:

$$\begin{aligned} p(\lambda_j(k)) &\sim \text{Gamma}(\alpha_0, \beta_0), \\ p(\theta_j(k) | \lambda_j(k), x_j(k)) &\sim \mathcal{N}(\mu_j(k), (\kappa_0 \lambda_j(k))^{-1}), \\ \text{where } \mu_j(k) &= \mu_0 + \delta_0 x_j(k), \end{aligned} \quad (5)$$

where $\alpha_0, \beta_0, \kappa_0, \mu_0$, and δ_0 are known positive hyperparameters, and $x_j(k)$ is the hidden Boolean state of gene j at time k . The intuition behind the prior (5) is that when gene j at time k is on or off, that is, $x_j(k) = 1$ or 0, the hypermean of the expression for that gene is $\mu_j(k) = \mu_0 + \delta_0$ or $\mu_j(k) = \mu_0$, respectively, at time k . In (5), μ_0 is the baseline expression level and δ_0 is the expression coefficient. The hyperparameters α_0, β_0 , and κ_0 determine the level of uncertainty, by which we can control the variance of the outputs. We assume the same values of hyperparameters for all genes at all times.

IBR classifier

If one knows the feature-label distribution, then the error of any classifier can be found and an optimal (Bayes) classifier minimizes classifier error. If the feature-label distribution is unknown but belongs to an uncertainty class Θ of feature-label distributions, then we desire a classifier to minimize the expected error over the uncertainty class. Given a classifier ψ , from the perspective of mean-square error (MSE), the best error estimate minimizes the MSE between its true error (a function of parameter θ) and an error estimate. This Bayesian minimum-mean-square-error (MMSE) estimate is given by the expected true error, $\hat{\varepsilon}(\psi) = E_\theta[\varepsilon(\psi, \theta)]$, where $\varepsilon(\psi, \theta)$ is the error of ψ on the feature-label distribution parameterized by θ and the expectation is taken relative to the prior distribution $\pi(\theta)$ [18].

An IBR classifier minimizes the Bayesian MMSE estimate. If $\psi(\mathbf{x}) = 0$ if $\mathbf{x} \in R_0$ and $\psi(\mathbf{x}) = 1$ if $\mathbf{x} \in R_1$, where \mathbf{x} is a multidimensional vector of data, and R_0 and R_1 partition the sample space, then [8]

$$\hat{\varepsilon}(\psi) = E_\pi[c] \int_{R_1} f_\Theta(\mathbf{x}|0) d\mathbf{x} + (1 - E_\pi[c]) \int_{R_0} f_\Theta(\mathbf{x}|1) d\mathbf{x},$$

where

$$f_\Theta(\mathbf{x}|y) = \int_{\Theta_y} f_{\theta_y}(\mathbf{x}|y) \pi(\theta_y) d\theta_y$$

is the *effective class-conditional density* for class y , Θ_y being the space for $\theta_y, f_{\theta_y}(\mathbf{x}|y)$ is the class-conditional density, and c is the prior probability of the class 0. The IBR classifier is given by [8]

$$\psi_{\text{IBR}}(\mathbf{x}) = \begin{cases} 0 & \text{if } E_\pi[c] f_\Theta(\mathbf{x}|0) \geq \\ & (1 - E_\pi[c]) f_\Theta(\mathbf{x}|1) \\ 1 & \text{otherwise} \end{cases} \quad (6)$$

Trajectory-based IBR classifier

Let $\Theta_s = [p_{2:m}, \theta_{1:n}(1 : m), \lambda_{1:n}(1 : m)]$ denote the parameters of the class $S = s$, for $s = 0, 1$, where $p_{2:m}$ means the parameters p_2, p_3, \dots, p_m , and similarly for $\theta_{1:n}(1 : m)$ and $\lambda_{1:n}(1 : m)$. Furthermore, let \mathbf{f}_s denote the Boolean network function of the class $S = s$ and $\mathcal{X} = [\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_m]$ denote the Boolean state trajectory at m consecutive times points at which \mathcal{Y} has been observed. The $n \times 1$ Boolean vector $\mathbf{X}_k = [x_1(k), x_2(k), \dots, x_n(k)]^T$ has the states of the n genes at time k , which are hidden and not observed.

Suppose that we obtain the expressions of n genes at m consecutive time points. Let $\mathbf{Y}_k = [y_1(k), \dots, y_n(k)]^T$ denote the $n \times 1$ expression vector of n genes at time k , and $\mathcal{Y} = [\mathbf{Y}_1, \dots, \mathbf{Y}_m]$ denote a time trajectory of length m , containing the expression vectors at the m consecutive times. The problem is to optimally classify this observed trajectory \mathcal{Y} to the class 0 (wild-type) or class 1 (mutated). Let c and $1 - c$ be the prior probabilities of the class $S = 0$ and class $S = 1$, respectively. Since we are uncertain about c , we use a beta prior

$$g(c) \sim \text{Beta}(a_c, b_c) = \frac{\Gamma(a_c + b_c)}{\Gamma(a_c)\Gamma(b_c)} c^{a_c-1} (1-c)^{b_c-1}, \quad (7)$$

with mean

$$E[c] = \frac{a_c}{a_c + b_c}, \quad (8)$$

where a_c and b_c are known parameters.

According to (6), the IBR classifier for the trajectories is

$$\psi_{\text{IBR}}(\mathcal{Y}) = \begin{cases} 0 & \text{if } E[c]p(\mathcal{Y}|S=0) \geq \\ & (1 - E[c])p(\mathcal{Y}|S=1) \\ 1 & \text{otherwise} \end{cases}, \quad (9)$$

where $p(\mathcal{Y}|S = s)$ is the effective class-conditional density of the trajectory \mathcal{Y} in the class $S = s$ for $s = 0, 1$.

Effective class-conditional densities of trajectories

The joint distribution of \mathcal{Y}, \mathcal{X} , and Θ_s given the class $S = s$ can be factorized as

$$p(\mathcal{Y}, \mathcal{X}, \Theta_s | S = s) = g(p_{2:m})P(\mathcal{X} | p_{2:m}, S = s) \times p(\mathcal{Y} | \theta_{1:n}(1 : m), \lambda_{1:n}(1 : m)) \times p(\theta_{1:n}(1 : m), \lambda_{1:n}(1 : m) | \mathcal{X}). \quad (10)$$

In deriving (10), it is assumed that the state parameters $\{p_{2:m}\}$ are independent of the observation parameters

$\{\theta_{1:n}(1 : m), \lambda_{1:n}(1 : m)\}$. Due to the independence assumption in the priors,

$$g(p_{2:m}) = \prod_{k=1}^{m-1} g(p_{k+1}) = \prod_{k=1}^{m-1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} p_{k+1}^{a-1} (1-p_{k+1})^{b-1}, \quad (11)$$

$$p(\theta_{1:n}(1 : m), \lambda_{1:n}(1 : m) | \mathcal{X}) = \prod_{k=1}^m \prod_{j=1}^n p(\theta_j(k) | \lambda_j(k), x_j(k)) p(\lambda_j(k)) = \prod_{k=1}^m \prod_{j=1}^n \frac{1}{\sqrt{2\pi(\kappa_0 \lambda_j(k))^{-1}}} \frac{\beta_0^{\alpha_0}}{\Gamma(\alpha_0)} \times \exp\left(-\frac{(\theta_j(k) - \mu_j(k))^2}{2(\kappa_0 \lambda_j(k))^{-1}}\right) \lambda_j(k)^{\alpha_0-1} \exp(-\beta_0 \lambda_j(k)). \quad (12)$$

If we assume that the conditional expressions, given the parameters, of the n genes at m time points are independent, the likelihood of \mathcal{Y} given the parameters in (10) can be written as

$$p(\mathcal{Y} | \theta_{1:n}(1 : m), \lambda_{1:n}(1 : m)) = \prod_{k=1}^m \prod_{j=1}^n p(y_j(k) | \theta_j(k), \lambda_j(k)) = \prod_{k=1}^m \prod_{j=1}^n \frac{1}{\sqrt{2\pi \lambda_j(k)^{-1}}} \exp\left(-\frac{(y_j(k) - \theta_j(k))^2}{2\lambda_j(k)^{-1}}\right). \quad (13)$$

We should note that the independency assumption in (13) is only for the observations, whereas the genes have interactions at the state level, following the underlying Boolean network, so that they cannot be considered independent. Due to the Markov property in (1), $p(\mathcal{X} | p_{2:m}, S = s)$ in (10) can be factored as

$$P(\mathcal{X} | p_{2:m}, S = s) = P(\mathbf{X}_1 | S = s) \prod_{k=1}^{m-1} P(\mathbf{X}_{k+1} | \mathbf{X}_k, p_{k+1}, S = s), \quad (14)$$

where $P(\mathbf{X}_{k+1} | \mathbf{X}_k, p_{k+1}, S = s)$ is the probability of transitioning from state \mathbf{X}_k at time k to state \mathbf{X}_{k+1} at time $k + 1$, given the perturbation probability p_{k+1} , in the class $S = s$, and $P(\mathbf{X}_1 | S = s)$ is the probability of the first state \mathbf{X}_1 in the class $S = s$. Let \mathbf{x}^i denote the $n \times 1$ Boolean vector of the state i , for $i = 1, 2, \dots, 2^n$. Given the perturbation probability p_{k+1} , the conditional transition

probability matrix (TPM) at time $k + 1$, which is a $2^n \times 2^n$ matrix, in the class $S = s$ can be derived from (1) as

$$\begin{aligned} \mathbf{A}_{ij}^{(s)}(k + 1) &= P(\mathbf{X}_{k+1} = \mathbf{x}^j | \mathbf{X}_k = \mathbf{x}^i, p_{k+1}, S = s) \\ &= p_{k+1}^{\mathbf{d}(\mathbf{x}^j, \mathbf{f}_s(\mathbf{x}^i))} (1 - p_{k+1})^{n - \mathbf{d}(\mathbf{x}^j, \mathbf{f}_s(\mathbf{x}^i))}, \end{aligned} \tag{15}$$

where $\mathbf{d}(\mathbf{x}^j, \mathbf{f}_s(\mathbf{x}^i))$ is the Hamming distance between the two Boolean vectors \mathbf{x}^j and $\mathbf{f}_s(\mathbf{x}^i)$.

For obtaining $p(\mathcal{Y} | S = s)$, we need to integrate out the joint distribution $p(\mathcal{Y}, \mathcal{X}, \Theta_s | S = s)$ in (10) with respect to Θ_s and \mathcal{X} :

$$\begin{aligned} p(\mathcal{Y} | S = s) &= \sum_{\mathcal{X}} \int_{\Theta_s} p(\mathcal{Y}, \mathcal{X}, \Theta_s | S = s) \\ &= \sum_{\mathcal{X}} \left\{ P(\mathbf{X}_1 | S = s) \prod_{k=1}^{m-1} \int_{p_{k+1}} g(p_{k+1}) p_{k+1}^{\mathbf{d}(\mathbf{X}_{k+1}, \mathbf{f}_s(\mathbf{X}_k))} \right. \\ &\quad \times (1 - p_{k+1})^{n - \mathbf{d}(\mathbf{X}_{k+1}, \mathbf{f}_s(\mathbf{X}_k))} dp_{k+1} \\ &\quad \times \prod_{k=1}^m \prod_{j=1}^n \int_{\theta_j(k)} \int_{\lambda_j(k)} p(y_j(k) | \theta_j(k), \lambda_j(k)) \\ &\quad \left. \times p(\theta_j(k) | \lambda_j(k), x_j(k)) p(\lambda_j(k)) d\theta_j(k) d\lambda_j(k) \right\}. \end{aligned} \tag{16}$$

Fortunately, as the priors are conjugate, we can analytically solve the integrals in (16). We will use the following lemmas to do so.

Lemma 1 Let $\mathbf{P} = (0 \ 1)$ be the domain of p_{k+1} . The following equation holds:

$$\begin{aligned} K_1 &\triangleq \int_{\mathbf{P}} g(p_{k+1}) p_{k+1}^{\mathbf{d}(\mathbf{X}_{k+1}, \mathbf{f}_s(\mathbf{X}_k))} \\ &\quad \times (1 - p_{k+1})^{n - \mathbf{d}(\mathbf{X}_{k+1}, \mathbf{f}_s(\mathbf{X}_k))} dp_{k+1} \\ &= \frac{\Gamma(\mathbf{d}(\mathbf{X}_{k+1}, \mathbf{f}_s(\mathbf{X}_k)) + a) \Gamma(n - \mathbf{d}(\mathbf{X}_{k+1}, \mathbf{f}_s(\mathbf{X}_k)) + b)}{\Gamma(a) \Gamma(b) \Gamma(a + b + n) \Gamma(a + b)^{-1}}. \end{aligned} \tag{17}$$

Proof See Appendix 1 in Additional file 1. \square

We assume that the observations \mathcal{Y} occur in the steady state. As a result, in (16), $P(\mathbf{X}_1 | S = s)$ is the steady-state probability of the state \mathbf{X}_1 in the class $S = s$, for $s = 0, 1$. The following lemma gives the steady-state distribution.

Lemma 2 Let $\pi_i^{(s)} = P(\mathbf{X}_1 = \mathbf{x}^i | S = s)$ denote the steady-state probability of the i -th state in the class $S = s$, and $\pi^{(s)} = [\pi_1^{(s)}, \dots, \pi_{2^n}^{(s)}]$ be the 1×2^n vector of the steady-state distribution. Then $\pi^{(s)}$ can be calculated from

$$\pi^{(s)} = \pi^{(s)} \mathbf{M}^{(s)}, \quad \sum_{i=1}^{2^n} \pi_i^{(s)} = 1, \tag{18}$$

where $\mathbf{M}^{(s)}$ is the transition probability matrix of the class $S = s$ with the entries

$$\mathbf{M}_{ij}^{(s)} = \frac{\Gamma(\mathbf{d}(\mathbf{x}^j, \mathbf{f}_s(\mathbf{x}^i)) + a) \Gamma(n - \mathbf{d}(\mathbf{x}^j, \mathbf{f}_s(\mathbf{x}^i)) + b)}{\Gamma(a) \Gamma(b) \Gamma(a + b + n) \Gamma(a + b)^{-1}}. \tag{19}$$

Proof See Appendix 2 in Additional file 1. \square

Lemma 3 Let $\Omega = (-\infty \ \infty)$ and $\Lambda = (0 \ \infty)$ be the domains of $\theta_j(k)$ and $\lambda_j(k)$, respectively, for $j = 1, \dots, n$, and $k = 1, \dots, m$. The following equation holds:

$$\begin{aligned} K_2 &\triangleq \int_{\Omega} \int_{\Lambda} p(y_j(k) | \theta_j(k), \lambda_j(k)) \\ &\quad \times p(\theta_j(k) | \lambda_j(k), x_j(k)) p(\lambda_j(k)) d\theta_j(k) d\lambda_j(k) \\ &= \frac{1}{(2\pi)^{\frac{1}{2}}} \left(\frac{\kappa_0}{\kappa_1} \right)^{\frac{1}{2}} \frac{\Gamma(\alpha_1) \beta_0^{\alpha_0}}{\Gamma(\alpha_0) \beta_1^{\alpha_1}}, \end{aligned} \tag{20}$$

where

$$\begin{aligned} \kappa_1 &= \kappa_0 + 1, \\ \alpha_1 &= \alpha_0 + \frac{1}{2}, \\ \beta_1 &= \beta_0 + \frac{\kappa_0 (y_j(k) - \mu_0 - \delta_0 x_j(k))^2}{2(\kappa_0 + 1)}. \end{aligned} \tag{21}$$

Proof See Appendix 3 in Additional file 1. \square

Summing out \mathcal{X}

Using (16), (17), (19), and (20), we have

$$\begin{aligned} p(\mathcal{Y} | S = s) &= \sum_{\mathbf{X}_1} \dots \sum_{\mathbf{X}_m} \left\{ \pi_{\mathbf{X}_1}^{(s)} \prod_{k=1}^{m-1} \frac{\Gamma(\mathbf{d}(\mathbf{X}_{k+1}, \mathbf{f}_s(\mathbf{X}_k)) + a) \Gamma(n - \mathbf{d}(\mathbf{X}_{k+1}, \mathbf{f}_s(\mathbf{X}_k)) + b)}{\Gamma(a) \Gamma(b) \Gamma(a + b + n) \Gamma(a + b)^{-1}} \right. \\ &\quad \left. \times \prod_{k=1}^m \prod_{j=1}^n \frac{1}{(2\pi)^{\frac{1}{2}}} \left(\frac{\kappa_0}{\kappa_1} \right)^{\frac{1}{2}} \frac{\Gamma(\alpha_1) \beta_0^{\alpha_0}}{\Gamma(\alpha_0) \beta_1^{\alpha_1}} \right\}. \end{aligned} \tag{22}$$

We use the sum-product algorithm [19] to efficiently compute the summation in (22). Define the $2^n \times 1$ vector $\Phi(k)$ by

$$\begin{aligned} \Phi_i(k) &= \left(\frac{\beta_0^{\alpha_0}}{(2\pi)^{\frac{1}{2}}} \left(\frac{\kappa_0}{\kappa_0 + 1} \right)^{\frac{1}{2}} \frac{\Gamma(\alpha_0 + \frac{1}{2})}{\Gamma(\alpha_0)} \right)^n \\ &\quad \times \prod_{j=1}^n \left[\beta_0 + \frac{\kappa_0 (y_j(k) - \mu_0 - \delta_0 x_j^i)^2}{2(\kappa_0 + 1)} \right]^{-(\alpha_0 + \frac{1}{2})}, \end{aligned} \tag{23}$$

for $i = 1, \dots, 2^n$, where x_j^i is the j -th entry of the Boolean state \mathbf{x}^i . We define an auxiliary $2^n \times 1$ vector $\Pi^{(s)}(k)$ at the

time k , for $k = 1, \dots, m$, which is initialized and updated as follows:

$$\begin{aligned} \Pi^{(s)}(1) &= \left(\pi^{(s)}\right)^T \circ \Phi(1), \\ \Pi^{(s)}(k+1) &= \left[\mathbf{M}^{(s)T} \Pi^{(s)}(k)\right] \circ \Phi(k+1), \end{aligned} \quad (24)$$

for $k = 1, \dots, m - 1$, where T denotes the transpose, and \circ is the Hadamard product. Once we have calculated $\Pi^{(s)}(m)$, the summation of (22) is equal to the l_1 norm $\|\Pi^{(s)}(m)\|_1$, which is the summation of the all 2^n entries of $\Pi^{(s)}(m)$. Therefore, (22) can be written as

$$p(\mathcal{Y}|S = s) = \|\Pi^{(s)}(m)\|_1. \quad (25)$$

Results and discussion

In this section, we consider a mammalian cell-cycle gene regulatory network [17], depicted in Fig. 1, for evaluating our proposed trajectory-based IBR classifier. This GRN consists of $n = 10$ genes, whose interactions are shown in Fig. 1. The Boolean functions of the corresponding Boolean network for this GRN are given in Table 1 [17]. We define class $S = 0$ as the wild-type class, whose network function \mathbf{f}_0 is in Table 1. The value of the gene CycD is determined by extracellular signals, and we assume it is $f_1 = 0$. According to [17], one mutated case which leads to cancer is when the gene p27 in the network is shut down and cannot be activated by its regulating genes, that is, $f_3 = 0$. Therefore, we define class $S = 1$ as the mutated class with the network function \mathbf{f}_1 , which is the same as Table 1 with $f_3 = 0$. We analyze the effects of the hyperparameters and m on the classification error in Figs. 2, 3, 4, 5, 6, 7, 8 and 9. In the simulations, we have set $a_c = b_c = 10$

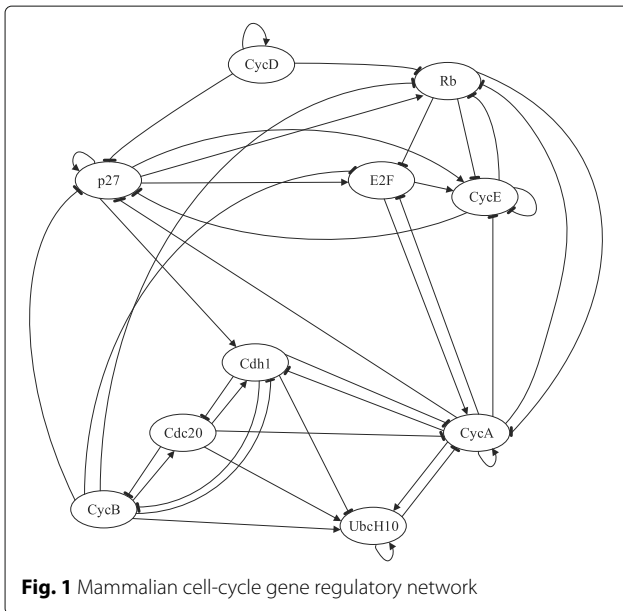


Fig. 1 Mammalian cell-cycle gene regulatory network

and $\mu_0 = 10$. Therefore, the prior probability c of the class $S = 0$ will have the mean value $E[c] = 0.5$.

Figure 2 shows the classification error versus m for $a = 1, 3, 5$, when the values of the other hyperparameters are $b = 100, \alpha_0 = 100, \beta_0 = 10^4, \delta_0 = 40$, and $\kappa_0 = 100$. We see that the classification error decreases by increasing m and converges to zero. This means that for long enough trajectories we can have perfect classification. The value of the hyperparameter a determines the amount of uncertainty for the gene perturbation probability p_k at time k . From a biological perspective, we know that p_k should be small. As a result, we have chosen $b = 100$, and $a = 1, 3, 5$, leading to the mean values of p_k , respectively, as $E[p_k] = \frac{a}{a+b} \approx 0.01, 0.03, 0.05$. For a given b , the bigger value of a allows a wider range of p_k , which results in higher classification error.

Figure 3 represents the classification error versus m for different values of α_0 and β_0 , when $a = 1, b = 100, \delta_0 = 40$, and $\kappa = 100$. As the precision (inverse of variance) has a Gamma(α_0, β_0) distribution, its mean and variance are equal to $E[\lambda] = \frac{\alpha_0}{\beta_0}$ and $\text{Var}[\lambda] = \frac{\alpha_0}{\beta_0^2} = \frac{E[\lambda]}{\beta_0}$. Figure 3 shows the error curves for the two cases $\frac{\alpha_0}{\beta_0} = 10^{-3}$ and 2×10^{-3} , each having a different value of β_0 , leading to a different variance of λ . As such, we can see the effects of both the mean and variance of the precision λ . Whenever $\frac{\alpha_0}{\beta_0}$ increases, there is lower variance in the outputs, which results in lower errors. We also notice from Fig. 3 that for a given value of $\frac{\alpha_0}{\beta_0}$, increasing β_0 decreases the error, the reason being that increased β_0 yields lower variance for the precision.

Figure 4 plots the error versus α_0 for $a = 1, 3, 5$, when $m = 5, b = 100, \delta_0 = 40, \kappa_0 = 100$, and $\beta_0 = 10^5$. For all values of a , the classification error is a decreasing function of α_0 . Similarly, Fig. 5 gives the error versus β_0 for $a = 1, 3, 5$, when $\alpha_0 = 10$ and the other hyperparameters are the same as in Fig. 4. Figure 5 shows an increasing trend of classification error as β_0 grows.

Figure 6 shows the error as a function of a , when $m = 5, b = 100, \delta_0 = 40, \kappa_0 = 100, \alpha_0 = 100$, and $\beta_0 = 10^5$. When a grows, the uncertainty of the perturbation probability grows as well. The error is an increasing function of a . Similarly, Fig. 7 is for error versus b , when $a = 1$, and the others are the same as in Fig. 6. In Fig. 7 the classification error is decreasing as b increases.

Figure 8 illustrates the error versus κ_0 , for $m = 5, a = 1, b = 100, \delta_0 = 40, \alpha_0 = 100$, and $\beta_0 = 10^5$. From (5), the hyperparameter κ_0 controls the variance of the mean parameters $\theta_j(k)$. When κ_0 increases, the uncertainty of $\theta_j(k)$ is reduced and its density peaks at μ_0 and $\mu_0 + \delta_0$ for the unexpressed and expressed states, respectively. Consequently, we expect better error rates for higher κ_0 . Accordingly, in Fig. 8 the classification error is a decreasing function of κ_0 .

Table 1 Definitions of Boolean functions for the wild-type mammalian cell-cycle BN with 10 genes

Order	Gene	Regulating function
x_1	<i>CycD</i>	$f_1 = \text{Extracellular signals}$
x_2	<i>Rb</i>	$f_2 = (\overline{CycD} \wedge \overline{CycE} \wedge \overline{CycA} \wedge \overline{CycB}) \vee (p27 \wedge \overline{CycD} \wedge \overline{CycB})$
x_3	<i>p27</i>	$f_3 = (\overline{CycD} \wedge \overline{CycE} \wedge \overline{CycA} \wedge \overline{CycB}) \vee (p27 \wedge (\overline{CycE} \wedge \overline{CycA}) \wedge \overline{CycD} \wedge \overline{CycB})$
x_4	<i>E2F</i>	$f_4 = (\overline{Rb} \wedge \overline{CycA} \wedge \overline{CycB}) \vee (p27 \wedge \overline{Rb} \wedge \overline{CycB})$
x_5	<i>CycE</i>	$f_5 = (E2F \wedge \overline{Rb})$
x_6	<i>CycA</i>	$f_6 = (E2F \wedge \overline{Rb} \wedge \overline{Cdc20} \wedge \overline{(Cdh1 \wedge UbcH10)}) \vee (CycA \wedge \overline{Rb} \wedge \overline{Cdc20} \wedge \overline{(Cdh1 \wedge UbcH10)})$
x_7	<i>Cdc20</i>	$f_7 = CycB$
x_8	<i>Cdh1</i>	$f_8 = (\overline{CycA} \wedge \overline{CycB}) \vee Cdc20 \vee (p27 \wedge \overline{CycB})$
x_9	<i>UbcH10</i>	$f_9 = \overline{Cdh1} \vee (Cdh1 \wedge UbcH10 \wedge (Cdc20 \vee CycA \vee CycB))$
x_{10}	<i>CycB</i>	$f_{10} = (\overline{Cdc20} \wedge \overline{Cdh1})$

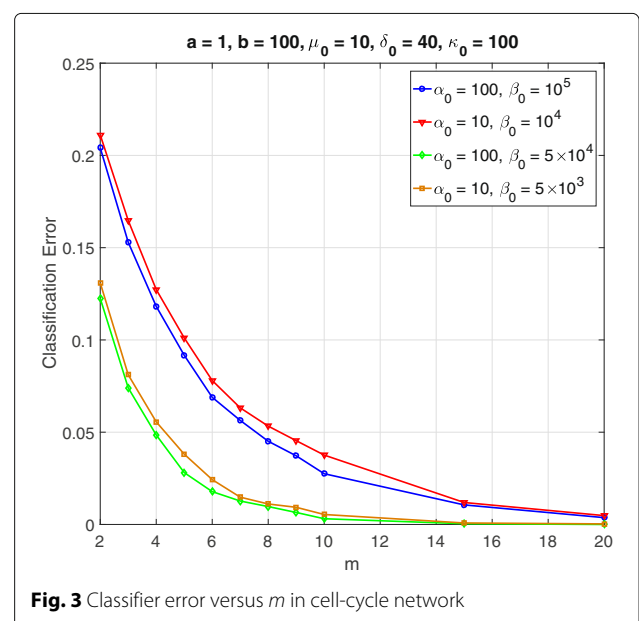
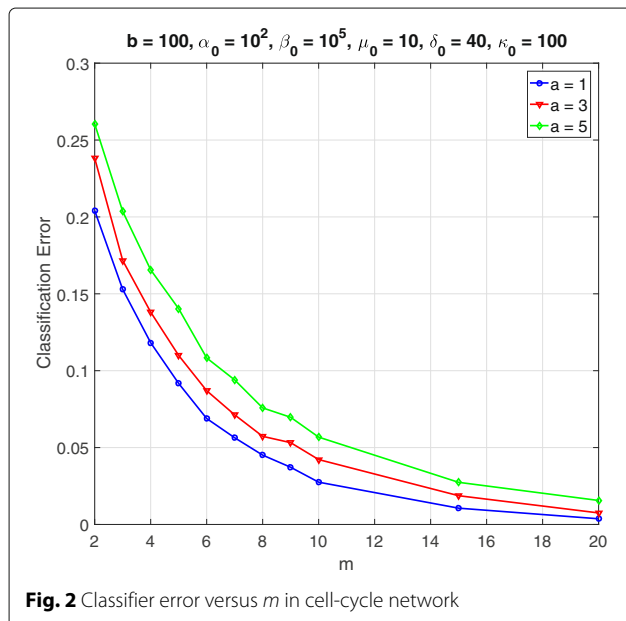
Figure 9 illustrates the error versus δ_0 , the expression coefficient, for $m = 5, a = 1, b = 100, \kappa_0 = 100, \alpha_0 = 100$, and $\beta_0 = 10^5$. Having larger δ_0 means that the mean values of data for each gene in the unexpressed and expressed cases are well separated, which leads to lower classification error. As expected, in Fig. 9 the error is decreasing as δ_0 gets larger.

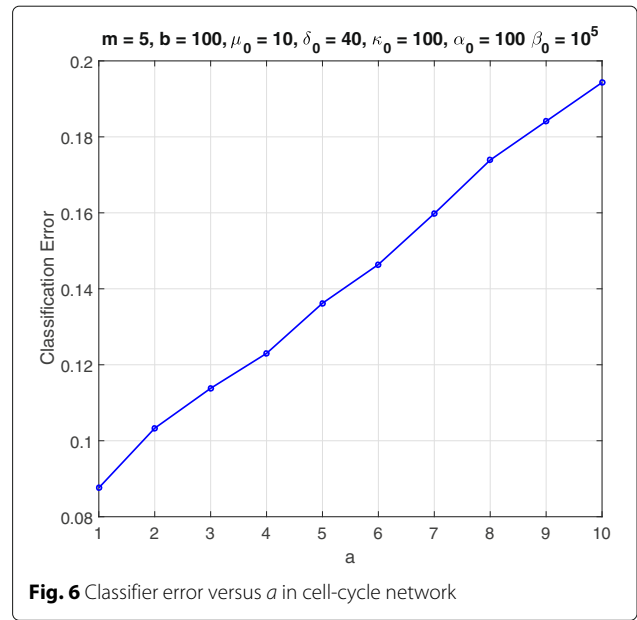
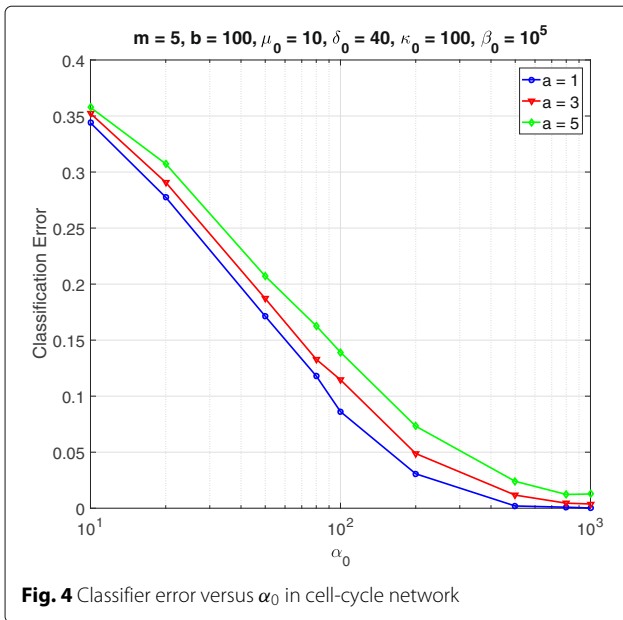
Conclusions

In this paper, we propose a trajectory-based intrinsically Bayesian robust classifier for classification of single-cell gene-expression trajectories. We assume that the expressions of the n genes, whose interactions are known in terms of a Boolean network, are observed in m consecutive time points, for both the wild-type class ($S = 0$) and mutated class ($S = 1$). As the parameters have uncertainty, we assign priors for them. We assume a beta distribution

as a prior for both the probability of the class $S = 0$ and the gene perturbation probabilities at each time. We assume a normal-gamma distribution for the mean and precision of the expressions at each time and for each gene, given the underlying states. As such, we derive closed-form solutions for the effective class-conditional densities of the trajectories in each class, by which we define the IBR classifier. The performance of the IBR classifier is evaluated in a cell-cycle gene regulatory network with 10 genes, for which we know the Boolean networks of the two classes. We have analyzed the effects of m and all the hyperparameters on the classification error.

In this paper we have assumed the form of the prior distributions, including the hyperparameter values. Once this is done, the analysis is mathematical: derive the effective class-conditional densities. As is generally the case with Bayesian methods, two issues arise: how are the

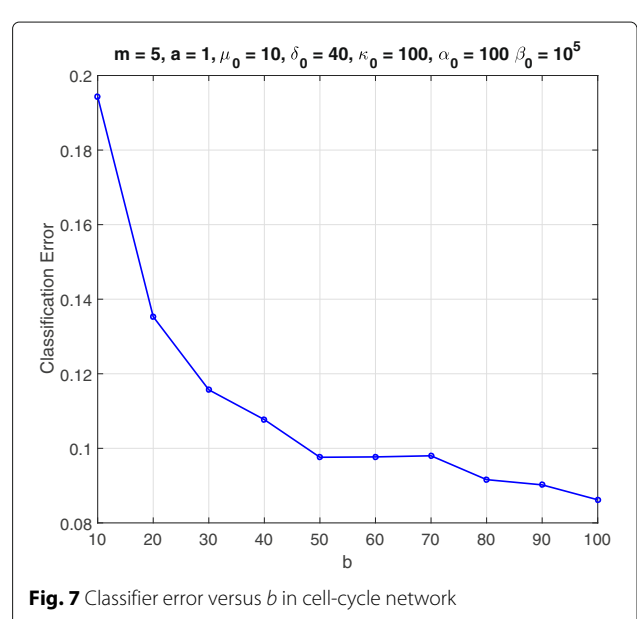
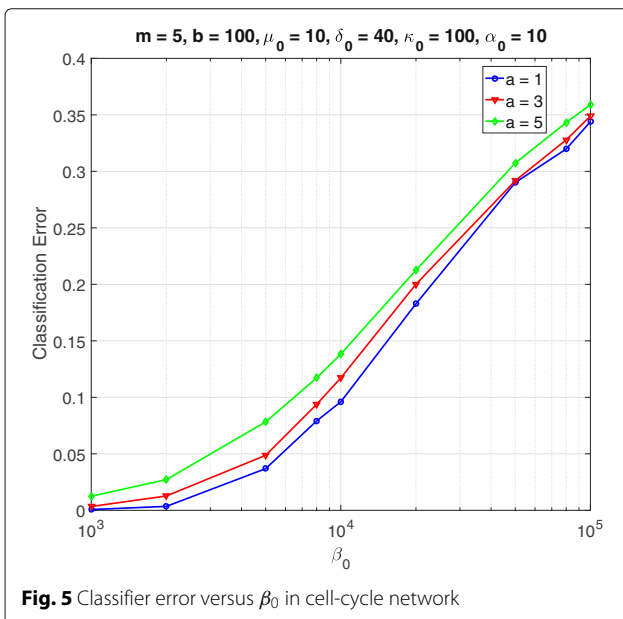


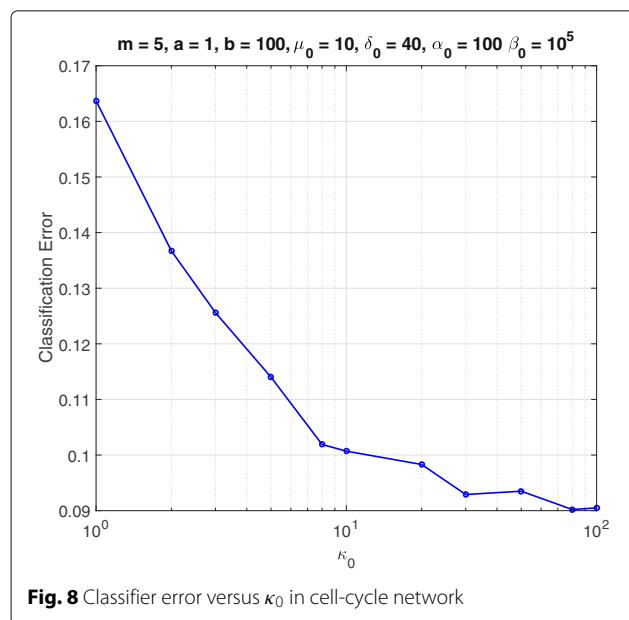


priors developed and how robust are the methods to the prior assumptions? Both issues have been extensively studied in regard to the OBC, of which the IBR classifier is the special case in which there is no update to a posterior.

In [9] robustness of the OBC relative to various modeling assumptions has been investigated, including false assumptions on the covariance structure and falsely assuming Gaussianity. In [20], similar robustness issues have been considered for the MMSE error estimator, which is required for defining the OBC.

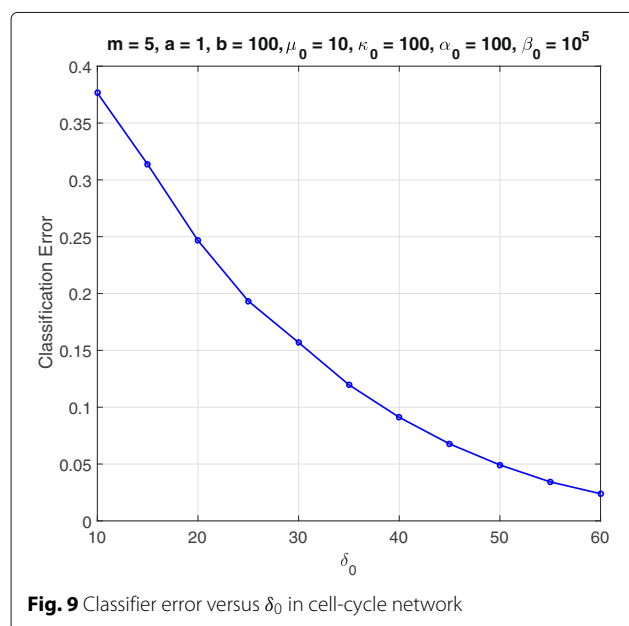
Regarding prior construction, the question is how to transform existing scientific knowledge into a prior distribution. Because the IBR classifier and other IBR filtering methods involve prior distributions on the underlying scientific model, for instance, the stochastic process (power spectra) in Wiener filtering and the feature-label distribution in classification, uncertainty arises from insufficient scientific knowledge, the prior characterizes that uncertainty, and, in effect, constrains the optimization relative to that uncertainty. For IBR/OBC classification in





the context of uncertain gene regulatory networks, prior construction methods based on constrained optimization have been developed for both Gaussian [21] and discrete [22] gene regulatory networks under the assumption of static observations. Future work involves extending these prior-construction methods to trajectories, which should be possible, albeit, with more difficult mathematical analysis and substantially more computation.

Another issue to be addressed in future work is extending the trajectory-based IBR classifier to optimal Bayesian classification, which will require analytic representation



of the effective class-conditional densities relative to a posterior distribution derived from the prior distribution utilizing sample data.

Additional file

Additional file 1: Appendices. This additional file contains the appendices for the proofs of the lemmas 1, 2, and 3. (PDF 33 kb)

Abbreviations

BN: Boolean network; BNp: Boolean network with perturbation; BOA: Basin of attraction; BPOBDS: Bayesian partially observed Boolean dynamical system; EM: Expectation maximization; GRN: Gene regulatory network; IBR: Intrinsically Bayesian robust; MMSE: Minimum mean square error; MSE: Mean square error; OBC: Optimal Bayesian classifier; TPM: Transition probability matrix

Acknowledgments

Not applicable.

Funding

Publication cost of this article was funded by the Robert M. Kennedy '26 Chair at Texas A&M University.

Availability of data and materials

Not applicable.

About this supplement

This article has been published as part of *BMC Systems Biology* Volume 12 Supplement 3, 2018: Selected original research articles from the Fourth International Workshop on Computational Network Biology: Modeling, Analysis, and Control (CNB-MAC 2017): systems biology. The full contents of the supplement are available online at <https://bmcsystbiol.biomedcentral.com/articles/supplements/volume-12-supplement-3>.

Authors' contributions

AK mathematically derived the classifier and worked out the example. UB with ED conceived the modeling and edited the original draft. ED structured the classifier, while UB conceived the modeling, and edited the original draft. All authors read and approved the final manuscript.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

¹Department of Electrical and Computer Engineering, Texas A&M University, 77843 College Station, TX, USA. ²Center for Bioinformatics and Genomic Systems Engineering, 77843 College Station, TX, USA.

Published: 21 March 2018

References

1. Wu AR, Neff NF, Kalisky T, Dalerba P, Treutlein B, Rothenberg ME, Mburu FM, Mantalas GL, Sim S, Clarke MF, et al. Quantitative assessment of single-cell rna-sequencing methods. *Nat Methods*. 2014;11(1):41–6.
2. Ståhlberg A, Kubista M. The workflow of single-cell expression profiling using quantitative real-time pcr. *Expert Rev Mol Diagn*. 2014;14(3):323–31.
3. Shmulevich I, Dougherty ER, Zhang W. From boolean to probabilistic boolean networks as models of genetic regulatory networks. *Proc IEEE*. 2002;90(11):1778–92.

4. Karbalayghareh A, Hu T. Inference of sparse gene regulatory network from rna-seq time series data. In: 2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP). 2015. p. 967–71. <https://doi.org/10.1109/GlobalSIP.2015.7418341>.
5. Karbalayghareh A, Braga-Neto U, Hua J, Dougherty ER. Classification of state trajectories in gene regulatory networks. *IEEE/ACM Trans Comput Biol Bioinforma.* 2017;PP(99):1. <https://doi.org/10.1109/TCBB.2016.2616470>.
6. Karbalayghareh A, Braga-Neto U, Dougherty ER. Classification of single-cell gene expression trajectories from incomplete and noisy data. *IEEE/ACM Trans Comput Biol Bioinforma.* 2017;PP(99):1. <https://doi.org/10.1109/TCBB.2017.2763946>.
7. Karbalayghareh A, Braga-Neto U, Dougherty ER. Classification of gaussian trajectories with missing data in boolean gene regulatory networks. In: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2017. p. 1078–82. <https://doi.org/10.1109/ICASSP.2017.7952322>.
8. Dalton LA, Dougherty ER. Optimal classifiers with minimum expected error within a bayesian framework—part i: Discrete and gaussian models. *Pattern Recogn.* 2013;46(5):1301–14.
9. Dalton LA, Dougherty ER. Optimal classifiers with minimum expected error within a bayesian framework — part ii: Properties and performance analysis. *Pattern Recogn.* 2013;46(5):1288–300.
10. Dalton LA, Dougherty ER. Intrinsically optimal bayesian robust filtering. *IEEE Trans Signal Process.* 2014;62(3):657–70.
11. Qian X, Dougherty ER. Bayesian regression with network prior: Optimal bayesian filtering perspective. *IEEE Trans Signal Process.* 2016;64(23):6243–53.
12. Dehghannasiri R, Esfahani MS, Dougherty ER. Intrinsically bayesian robust kalman filter: An innovation process approach. *IEEE Trans Signal Process.* 2017;65(10):2531–46.
13. Boluki S, Esfahani MS, Qian X, Dougherty ER. Constructing pathway-based priors within a gaussian mixture model for bayesian regression and classification. *IEEE/ACM Trans Comput Biol Bioinforma.* 2017;PP(99):1.
14. Boluki S, Esfahani MS, Qian X, Dougherty ER. Incorporating biological prior knowledge for bayesian learning via maximal knowledge-driven information priors. *BMC Bioinformatics.* 2017;18(14):552.
15. Dadaneh SZ, Qian X, Zhou M. Bnp-seq: Bayesian nonparametric differential expression analysis of sequencing count data. *J Am Stat Assoc.* 2017;0(ja):0. <https://doi.org/10.1080/01621459.2017.1328358>.
16. Imani M, Braga-Neto UM. Maximum-likelihood adaptive filter for partially observed boolean dynamical systems. *IEEE Trans Signal Process.* 2017;65(2):359–71.
17. Fauré A, Naldi A, Chaouiya C, Thieffry D. Dynamical analysis of a generic boolean model for the control of the mammalian cell cycle. *Bioinformatics.* 2006;22(14):124–31.
18. Dalton LA, Dougherty ER. Bayesian minimum mean-square error estimation for classification error-part i: Definition and the bayesian mmse error estimator for discrete classification. *IEEE Trans Signal Process.* 2011;59(1):115–29.
19. Kschischang FR, Frey BJ, Loeliger HA. Factor graphs and the sum-product algorithm. *IEEE Trans Inf Theory.* 2001;47(2):498–519.
20. Dalton LA, Dougherty ER. Bayesian minimum mean-square error estimation for classification error-part ii: Linear classification of gaussian models. *IEEE Trans Signal Process.* 2011;59(1):130–44.
21. Esfahani MS, Dougherty ER. Incorporation of biological pathway knowledge in the construction of priors for optimal bayesian classification. *IEEE/ACM Trans Comput Biol Bioinforma.* 2014;11(1):202–18.
22. Esfahani MS, Dougherty ER. An optimization-based framework for the transformation of incomplete biological knowledge into a probabilistic structure and its application to the utilization of gene/protein signaling pathways in discrete phenotype classification. *IEEE/ACM Trans Comput Biol Bioinforma.* 2015;12(6):1304–21.

Submit your next manuscript to BioMed Central and we will help you at every step:

- We accept pre-submission inquiries
- Our selector tool helps you to find the most relevant journal
- We provide round the clock customer support
- Convenient online submission
- Thorough peer review
- Inclusion in PubMed and all major indexing services
- Maximum visibility for your research

Submit your manuscript at
www.biomedcentral.com/submit

