2023 IEEE Symposium Series on Computational Intelligence (SSCI)
Mexico City, Mexico. December 5-8, 2023

Inference of Genetic Networks from Steady-state and Pseudo Time-series of Single-cell Gene Expression Data using Modified Random Forests

Shuhei Kimura
Faculty of Engineering
Tottori University
Tottori, Japan
kimura@tottori-u.ac.jp

Hirotaka Kitajima
Graduate School of Sustainability Sciences
Tottori University
Tottori, Japan
m23j4011b@edu.tottori-u.ac.jp

Masato Tokuhisa
Faculty of Engineering
Tottori University
Tottori, Japan
tokuhisa@tottori-u.ac.jp

Mariko Okada
Institute for Protein Research
Osaka University
Osaka, Japan
mokada@protein.osaka-u.ac.jp

***Abstract*—A number of the genetic network inference methods have been proposed. These methods have been basically designed to analyze gene expression data of bulk cells. Recently, on the other hand, researchers have been capable of using gene expression data measured at single-cell resolution. The existing inference methods are however incapable of analyzing time-series of single-cell data because of high cell-to-cell variation in gene expression. This study therefore proposed the new inference method that has an ability to analyze steady-state and pseudo time-series of single-cell gene expression data. The pseudo time-series data are obtained through the pseudo-temporal ordering analysis. As the precise information about the measurement time is unavailable in pseudo time-series data, our method infers a genetic network using the signs of time derivatives of gene expression levels, that can be estimated from the given data. Through the numerical experiments, we finally confirmed the effectiveness of the proposed method.**

I. Introduction

High-throughput technologies, such as RNA-seq using next generation sequencers, enable us to measure expression levels of more than hundreds of genes. In order to extract useful information from these data, several researchers have focused on the inference of genetic networks. Many kinds of mathematical models have been proposed to describe genetic networks, and numerous inference methods based on individual models have been developed [2], [4], [9], [12], [16]. The inferred models, for example, can be used to identify genes that relate to a disease. In this study, we focus only on the inference methods based on ordinary differential equations. While these methods often require high computational costs to infer genetic networks, they obtain reasonable models capable of capturing various dynamics in gene expression.

Most of the existing inference methods have been designed to analyze gene expression data of bulk cells. Recently, on the other hand, gene expression data measured at single-cell resolution have been available for researchers. The inference methods capable of analyzing steady-state gene expression data of bulk cells are also capable of analyzing steady-state data measured at single-cell resolution. On the other hand, time-series datasets of single-cell gene expression levels, that are measured in different series of experiments, often show qualitatively different trends from each other because of high cell-to-cell variation in gene expression. Even when we apply the existing methods developed for analyzing time-series data of bulk cells to those measured at single-cell resolution, therefore, it is difficult to extract useful information.

In the single-cell data analysis, instead, we can use pseudo time-series of gene expression data. The pseudo time-series data are obtained through the pseudo-temporal ordering analysis that arranges measurements of single-cells according to their similarities [18]. The pseudo time-series data consist of the measurements of individual cells that are ordered along the pseudo time. The pseudo time reflects the progression of a cellular process, such as proliferation, differentiation, and so on. Note here that, although pseudo time-series of gene expression data are similar to time-series ones, they contain no precise information about the measurement time. As the existing inference methods capable of analyzing time-series data of bulk cells generally require the information about the measurement time, therefore, they are unable to analyze pseudo time-series data. Some inference methods that obtain ordinary differential equations from pseudo time-series data have been already proposed [11], [13]. These methods implicitly assume that the pseudo time defined through the pseudo-temporal ordering analysis is almost linearly proportional to the actual time. However, there is no guarantee that this assumption is always satisfied. When the assumption is unsatisfied, it is unclear that these inference methods are capable of producing reasonable results.

This study proposes a new method that has an ability to infer a genetic network from steady-state and pseudo time-series of single-cell gene expression data. While a lot of the existing methods capable of analyzing time-series data of bulk cells use time derivatives of the gene expression levels to infer genetic networks, the proposed method uses their signs. The estimation of the time derivatives requires the precise information about the measurement time. As mentioned previously, on the other hand, the pseudo time-series data consist of the measurements of individual cells that are ordered along the progression of a cellular process. Even when no precise information about the measurement time is available, therefore, we can estimate the signs of the

This work was supported by KAKENHI Grant Number 22K12193.

978-0-7381-4408-5/23/$31.00 ©2023 IEEE

time derivatives of the gene expression levels. Because of the feature of using the signs, thus, the proposed method has an ability to analyze pseudo time-series data. The performances of the random-forest-based inference methods, that were basically designed to obtain ordinary differential equations from gene expression data of bulk cells, are reportedly good [5], [6], [7], [10], [14]. The inference method proposed in this study is therefore designed based on two of them, i.e., GENIE3 [5] and its extension proposed by the author and colleagues [7]. Through numerical experiments with several artificial genetic network inference problems, we finally check the performance of the proposed inference method.

II. GENIE3 AND ITS EXTENSION

The basic concept of the proposed method comes from the model used in the random-forest-based inference method proposed by the author and colleagues [7]. On the other hand, the method proposed in this study repeatedly executes GENIE3 [5]. First, thus, this section will describe the random-forest-based inference method [7] that is an extension of GENIE3. Then, we will explain the relation between the inference method [7] and GENIE3 in the section II-D.

A. Model for describing genetic networks

The inference method proposed by the author and colleagues [7] assumes that a genetic network is represented as ordinaly differential equations of the form

$$\frac{dX_n}{dt} = F_n(\mathbf{X}_{-n}) - \beta_n X_n, \qquad (n = 1, 2, \cdots, N), \tag{1}$$

where $\mathbf{X}_{-n} = (X_1, \cdots, X_{n-1}, X_{n+1}, \cdots, X_N)$, X_m is the expression level of the m-th gene, N is the number of genes contained in the target network, β_n (> 0) is a constant parameter, and F_n is a function of arbitrary form.

The method [7] divides an inference problem of a genetic network consisting of N genes into N subproblems, each of which corresponds to each gene. By solving the n-th subproblem, the method obtains a reasonable approximation of the function F_n and a reasonable value for the parameter β_n. The method then computes the confidence values of the regulations of the n-th gene from the other genes by analyzing the obtained approximation of the function F_n. The sections II-B and II-C will describe ways to solve the n-th subproblem and to compute the confidence values, respectively.

B. Solving the n-th subproblem

The inference method [7] accomplishes the acquisition of an approximation of F_n and a value for β_n by using a weighted least-squares method. The method thus defines the n-th subproblem as the optimization problem of the following one-dimensional function.

$$S_n(\beta_n) = \sum_{k=1}^{K_T} \frac{w_k^T}{\beta_n} \left[\left.\frac{dX_n}{dt}\right|_{t_k} - \hat{F}_n\left(\mathbf{X}_{-n}|_{t_k}; \beta_n\right) + \beta_n X_n|_{t_k} \right]^2 + \sum_{k=1}^{K_S} \frac{w_k^S}{\beta_n} \left[\left.\frac{dX_n}{dt}\right|_{s_k} - \hat{F}_n\left(\mathbf{X}_{-n}|_{s_k}; \beta_n\right) + \beta_n X_n|_{s_k} \right]^2, \tag{2}$$

where $\mathbf{X}_{-n}|_{t_k} = (X_1|_{t_k}, \cdots, X_{n-1}|_{t_k}, X_{n+1}|_{t_k}, \cdots, X_N|_{t_k})$, $\mathbf{X}_{-n}|_{s_k} = (X_1|_{s_k}, \cdots, X_{n-1}|_{s_k}, X_{n+1}|_{s_k}, \cdots, X_N|_{s_k})$, and $X_m|_{t_k}$ and $X_m|_{s_k}$ are the expression levels of the m-th gene at the k-th measurements of time-series and steady-state experiments, respectively. $\left.\frac{dX_n}{dt}\right|_{t_k}$ and $\left.\frac{dX_n}{dt}\right|_{s_k}$ are the time derivatives of the expression levels of the n-th gene at the k-th measurements of the time-series and steady-state experiments, respectively. We can estimate $\left.\frac{dX_n}{dt}\right|_{t_k}$'s directly from the measured time-series of the gene expression levels using some smoothing technique. On the other hand, all of the values for $\left.\frac{dX_n}{dt}\right|_{s_k}$'s can be set to zeros because the data were measured under steady-state conditions. K_T (≥ 2) and K_S (≥ 0) are the numbers of measurements performed in the time-series and steady-state experiments, respectively. w_k^T and w_k^S are weight parameters for the k-th measurements in the time-series and steady-state experiments, respectively. Our earlier study [7] showed that the performance of the random-forest-based inference method is improved by assigning appropriate values to the parameters w_k^T's and w_k^S's. The author and colleagues have thus proposed the techniques to determine these values [7], [8].

$\hat{F}_n(\,\cdot\,; \beta_n)$ is an approximation of the function F_n trained under the given β_n. The computation of a value for the objective function (2) requires an approximation of the function F_n, i.e., $\hat{F}_n$. As the approximation of the function F_n, the inference method [7] uses the random forest [1] that is trained on the basis of the training data consisting of the following set of input-output pairs,

$$\left\{ \left(\mathbf{X}_{-n}|_{t_k}, \left.\frac{dX_n}{dt}\right|_{t_k} + \beta_n X_n|_{t_k} \right) \middle| k = 1, 2, \cdots, K_T \right\} \cup \left\{ \left(\mathbf{X}_{-n}|_{s_k}, \left.\frac{dX_n}{dt}\right|_{s_k} + \beta_n X_n|_{s_k} \right) \middle| k = 1, 2, \cdots, K_S \right\}.$$

Note that, when the method [7] trains the random forest, it considers the weight parameters w_k^T's and w_k^S's in order to keep consistency with the objective function (2). On the other hand, the training data described above contain the parameter β_n whose value must be determined. Note however that, when trying to compute a value for the objective function (2), a value for the parameter β_n is always given.

The random-forest-based inference method [7] uses the golden section search [15] to minimize the function (2).

C. Assigning confidence values to regulations

By analyzing the random forest obtained through the optimization of the function (2), the inference method [7] computes the confidence values of the regulations of the n-th gene from the other genes. The method assigns the confidence values to the regulations using the standard variable importance measure [1]. In order to compute the degree to which each of the input variables contributes the prediction of the output, this measure uses the total reduction of the variance of the output values due to the split. The method thus computes the confidence value of the regulation of the

n-th gene from the m-th gene, $C_{n,m}$, according to

$$C_{n,m} = \frac{1}{Sq_{w0}} \frac{1}{N_{tree}} \sum_{i=1}^{N_{tree}} \sum_{\nu \in V_i(m)} I(\nu), \tag{3}$$

where

$$\begin{aligned}
Sq_{w0} &= \sum_{k=1}^{K_T} w_k^T (y_{t_k} - \bar{y}_{w0})^2 + \sum_{k=1}^{K_S} w_k^S (y_{s_k} - \bar{y}_{w0})^2, && (4)\\
\bar{y}_{w0} &= \frac{1}{N_{w0}} \left[\sum_{k=1}^{K_T} w_k^T y_{t_k} + \sum_{k=1}^{K_S} w_k^S y_{s_k} \right], && (5)\\
N_{w0} &= \sum_{k=1}^{K_T} w_k^T + \sum_{k=1}^{K_S} w_k^S, && (6)\\
y_{t_k} &= \left. \frac{dX_n}{dt} \right|_{t_k} + \beta_n^* \, X_n|_{t_k}, && (7)\\
y_{s_k} &= \beta_n^* \, X_n|_{s_k}, && (8)\\
I(\nu) &= N_w(\nu) Sq_w(\nu) - N_w(\nu_L) Sq_w(\nu_L) \\
&\quad - N_w(\nu_R) Sq_w(\nu_R), && (9)\\
Sq_w(\nu) &= \sum_{k \in T(\nu)} w_k^T \left[y_{t_k} - \bar{y}_w(\nu) \right]^2 \\
&\quad + \sum_{k \in S(\nu)} w_k^S \left[y_{s_k} - \bar{y}_w(\nu) \right]^2, && (10)\\
\bar{y}_w(\nu) &= \frac{1}{N_w(\nu)} \left[\sum_{k \in T(\nu)} w_k^T y_{t_k} + \sum_{k \in S(\nu)} w_k^S y_{s_k} \right], && (11)\\
N_w(\nu) &= \sum_{k \in T(\nu)} w_k^T + \sum_{k \in S(\nu)} w_k^S, && (12)
\end{aligned}$$

N_{tree} is the number of trees in the random forest $\hat{F}_n^*$, and $V_i(m)$ is a set of nodes that use the expression levels of the m-th gene to split the training examples in the i-th decision tree of $\hat{F}_n^*$. $\hat{F}_n^*$ and β_n^* are the approximation of the function F_n and the value for the parameter β_n, respectively, that are obtained through the optimization of the function (2). ν_L and ν_R are the left and right children nodes of the node ν, respectively, and $T(\nu)$ and $S(\nu)$ are sets of indices of the training examples generated from time-series and static gene expression data, respectively, and allocated to the node ν.

D. Relation to GENIE3

The inference method [7] was designed based on GEINE3. GENIE3 proposed by Huynh-Thu and colleagues [5] is the first method that uses the random forests for the genetic network inference. GENIE3 is capable of analyzing the steady-state gene expression data only. Under the condition that the steady-state data are only given, the model (1) corresponding to the n-th gene can be described as

$$\left(\frac{dX_n}{dt} = \right) 0 = F_n \left(\mathbf{X}_{-n} \right) - \beta_n X_n. \tag{13}$$

By dividing both sides of the equation (13) by β_n and replacing $F_n \left(\mathbf{X}_{-n} \right) / \beta_n$ by $G_n \left(\mathbf{X}_{-n} \right)$, we have

$$0 = G_n \left(\mathbf{X}_{-n} \right) - X_n. \tag{14}$$

Note here that the inference method [7] computes the confidence values of the regulations of the n-th gene from the other genes by analyzing the approximation of the function F_n. On the other hand, the equation (14) indicates that the confidence values can be obtained also by analyzing a good approximation of the function G_n. GENIE3 obtains the approximation of the function G_n by training the random forest on the basis of the following set of input-output pairs.

$$\left\{ \left(\mathbf{X}_{-n}|_{s_k}, X_n|_{s_k} \right) \middle| k = 1, 2, \cdots, K_S \right\},$$

where $\mathbf{X}_{-n}|_{s_k} = (X_1|_{s_k}, \cdots, X_{n-1}|_{s_k}, X_{n+1}|_{s_k}, \cdots, X_N|_{s_k})$, $X_m|_{s_k}$ is the expression level of the m-th gene at the k-th measurement of a steady-state experiment, and K_S is the number of measurements performed in the steady-state experiment. Here, we should note again that GENIE3 infers a genetic network only from steady-state gene expression data.

While the inference method [7] uses time derivatives of gene expression levels to infer a genetic network, GENIE3 does not use them. GENIE3 is therefore capable of analyzing single-cell gene expression data whenever they are measured under steady-state conditions.

III. Inference method for single-cell data

In this section, we propose a new method that infers a genetic network from steady-stare and pseudo time-series of single-cell gene expression data. Similar to the random-forest-based inference method [7] described in the previous section, the method proposed in this study also divides an inference problem of a genetic network consisting of N genes into N subproblems, each of which corresponds to one gene. The remainder of this section will describe the n-th subproblem corresponding to the n-th gene.

A. Data for analysis

As described previously, in order to obtain the confidence values of the regulations of the n-th gene from the other genes, the inference method [7] solves the n-th subproblem. In the n-th subproblem, the method [7] uses the following data.

$$\left\{ \left(\mathbf{X}|_{t_k}, \left. \frac{dX_n}{dt} \right|_{t_k} \right) \middle| k = 1, 2, \cdots, K_T \right\}$$
$$\cup \left\{ \left(\mathbf{X}|_{s_k}, \left. \frac{dX_n}{dt} \right|_{s_k} \right) \middle| k = 1, 2, \cdots, K_S \right\},$$

where $\mathbf{X}|_{t_k} = (X_1|_{t_k}, X_2|_{t_k}, \cdots, X_N|_{t_k})$, $\mathbf{X}|_{s_k} = (X_1|_{s_k}, X_2|_{s_k}, \cdots, X_N|_{s_k})$, and $X_m|_{t_k}$ and $X_m|_{s_k}$ are the expression levels of the m-th gene at the k-th measurements in the time-series and steady-state data, respectively. $\left. \frac{dX_n}{dt} \right|_{t_k}$ and $\left. \frac{dX_n}{dt} \right|_{s_k}$ are the time derivatives of the expression levels of the n-th gene at the k-th measurements in the time-series and steady-state data, respectively. While $\left. \frac{dX_n}{dt} \right|_{s_k}$'s are all set to zeros, $\left. \frac{dX_n}{dt} \right|_{t_k}$'s are directly estimated from the measured time-series gene expression data.

Because pseudo time-series of single-cell gene expression data, that are produced by the pseudo-temporal ordering analysis [18], do not contain the precise information about the measurement time, we cannot estimate time derivatives of gene expression levels from these data. This is a reason why we cannot use the inference method [7] to analyze pseudo time-series data. As the pseudo-temporal ordering analysis arranges measurements of individual cells along the progression of a cellular process, on the other hand, we can estimate signs of time derivatives of gene expression levels from pseudo time-series data. In the n-th subproblem, in order to obtain the confidence values of the regulations of the n-th gene from the other genes, therefore, the proposed method uses the following data.

$$\left\{ \left(\mathbf{X}|_{pt_k}, Y_n|_{pt_k} \right) \middle| k = 1, 2, \cdots, K_{PT} \right\} \\ \cup \left\{ \left(\mathbf{X}|_{s_k}, Y_n|_{s_k} \right) \middle| k = 1, 2, \cdots, K_S \right\},$$

where $\mathbf{X}|_{pt_k} = (X_1|_{pt_k}, X_2|_{pt_k}, \cdots, X_N|_{pt_k})$, $\mathbf{X}|_{s_k} = (X_1|_{s_k}, X_2|_{s_k}, \cdots, X_N|_{s_k})$, and $X_m|_{pt_k}$ and $X_m|_{s_k}$ are the expression levels of the m-th gene at the k-th measurements in pseudo time-series and steady-state data, respectively. $Y_n|_{pt_k}$ and $Y_n|_{s_k}$ are class labels that represent the signs of the time derivatives of the expression levels of the n-th gene at the k-th measurements in the pseudo time-series and steady-state data, respectively. K_{PT} and K_S are the numbers of the measurements contained in the pseudo time-series and steady-state data, respectively.

This study assigns '+', '−' or '0' to $Y_n|_{pt_k}$ according to the sign of the slope of the expression level of the n-th gene at the k-th measurement in the pseudo time-series data (see Fig. 1). Specifically, we assign '+' to $Y_n|_{pt_k}$ if the slope of the expression level of the n-th gene at the k-th measurement is obviously positive in the pseudo time-series data. If the slope is obviously negative, we assign '−' to $Y_n|_{pt_k}$. If we can determine that the slope is almost 0, we assign '0' to $Y_n|_{pt_k}$. If it is difficult to assign either '+', '−' or '0' to the measurement, we remove it from the training data. We think that the examples erroneously labeled are harmful for inferring a genetic network. Since steady-state gene expression data are measured under steady-state conditions, on the other hand, we always assign '0' to $Y_n|_{s_k}$. For convenience of explanation, this study divides the data for the n-th subproblem, mentioned just above, into three subsets, i.e., D_n^+, D_n^- and D_n^0, that consist only of the measurements labeled '+', '−' and '0', respectively.

B. Concept

The proposed method also assumes that the model (1) represents a genetic network. Similar to the section II-D, we focus on the equation corresponding to the n-th gene in the model (1). By dividing both sides of this equation by β_n and replacing $F_n(\mathbf{X}_{-n})/\beta_n$ by $G_n(\mathbf{X}_{-n})$, we have

$$\frac{dX_n}{dt} \Big/ \beta_n = G_n(\mathbf{X}_{-n}) - X_n. \tag{15}$$

Note here that the parameter β_n is positive. The equation (15) therefore suggests that, if we know the function G_n, we can estimate the sign of the time derivative of the expression level of the n-th gene from the expression levels of all of the genes. Specifically, when values for $G_n(\mathbf{X}_{-n}) - X_n$ are positive, negative and zero, we can conclude that the signs of the time derivative of the expression level of the n-th gene are positive, negative and zero, respectively. Note also that GENIE3 mentioned in the section II-D [5] is capable of obtaining an approximation of the function G_n only from steady-state gene expression data. Therefore, we could estimate the sign of the time derivative of the expression level of the n-th gene using $\hat{G}_n(\mathbf{X}_{-n}) - X_n$, where $\hat{G}_n$ is the approximation of the function G_n, and this study obtains it using GENIE3.

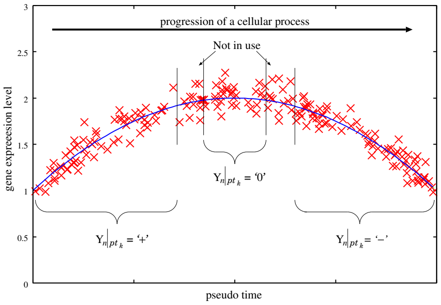

Fig. 1. Assignment of class labels to $Y_n|_{pt_k}$ from pseudo time-series data. × symbols indicate measurements in the pseudo time-series data. Solid line represent the smoothed expression data of the n-th gene.

The method proposed in this study uses the idea mentioned above to infer a genetic network from steady-state and pseudo time-series of single-cell gene expression data.

C. Algorithm

In this study, we propose a new inference method based on the concept described just above. Specifically, in the n-th subproblem, the proposed method executes GENIE3 only using the subset D_n^0, and then obtains the random forest that approximates the function G_n. Note that the random forest obtained by GENIE3 consists of multiple regression trees, each of which is also an approximation of the function G_n. From the regression trees in the random forest obtained, then, our method selects the most consistent one with the examples of the subsets D_n^+ and D_n^-. By repeating the procedure described above, the method tries to construct a tree ensemble that is consistent with all of the examples in the given data. Note that, although the tree ensemble is constructed through the algorithm slightly different from that of the random forest, its structure is identical to that of the random forest. By analyzing the tree ensemble obtained, thus, the proposed method computes the confidence values of the regulations of the n-th gene from the other genes.

Specifically, the proposed method solves the n-th subproblem corresponding to the n-th gene according to the following procedure.

1) Step 1 (Initialization): As input data, receive a set of pairs of single-cell gene expression data and their class labels, i.e., $D_n^+ \cup D_n^- \cup D_n^0$. As mentioned in the section II, on the other hand, our earlier study [7] showed that the performance of the existing random-forest-based inference method is improved by assigning appropriate values to the weight parameters. The proposed inference method therefore introduces them. We represent the weight parameters for the k-th measurements in the subsets D_n^+, D_n^- and D_n^0 as w_k^+, w_k^- and w_k^0, respectively. The proposed method also requires to have their values. In order to determine these values, we can use the technique proposed by the author and colleagues [8]. If the user does not want to use the weight parameters, on the other hand, set the values to 1.0. Finally, compute

$$W_n^+ = \sum_{k=1}^{|D_n^+|} w_k^+, \quad W_n^- = \sum_{k=1}^{|D_n^-|} w_k^-.$$

2) Step 2 (Execution of GENIE3): Run GENIE3 [5] using the subset D_n^0. Note that we can construct the training dataset for GENIE3 from the subset D_n^0. The number of trees in the random forest of GENIE3 is represented as $N_{subtree}$. Note also that, while the original GENIE3 does not consider the weight parameters, GENIE3 used in this study analyzes the given data considering the weight parameters w_k^0's.

3) Step 3 (Selection of examples): Select $K_{example}$ examples randomly from the subset D_n^+ if $W_n^+ \leq W_n^-$. Otherwise, select $K_{example}$ examples randomly from the subset D_n^-. Note that, if $K_{example}$ exceeds the number of the elements in the subset, select all examples from the subset. Compute the sum of the weight values corresponding to the selected examples. Here, we represent its value as $W_{example}$. Then, select examples randomly from the opposite subset while the sum of the weight values corresponding to the selected examples is less than $W_{example}$. The sets of indices of the examples selected from the subsets D_n^+ and D_n^- are denoted by S^+ and S^-, respectively.

4) Step 4 (Selection of regression tree): The random forest constructed by GENIE3 in the step 2 consists of $N_{subtree}$ regression trees, denoted here by $\hat{G}_n^1, \hat{G}_n^2, \cdots, \hat{G}_n^{N_{subtree}}$. Select the best tree among these trees, and remove the others. The goodness of the tree $\hat{G}_n^i$ is evaluated using

$$\begin{aligned} T_n(\hat{G}_n^i) = &\sum_{k \in S^+} w_k^+ \times max\left\{-[\hat{G}_n^i(\mathbf{X}_{-n}|_k^+) - X_n|_k^+], 0\right\} \\ &+ \sum_{k \in S^-} w_k^- \times max\left\{[\hat{G}_n^i(\mathbf{X}_{-n}|_k^-) - X_n|_k^-], 0\right\}, \end{aligned} \quad (16)$$

where $\mathbf{X}_{-n}|_k^+ = (X_1|_k^+, \cdots, X_{n-1}|_k^+, X_{n+1}|_k^+, \cdots, X_N|_k^+)$, $\mathbf{X}_{-n}|_k^- = (X_1|_k^-, \cdots, X_{n-1}|_k^-, X_{n+1}|_k^- \cdots, X_N|_k^-)$, and $X_m|_k^+$ and $X_m|_k^-$ are the expression levels of the m-th gene of the k-th measurements in the subsets D_n^+ and D_n^-, respectively. Note that, as mentioned in the section III-B, the good approximation of the function G_n enables us to estimate the sign of the time derivative of the expression level of the n-th gene. A value for the function T_n thus represents the degree of the misclassification done by the tree. The best tree is therefore the one having the minimum value for the function T_n. If there are multiple trees having the minimum value for the function T_n, choose one randomly from them.

5) Step 5 (Conditional branch): If the total number of the regression trees selected in the previous step reaches N_{tree}, proceed to the next step. Otherwise, return to the step 2.

6) Step 6 (Computation of confidence values): An ensemble of the selected regression trees has a structure equivalent to that of the random forest. Therefore, we can compute the confidence values of the regulations by the means identical to that of the random-forest-based inference method [7]. Specifically, the proposed method computes the confidence value of the regulation of the n-th gene from the m-th gene according to the equation (3). Note however that the proposed method only uses the gene expression data in the subset D_n^0 to train the regression trees. When computing the confidence values, therefore, we do not use the gene expression data in the subsets D_n^+ and D_n^-.

D. Remarks

As the inference method proposed in this study uses GENIE3, we must give gene expression data labeled '0'. Although a small amount of these data might be constructed from pseudo time-series data, most of them will be generated from steady-state data. When using the proposed inference method, therefore, we should give a sufficient amount of steady-state gene expression data. As described previously, however, our earlier study [7] suggests that, even when we give measurements similar to each other, they do not contribute to improve the quality of the inferred genetic network. Thus, it is desirable if these data contain gene expression data of different gene knockout cells, cells took from different patients, or the like.

While the proposed method does not work without gene expression data labeled '0', on the other hand, the method is capable of inferring a genetic network without those labeled '$+$' or '$-$'. Note however that, when gene expression data labeled either '$+$' or '$-$' are unavailable, the proposed inference method is equivalent to GENIE3. The inference method proposed in this study can be thus regarded as yet another extension of GENIE3.

IV. Numerical experiments

In order to check the performance of the proposed method, this study performs the experiments. However, we do not yet have any artificial problem that simulates the genetic network inference using single-cell gene expression data. In this study, we thus applies the proposed method to five artificial genetic network inference problems obtained from the DREAM4 *in silico* network challenges (http://dreamchallenges.org/). The DREAM4 problems simulate the genetic network inference using gene expression data of bulk cells.

A. Construction of training data

In the DREAM4 problems, we can utilize the information about the measurement time. Note however that the purpose in this study is to propose a new method capable of inferring genetic networks from single-cell gene expression data that contain no precise information about the measurement time. In this experiments, therefore, we inferred genetic networks without using this information. For the proposed method, thus, we constructed the training data that do not contain

TABLE I
THE AURPCS OF THE PROPOSED METHOD ON THE DREAM4 PROBLEMS. THE PERFORMANCES OF GENIE3 [5] AND THE RANDOM-FOREST-BASED INFERENCE METHOD [7] ARE ALSO SHOWN. NOTE THAT GENIE3 USES THE TRAINING DATA LABELED ‘0’ ONLY. NOTE ALSO THAT THE RANDOM-FOREST-BASED INFERENCE METHOD USES THE ESTIMATED TIME DERIVATIVES OF THE GENE EXPRESSION LEVELS. AVG AND STD REPRESENT THE AVERAGED AURPC AND ITS STANDARD DEVIATION, RESPECTIVELY.

	Network1	Network2	Network3	Network4	Network5	Note
	AVG	AVG	AVG	AVG	AVG	
	± STD	± STD	± STD	± STD	± STD	
The proposed method	0.41431 ±0.00272	0.34206 ±0.00121	0.37616 ±0.00138	0.41877 ±0.00162	0.28840 ±0.00213	The training data labeled ‘$+$’, ‘$-$’ and ‘0’ are used.
GENIE3 [5]	0.26522 ±0.00690	0.23955 ±0.00254	0.31216 ±0.00412	0.27497 ±0.00272	0.20952 ±0.00188	The training data labeled ‘0’ are only used.
Random-forest-based inference method [7]	0.41050 ±0.00160	0.28021 ±0.00292	0.35255 ±0.00429	0.34084 ±0.00420	0.29945 ±0.00294	The time derivatives of gene expression levels are required as the training data.

TABLE II
THE AURPCS OF THE PROPOSED METHOD WITH $K_{example} = 2, 4, 8, 16$ AND 32 ON THE DREAM4 PROBLEMS.

	Network1	Network2	Network3	Network4	Network5
	AVG	AVG	AVG	AVG	AVG
	± STD	± STD	± STD	± STD	± STD
The proposed method $K_{example} = 2$	0.43114 ±0.00181	0.32482 ±0.00193	0.36211 ±0.00164	0.38219 ±0.00214	0.27491 ±0.00270
The proposed method $K_{example} = 4$	0.43111 ±0.00256	0.33903 ±0.00213	0.36876 ±0.00173	0.40670 ±0.00191	0.28296 ±0.00197
The proposed method $K_{example} = 8$	0.41431 ±0.00272	0.34206 ±0.00121	0.37616 ±0.00138	0.41877 ±0.00162	0.28840 ±0.00213
The proposed method $K_{example} = 16$	0.39608 ±0.00217	0.33514 ±0.00266	0.37586 ±0.00250	0.41292 ±0.00194	0.28604 ±0.00161
The proposed method $K_{example} = 32$	0.38696 ±0.00191	0.32400 ±0.00184	0.37587 ±0.00169	0.39650 ±0.00115	0.28081 ±0.00109

the time derivatives of the gene expression levels but contain their signs, as described below.

The target networks in the DREAM4 problems consisted of 100 genes, and were designed based on actual biochemical networks. Each problem contained both the time-series and static expression data of all 100 genes. The time-series data were 10 datasets of time-series of gene expression levels obtained by solving a set of differential equations on the target network [17]. Each time-series dataset consisted of the expression levels at 21 time points, and was polluted by internal and external noise. A dataset was constructed by applying a perturbation to the network at the 1st time point and removing the perturbation at the 11th time point. The perturbation affected the transcription rates of a different set of genes in each dataset. To take the perturbations into account explicitly, we added 10 elements to the gene expression data, each corresponding to each of the perturbations. The i-th added element had a value of 1.0 for the measurements between the 1st and 10th time points in the i-th time-series dataset generated by adding the i-th perturbation, and a value of 0.0 for the other measurements. The number of elements, N, was therefore $100 + 10 = 110$. The local linear regression [3] was used to smooth the given time-series data and to estimate the time derivatives of the gene expression levels. For the n-th subproblem corresponding to the n-th gene, we then assigned the class label to each of the measurements according to the following rule: if the estimated value for the time derivative of the expression level of the n-th gene at the k-th measurement, i.e., $\frac{dX_n}{dt}\Big|_{t_k}$, is larger than 0.0004, we set the class label of the measurement to ‘$+$’; if $\frac{dX_n}{dt}\Big|_{t_k} < -0.0004$, we assign ‘$-$’ to the measurement; if $\left|\frac{dX_n}{dt}\Big|_{t_k}\right| \leq 0.0001$, we set its class label to ‘0’; otherwise, we did not use the measurement in the n-th subproblem. Note therefore that, although the time-series data in each of the DREAM4 problems consisted of $10 \times 21 = 210$ measurements, our method did not use some of them.

The static data, on the other hand, consisted of wild-type, knockout and knockdown data. The wild-type data contained the steady-state gene expression levels of the unperturbed network. The knockout and knockdown data contained the steady-state expression levels of every single-gene knockout and knockdown, respectively. When trying to solve the n-th subproblem corresponding to the n-th gene, however, we removed the static data of the knockout and the knockdown of the n-th gene. The number of measurements in the steady-state data was thus $1 + 100 + 100 - 2 = 199$. We assigned ‘0’ to all of the measurements in the steady-state data.

We inferred genetic networks only using the gene expression levels and their class labels, that were constructed according to the procedure described just above.

B. Experimental setup

According to the recommended values in the random-forest-based inference method [7], we set the number of trees in the tree ensemble, i.e., N_{tree}, the number of input variables to be considered in each internal node of each tree, i.e., N_{test}, and the maximum height of each tree, i.e., N_{hmax}, to 1000, $\left\lceil \frac{N-1}{3} \right\rceil$, and 32, respectively. Based on the results of our preliminary experiment, on the other hand, we set the number of examples selected in the step 3 of the proposed method, i.e., $K_{example}$, and the number of trees in the random forest of GENIE3 used in the proposed method, i.e., $N_{subtree}$, to 8 and 50, respectively. The values for the weight parameters w_k^+'s, w_k^-'s and w_k^0's were determined using the weighting method [8] (see Appendix). This study applied the proposed inference method to each of the five DREAM4 problems ten times.

C. Results

1) Performance: In order to quantify the performance of the inference method, we used the area under the recall-precision curve (AURPC). The recall-precision curve of an algorithm is obtained by checking the recalls and precisions of the algorithm. The recall and the precision are defined as

$$\text{recall} = \frac{TP}{TP+FN}, \quad \text{precision} = \frac{TP}{TP+FP},$$

where TP, FP and FN are the numbers of true-positive, false-positive and false-negative regulations, respectively. This study computed the recall and precision by constructing a network of regulations whose confidence values exceeded a threshold and then comparing it with the gold-standard network. Next, we obtained the recall-precision curve of the algorithm by changing the threshold for the confidence values. Auto-regulations/auto-degradations were disregarded in the evaluation of the performance. In addition, although we inferred the regulations of the 100 genes from these genes and the 10 additional elements representing 10 perturbations, we disregarded the regulations of the genes from the additional elements for the evaluation of the performance.

The AURPCs of the proposed method on the DREAM4 problems are listed in Table I. We compared the performance of the proposed method with that of GENIE3 [5]. As shown in the table, our method always outperformed GENIE3 on the DREAM4 problems. Note here that, while our method is capable of using the dataset $D_n^+ \cup D_n^- \cup D_n^0$, GENIE3 uses the dataset D_n^0 only. The experimental results thus indicate that the use of the training data labeled '+' and '−' improves the quality of the inferred network.

Table I also shows the AURPCs of the random-forest-based inference method [7] using the appropriate weight values [8]. Note that, while the existing inference method [7] uses time derivatives of gene expression levels, the method proposed in this study uses their signs. In order to infer genetic networks, therefore, the proposed method utilizes a smaller amount of information. Surprisingly, however, the proposed method outperformed the existing method [7] on four of the five DREAM4 problems. The poor performance of the random-forest-based inference method might be caused by a reason that the erroneously estimated time derivatives of gene expression levels were given in these problems.

2) Parameter $K_{example}$: The number of training examples labeled '+' and '−', that are used to find the best regression tree in the step 4 of the proposed method, is determined according to the parameter $K_{example}$. When the regression trees selected are consistent with a large number of training examples labeled '+' and '−', the tree ensemble finally obtained would be better. Note however that our method constructs regression trees without considering any training examples labeled '+' and '−'. Even when we set the parameter $K_{example}$ to a large value, therefore, we do not always obtain regression trees consistent with many training examples. In order to find a reasonable value for the parameter $K_{example}$, thus, we performed the experiments by changing its value.

Table II lists the AURPCs of the proposed method with different values for the parameter $K_{example}$ on the DREAM4 problems. As shown in the table, our setting, i.e., $K_{example} = 8$, made the performance of the proposed method relatively good. However, the table indicates that the best value for the parameter $K_{example}$ depends on the problem applied. In addition, we think that the best value for this parameter also depends on the parameter $N_{subtree}$. In our future work, thus, we should find a way to determine its value.

3) Direct estimation of time derivatives from pseudo time-series data: As mentioned previously, it is difficult to estimate time derivatives of gene expression levels from pseudo time-series data. In this study, we thus proposed the new inference method that does not require the precise estimation of time derivatives of gene expression levels. If the existing inference method is capable of inferring a genetic network without using the precisely estimated time derivatives, however, the proposed method might be useless.

In order to confirm the effectiveness of the proposed approach, therefore, this study performed another experiment. In this experiment, we checked the performances of the proposed method and the random-forest-based inference method [7] on the problem where the estimated time derivatives of gene expression levels are unreliable. We constructed unreliable data by multiplying each of the estimated time derivatives of gene expression levels in the 1st DREAM4 problem, i.e., Network1, by a randomly generated positive value. Specifically, we multiplied the estimated time derivative of the expression level of the n-th gene at the i-th measurement by $5^{r_{n,i}}$, where $r_{n,i}$ is a random value drawn from $[-1,1]$. Note that this transformation of the time derivatives of gene expression levels keeps their signs unchanged. This experiment simulates the situation where time derivatives of gene expression levels are directly estimated from the given pseudo time-series data.

By applying the inference method [7] to the time-series and steady-state gene expression data and the unreliable time derivatives constructed, we inferred a genetic network. We also inferred a genetic network by applying the proposed method to the same gene expression data and their class labels that were generated from the unreliable time derivatives according to the procedure described in the section IV-A. We performed ten trials by changing the unreliable time derivatives of gene expression levels. The AURPCs of our method and the inference method [7] were 0.44339 ± 0.00986 and 0.30198 ± 0.00713, respectively. This result indicates that, while the inference method designed for analyzing bulk-cell data requires the precise estimation of time derivatives of gene expression levels, the proposed method does not always require it. This feature would be preferable since there is no guarantee that the time derivatives are always estimated precisely from pseudo time-series data. As mentioned previously, the inference methods capable of analyzing pseudo time-series data have been already proposed [11], [13]. However, these methods implicitly assume that the pseudo time is almost linearly proportional to the actual time. For the reason similar to that described here, therefore, we think that they do not always infer reliable network models.

V. Conclusion

In order to extract useful information from single-cell gene expression data, this study proposed the new genetic network inference method. It is difficult to extract information from time-series of single-cell data because of high cell-to-cell variation in gene expression. The proposed method therefore uses steady-state and pseudo time-series of single-cell gene expression data. As the precise information about the measurement time is unavailable in pseudo time-series data, our method infers a genetic network using the signs of time derivatives of gene expression levels, that can be estimated from the given data. Through the numerical experiments, we showed the effectiveness of the proposed method. However, this study confirmed its effectiveness only on the artificial genetic network inference problems. In our future work, therefore, we should use our method to analyze actual single-cell gene expression data.

Appendix

The performance of the random-forest-based inference method [7] can be improved by assigning appropriate values to the weight parameters. In order to determine these values, the weighting method has been proposed [8]. This method computes weight values based on the similarities between measurements. We should note however that this weighting method was designed not for the inference method proposed in this study but for the existing inference method [7]. This section thus describes a way to compute weight values for the proposed method using this weighting method.

As mentioned in the section III-A, our approach removes some measurements from the observed single-cell gene expression data. This study performs the removal of measurements after the computation of weight values. Before computing weight values, the weighting method normalizes the given gene expression data so that the expression levels of each gene range from 0.0 to 1.0. Here, we represent the normalized measurements as $\overline{\mathbf{X}}\big|_1, \overline{\mathbf{X}}\big|_2, \cdots, \overline{\mathbf{X}}\big|_K$, where $\overline{\mathbf{X}}\big|_k = (\overline{X}_1\big|_k, \overline{X}_2\big|_k, \cdots, \overline{X}_N\big|_k)$, $\overline{X}_m\big|_k$ is the normalized expression level of the m-th gene at the k-th measurement, and K is the total number of measurements. The weighting method used in this study then computes a weight value corresponding to the k-th measurement, w_k, according to

$$w_k = \left[\sum_{i=1}^{K} Sim\left(\overline{\mathbf{X}}\big|_k, \overline{\mathbf{X}}\big|_i \right) \right]^{-1}, \tag{17}$$

where

$$Sim(\mathbf{x}, \mathbf{y}) = \exp\left(-C\,|\mathbf{x} - \mathbf{y}|^2 \right), \tag{18}$$

$$C = \frac{C_w}{\text{median } S_{all}}, \tag{19}$$

$$S_{all} = \left\{ \left| \overline{\mathbf{X}}\big|_i - \overline{\mathbf{X}}\big|_j \right|^2 \middle| \, i,j = 1, \cdots, K, i < j \right\}, \tag{20}$$

and C_w (> 0) is a constant parameter. The recommended value for C_w is 15.

References

[1] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.

[2] I.-C. Chou and E.O. Voit, "Recent developments in parameter estimation and structure identification of biochemical and genomic systems," *Mathematical Biosciences*, vol. 219, pp. 57–83, 2009.

[3] W.S. Cleveland, "Robust locally weight regression and smoothing scatterplots," *J. of American Statistical Association*, vol. 79, pp. 829–836, 1979.

[4] M. Hecker, S. Lambeck, S. Toepfer, E. van Someren and R. Guthke, "Gene regulatory network inference: Data integration in dynamic models – a review," *BioSystems*, vol. 96, pp. 86–103, 2009.

[5] V.A. Huynh-Thu, A. Irrthum, L. Wehenkel and P. Geurts, "Inferring regulatory networks from expression data using tree-based methods," *PLoS One*, vol. 5: e12776, 2010.

[6] V.A. Huynh-Thu and P. Geurts, "dynGENIE3: Dynamical GENIE3 for the inference of gene networks from time series expression data," *Scientific Reports*, vol. 8: 3384, 2018.

[7] S. Kimura, M. Tokuhisa and M. Okada, "Inference of genetic network using random forests: Assigning different weights for gene expression data," *J. of Bioinformatics and Computational Biology*, vol. 17: 1950015, 2019.

[8] S. Kimura, K. Sota and M. Tokuhisa, "Inference of genetic networks using random forests: A quantitative weighting method for gene expression data," *Proc. of 2022 IEEE CIBCB*, pp. 123–130, 2022.

[9] R. Larrañaga, B. Calvo, R. Santana, C. Bielza, J. Galdiano, I. Inza, J.A. Lozano, R. Armañanzas, G. Santafé, A. Pérez and V. Robles, "Machine learning in bioinformatics," *Briefings in Bioinformatics*, vol. 7, pp. 86–112, 2006.

[10] D.A.K. Maduranga, J. Zheng, P.A. Mundra and J.C. Rajapakse, "Inferring gene regulatory networks from time-series expression using random forests ensemble," *Pattern Recognition in Bioinformatics*, pp. 13–22, 2013.

[11] H. Matsumoto, H. Kiryu, C. Furusawa, M.S.H. Ko, S.B.H. Ko, N. Gouda, T. Hayashi and I. Nikaido, "SCODE: an efficient regulatory network inference algorithm from single-cell DNA-seq during differentiation," *Bioinformatics*, vol. 33, pp. 2314–2321, 2017.

[12] H. Nguyen, D. Tran, B. Tran, B. Pehlivan and T. Nguyen, "A comprehensive survey of regulatory network inference methods using single-cell RNA sequencing data," *Briefings in Bioinformatics*, vol. 22: bbaa190, 2020.

[13] A. Ocone, L. Haghverdi, N.S. Mueller and F.J. Theis, "Reconstracting gene regulatory dynamics from high-dimensional single-cell snapshot data," *Bioinformatics*, vol. 31, pp. i89–i296, 2015.

[14] F. Petralia, P. Wang, J. Yang and Z. Tu, "Integrative random forest for gene regulatory network inference," *Bioinformatics*, vol. 31, pp. i197–i205, 2015.

[15] W. Press, S Teukolsky, W. Vetterling and B. Flannery, *Numerical Recipes in C, 2nd edn.*, Cambridge University Press, UK, 1995.

[16] M.M. Saint-Antoine and A. Singh, "Network inference in systems biology: Recent developments, challenges, and applications," *Current Opinion in Biotechnology*, vol. 63, pp. 89–98, 2020.

[17] T. Schaffter, D. Marbach and D. Floreano, "GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods," *Bioinformatics*, vol. 27, pp. 2263–2270, 2011.

[18] T. Trapnell, D. Cacchiarelli, J. Grimsby, P. Pokharel, S. Li, M. Morse, N.J. Lennon, K.J. Livak, T.S. Mikkelsen and J.L. Rinn, "The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells," *Nature Biotechnology*, vol. 32, pp. 381–386, 2014.