BIOINFORMATICS

Vol. 00 no. 00 2010
Pages 1–12

Leveraging Gene Networks to Estimate Perturbations on Gene Expression

Nirmalya Bandyopadhyay, Manas Somaiya, Tamer Kahveci, Sanjay Ranka

Computer and Information Science and Engineering, University of Florida, Gainesville, FL
{nirmalya,mhs, tamer, ranka}@cise.ufl.edu

ABSTRACT

External factors such as radiation, drugs or chemotherapy can alter the expressions of a set of genes. We call these genes the *primarily affected* genes. Primarily affected genes can in time change the expressions of other genes as they activate/suppress each other through interactions. Measuring the gene expressions before and after applying an external factor (i.e., perturbations) in microarray experiments can reveal how the expression of each gene changes. It however can not tell the cause of the change.

In this paper, we consider the problem of identifying primarily affected genes given the expression measurements of a set of genes before and after the application of an external perturbation. We develop a new probabilistic method to quantify the cause of differential expression of each gene. Our method considers the possible gene interactions in regulatory and signaling networks, for a large number of perturbations. It uses a Bayesian model with the help of Markov Random Fields to capture the dependency between the genes. It also provides the underlying distribution of the impact with confidence interval.

Our experiments on both real and synthetic datasets demonstrate that our method can find primarily affected genes with high accuracy. In our experiments, our method was 100% accurate when the difference between expected expressions of primarily and secondarily affected genes is at least half of the standard deviation of the gene expressions. Our experiments also suggest that our method is significantly more accurate then SSEM, a recent relevant method, and the Studet's t-test.

1 INTRODUCTION

A significant set of microarray experiments measure the expressions of genes in the presence of external perturbations [8, 20]. In these experiments, also called perturbation experiments, perturbations such as radiation [38], drug [28] or other biological conditions are administered on tissues and their responses are monitored using microarrays. The expressions of the genes that are not subject to perturbations are called *control data*, while the expressions of genes after perturbations are called *non-control data* [17].

As response to an external perturbation, a fraction of genes can change their expression values significantly between control and non-control groups. Such genes are called *differentially expressed (DE)* genes [3]. All the other genes without noticeable change in expression are called *equally expressed (EE)* genes.

Often, some of the DE genes are directly affected by the external perturbation [13]. We denote them as the *primarily affected genes*. Rest of the genes change their expressions due to interactions with primarily affected genes and each other through signaling and regulatory networks [9, 10, 29, 36, 33]. We call them as *secondarily affected genes*. We refer to signaling and regulatory networks by *gene networks* in this paper. Figure 1 shows the state of the genes [2, 35] in the Pancreatic Cancer pathway after a hypothetical external perturbation is applied. In this figure, genes K-Ras, Raf and Cob42Roc are primarily affected and MEK, Ral and RalGDS are secondarily affected through interactions.

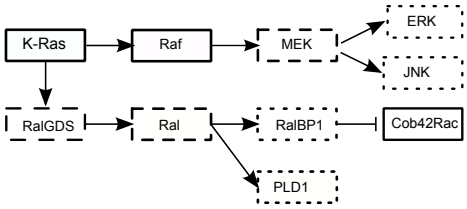


Fig. 1. Illustration of the impact of a hypothetical external perturbation on a small portion of the Pancreatic Cancer pathway. The pathway is taken from the KEGG database. The solid rectangles denote the genes affected directly by perturbation, the dashed rectangles indicate genes secondarily affected through the networks. The dotted rectangles denote the genes without any change in expression. $\rightarrow$ implies activation and $\dashv$ implies inhibition. In this figure, gene K-Ras, Raf and Cob42Roc are primarily affected and MEK, Ral and RalGDS are secondarily affected through interactions.

We consider the problem of identifying the primarily affected genes in a perturbation experiment, where gene expressions before and after the application of perturbation are provided for a set of samples.

Existing methods to identify the primarily affected genes using association analysis techniques [19, 28], haplotin-sufficiency profiling [16, 15, 27] and chemical-genetic interaction mapping [31] are limited to applications where additional information such as fitness based assays of drug response or a library of genetic mutants is available. Bernardo et al. [13] suggested a regression based approach, named MNI, that assumes that the internal genetic interactions are offset by the external perturbation. It estimates gene-gene interaction coefficients from the control data. It then uses those coefficients to predict the target genes in the non-control data. Cosgrove et. al. [9] proposed a method named SSEM that is similar to MNI. SSEM models the effect of perturbation by an explicit change of gene expression from that of the unperturbed state. These methods have several limitations.

1. *Lack of gene interaction data:* They do not employ regulatory or signaling (i.e. gene networks) to model gene-gene interactions. Since gene networks are manually curated using domain experts, they are reliable sources of gene interactions. Utilizing them has the potential to more accurately solve the problem of identifying primarily affected genes.

2. *Limited perturbations:* These methods are suitable when only a very small number of genes are perturbed, e.g., the genetic perturbation experiments are often designed for single gene perturbations [19]. However, external effects such as radiation can alter the expression of many genes directly making the existing methods to be of limited use.
3. *Simplistic models:* These methods provide only the set of genes that are directly affected by the perturbations and do not specify any error bounds. However, the change of the state of a gene is a stochastic event and a non-probabilistic inference oversimplifies the problem especially in cases when a small number of gene expression measurements are available. As a result, these methods can overfit the data, making the solution unreliable.

The method we propose in this paper addresses these limitations.

Contributions: Let $\mathcal{G} = \{g_1, g_2, \cdots, g_M\}$ denote the set of all genes. We assume that a microarray dataset with N samples is given for control and non-control groups. Let y_{ij} and $y_{ij}^{'}$ be the expression of the ith gene of the jth sample ($i \in \{1, 2, \cdots M\}$, $j \in \{1, 2, \cdots, N\}$) in the control and non-control groups respectively. For each gene g_i, we would like to estimate the probability that g_i is DE due to perturbation or due to each of the other genes.

In this paper, we propose a new probabilistic method that addresses the limitations of the existing methods to find the primarily affected genes in microarray dataset as defined above. Our method uses gene networks. We consider gene network as a directed graph where each node represents a gene, and a directed edge from gene a to gene b represents a genetic interaction (e.g a activates or inhibits b). We define two genes as *neighbors* of each other if they share an edge. For example, in Figure 1, genes K-Ras and Raf are neighbors as K-Raf activates Ras. We also classify a neighbor as *incoming* or *outgoing* if it is at the start or at the end of the directed edge respectively. In Figure 1, Raf is an incoming neighbor of MEK and MEK is an outgoing neighbor of Raf. When the expression level of a gene is altered, it can affect some of its outgoing neighbors. Thus, the expression of a gene can change due to external perturbation or because of one or more of the affected incoming neighbors. We build our solution based on this observation. We represent the external perturbation by a hypothetical gene (i.e. *metagene*) g_R in our the gene network. An edge from the metagene the to all the other genes imply that the external perturbation has the potential to affect all the other genes. So, g_R is an incoming neighbor to all the other genes. We call the resulting network the *extended gene network.*

Our method estimates the probability that a gene g_j is DE due to an alteration in the activity of gene g_i ($\forall\ g_i, g_j \in \mathcal{G} \cup \{g_R\}$) if there is an edge from g_i to g_j in the extended network. We use a Bayesian model in our solution with the help of Markov Random Field (MRF) to capture the dependency between the genes in the extended gene network. We optimize the likelihood of the joint posterior distribution over the random variables in the MRF using Iterated Conditional Mode (ICM) [6]. The optimization provides us the state of the genes and the pairwise causality among the genes including the metagene.

Our experiments on both real and synthetic datasets demonstrate that our method can find primarily affected genes with high accuracy. In our experiments, our method attains 100% when the difference between expected expressions of primarily and secondarily affected genes is at least half of the standard deviation of the gene expressions. We compared our method with SSEM and Student's t test and obtained significant higher accuracy in finding out the differentially expressed genes.

The rest of the paper is organized as follows. In Section 2 we describe our method in detail. In Section 3 we discuss the experiments and results. Finally, in Section 4 we describe our key conclusions.

2 METHODS

In this section, we describe our mathematical model and methods. Section 2.1 presents the notations. Section 2.2 provides an overview of our solution. Section 2.3 discusses the modeling of the MRF based prior distribution. Section 2.4 describes how we formulate a tractable approximate version of the objective function. Section 2.5 discusses how we compute the joint likelihood distribution of the expression of a gene. Section 2.6 explains how we optimize the objective function.

2.1 Notations and problem formulation

We start by describing the notation we use in the rest of this paper and provide a formal definition of the problem. We use two types of parameters to model this problem, namely *observed* and *hidden*. Observed variables are the ones whose values are available in the underlying data set. We derive the values of the hidden variables from the the observed data using the method.

Observed variables: There are two sets of observed variables.

- **Microarray dataset:** We denote the number of microarray samples and the number of genes by N and M respectively. We represent the set of all genes in the dataset with $\mathcal{G} = \{g_1, g_2, \cdots, g_M\}$. For each gene g_i, the dataset contains the expressions before and after the perturbation (i. e. control and non-control) respectively. We denote the expressions of g_i with y_{ij} and $y_{ij}^{'}$ in control and non-control group respectively, ($1 \leq i \leq M$, $1 \leq j \leq N$). Let $\mathbf{y_i} = \{y_{i1}, y_{i2}, \cdots y_{iN}\}$ and $\mathbf{y_i}^{'} = \{y_{i1}{}', y_{i2}{}', \cdots y_{iN}{}'\}$ denote the expression values of g_i in control and non-control groups respectively. We use Y_i to denote all the data for gene g_i in control and non-control groups (i.e. $Y_i = \mathbf{y_i} \cup \mathbf{y_i}^{'}$). We denote the collection of the entire gene expression data by $\mathcal{Y} = \bigcup_{i=1}^{M} Y_i$.
- **Neighborhood variables:** We use the term $\mathcal{W} = \{W_{ij}\}$ to represent if any two genes g_i and g_j are neighbors. We set the value of W_{ij} ($1 \leq i \leq M$, $1 \leq j \leq N$) to 1 if g_i is incoming neighbor of g_j (i.e. g_j has an incoming edge from g_i in the extended gene network.) and 0 otherwise.

Hidden Variables: There are two sets of hidden variables.

- **State variables:** Each gene g_i can attain one of the two states (i.e. DE or EE) We introduce the variables $\mathcal{S} = \{S_i\}$ to indicate the states of the genes. Formally, S_i is DE if g_i is differentially expressed and EE if g_i is equally expressed.
- **Interaction variables:** We define the set of random variables $\mathcal{X} = \{X_{ij}\}$ to represent the joint state of genes g_i and g_j

$(1 \leq i \leq M, 1 \leq j \leq N)$. Formally,

$$X_{ij} = \begin{cases} 1 & \text{if } S_i = \text{DE and } S_j = \text{DE;} \\ 0 & \text{if } S_i = \text{EE and } S_j = \text{EE;} \\ 2 & \text{if } S_i = \text{DE and } S_j = \text{EE;} \\ 3 & \text{if } S_i = \text{EE and } S_j = \text{DE;} \end{cases}$$

Problem formulation: We have microarray expression data $\mathcal{Y}$ and the gene network $\{\mathcal{G}, \mathcal{W}\}$ as input to the problem. We would like to estimate the conditional probability density function $p(X_{ij}|\mathcal{X} - X_{ij}, \mathcal{Y})$ when $W_{ij} = 1$.

2.2 Overview of our solution

An approach to solve our problem can be to maximize a likelihood distribution over the gene expression $\mathcal{Y}$ where $\mathcal{X}$ are the parameters of the distribution. The objective is to obtain the maximum likelihood estimate (MLE) of $\mathcal{X}$. However, there are two problems in this this approach. First, MLE requires a large number of data points to accurately estimate the parameters. Second, MLE depends only on the observed data and cannot utilize domain specific knowledge leading to overfitting of data and poor generalization.

We develop a method that uses a Bayesian estimation of $\mathcal{X}$ to address the above-mentioned limitations of the existing approaches. We compute a probability distribution over $\mathcal{X}$. To ameliorate overfitting, we incorporate the domain knowledge as the prior distribution. Also, Bayesian approach can generally estimate the parameters with fewer data-points, which makes our approach more suitable for perturbation experiments [7].

We estimate the probability of X_{ij} given the other observed and hidden variables. In this approach, we aim to maximize the joint probability of the X_{ij} variables given the gene expressions $\mathcal{Y}$. Thus, the objective is to maximize the joint distribution of $\mathcal{X}$ given by,

$$P(\mathcal{X}|\mathcal{Y}, \theta_Y, \theta_X) = \frac{P(\mathcal{Y}|\mathcal{X}, \theta_Y)P(\mathcal{X}|\theta_X)}{\sum_{\mathcal{X}} P(\mathcal{Y}|\mathcal{X}, \theta_Y)P(\mathcal{X}|\theta_X)} \quad (1)$$

Here θ_Y is the set of parameters for the data likelihood density function $P(\mathcal{Y}|\mathcal{X}, \theta_Y)$ and θ_X is the set of parameters for the density function of $\mathcal{X}$ (i.e. $P(\mathcal{X}|\theta_X)$). We define the set of parameters for θ_X and θ_Y in Sections 2.3 and 2.5 respectively.

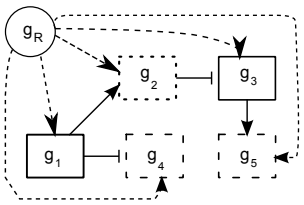
We obtain an assignment of the $\mathcal{X}$, θ_X and θ_Y after the optimization of Equation 1. Using these hidden variable assignments and the observed dataset we calculate the posterior probability of all X_{ij} variables given by $P(X_{ij}|\mathcal{X} - X_{ij}, \mathcal{Y}, \theta_X, \theta_Y)$. Using this posterior probability, we quantify the contribution of one DE gene on its outgoing neighbor for being DE.

In order to evaluate the joint distribution of $\mathcal{X}$ (i.e. Equation 1) we have to define the prior density function $P(\mathcal{X}|\theta_X)$ and the data likelihood function $P(\mathcal{Y}|\mathcal{X}, \theta_Y)$. We also have to define the formulation of the posterior probability of X_{ij}. We discuss these in the following sections.

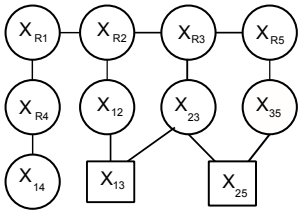
2.3 Computation of the prior density function

In this section, we describe how we build the prior density function $P(\mathcal{X}|\theta_X)$. We build this distribution based on two assumptions that follow from the underlying biological problem.

1. Each gene can affect the expressions of its outgoing neighbors.
2. The metagene g_R (i.e. external perturbation) can affect the expression of every other gene.



(a) Perturbation experiment



(b) Markov random field graph

Fig. 2. (a). A small hypothetical gene network with perturbation. The circle g_R represents the abstraction of the external perturbation i.e. g_R. Rectangles denote genes. $\rightarrow$ implies activation and $\dashv$ implies inhibition. The dotted arrow from g_R indicates potential effect on each genes. The directly impacted DE genes g_1 and g_3 are denoted by solid rectangle. Dashed rectangles g_4 and g_5 imply secondarily impacted DE genes. Dotted rectangle is for the EE gene g_2. **(b).** The graph for Markov random field created from the hypothetical gene network in (a). For each neighbor pair we create a circular node. We create three rectangular nodes that do not correspond to any neighbor pair, however they are part of the MRF graph. Two nodes are connected with an undirected edge if they share a subscript at same position. For example, node X_{R4} and X_{14} are connected as they share 4 at second position.

The first assumption follows from the fact that each gene can activate or inhibit its outgoing neighbors. So, if the activity of a gene is altered, the effect can propagate to its outgoing neighbors. The second assumption is evident as the external perturbation such as radiation can change the activity of any of the genes.

Under these assumptions, we observe that the X_{ij} variables can depend on each other. To capture this dependency we create a graph called Markov Random Field (MRF) graph over the $\mathcal{X}$ variables that encompasses the two above mentioned assumptions. This graph represents the probabilistic dependency between the genes in the extended gene network.

The MRF graph is an undirected graph $\mathcal{G} = (\mathcal{X}, \mathcal{E})$, where $X = \{X_{ij}\}$ variables represent the vertices of the graph. We denote the set of edges with $\mathcal{E} = \{(X_{ij}, X_{kj})|W_{ki} = W_{ij} = 1\} \cup \{(X_{ij}, X_{ip})|W_{jp} = W_{ij} = 1\}$. For example, in Figure 2(b) X_{12} and X_{13} are neighbors as in Figure 2(a) g_1 interacts with g_2 and g_2 interacts with g_3. Notice that, this graph represents the probabilistic dependency between the genes in the $\mathcal{X}$ domain, not in the domain of genes. So, the dependency of two neighbors are captured by an edge between two $\mathcal{X}$ variables. For example, in Fig 2(b) we draw the MRF graph corresponding to the hypothetical gene network in Figure 2(a). In the gene network, there is an edge from g_1 to g_4. So, g_1 can possibly alter the state of g_4. We have an edge between X_{R1} and X_{R4} that corresponds to the edge from g_1 to g_4. As R is common for X_{R1} and X_{R4}, if they attain the same value it implies that the genes g_1 and g_4 are in same state (i.e. DE or EE).

We employ MRF over $\mathcal{E}$ to evaluate this dependency among genes. We first describe how we build MRF over the MRF graph. Let us denote the neighbors of X_{ij} in the MRF graph as $X^*_{ij} = \{X_{kj}|W_{ki} = 1\} \cup \{X_{ip}|W_{jp} = 1\}$. In a Markov random field, a

$$p(X_{ij}|\mathcal{X}-X_{ij},\theta_X)=\frac{exp(\gamma_1\zeta(X_{ij})+\gamma_2\sum_{t\in\{0,2,3\}}f(X_{ij},t)+\gamma_3\sum_{k,W_{ki}=1}f(X_{ij},X_{kj})+\gamma_4\sum_{p,W_{jp}=1}f(X_{ij},X_{ip}))}{\sum_{X_{ij}\in\{0,1,2,3\}}exp(\gamma_1\zeta(X_{ij})+\gamma_2\sum_{t\in\{0,2,3\}}f(X_{ij},t)+\gamma_3\sum_{k,W_{ki}=1}f(X_{ij},X_{kj})+\gamma_4\sum_{p,W_{jp}=1}f(X_{ij},X_{ip}))} \quad (2)$$

Fig. 3. The equation for the conditional likelihood function.

node is independent of other nodes given its neighbors. Formally, $p(X_{ij}|\mathcal{X}-\{X_{ij}\},\theta_X)=p(X_{ij}|X^*_{ij},\theta_X)$. Using Hammersley-Clifford theorem [4, 18], we express the joint distribution of $\mathcal{X}$ in MRF as the product of potential functions in a way that it represents the conditional independence of the X_{ij} variables. More specifically, we express $p(X|\theta_X)$ as the product of the potential functions defined over the cliques in the graph $\mathcal{G}$, divided by a partition function Z as, $p(\mathcal{X}|\theta_X)=\frac{1}{Z}\prod_{C_{ij},W_{ij}=1}\psi(C_{ij})$. Here, C_{ij} is a clique in the MRF and $\psi(C_{ij})$ is a potential function over C_{ij} respectively. We create a clique C_{ij} over the variable X_{ij} and its neighbors X^*_{ij} when W_{ij} =1. To limit the complexity of our model, we consider only cliques of size one and two. We normalize the probability distribution with $Z=\sum_{\mathcal{X}}\prod_{C_{ij}}\psi(X_{ij})$.

The potential functions capture the dependency of the variables in the MRF graph. So, we incorporate our assumptions about the genetic network in the potential functions. The potential functions consist of feature functions. Each feature function takes two integers as input and produces a binary integer as output. Here, we define each feature function as $f(v_1,v_2)$ =1 if v_1 = v_2 and 0 otherwise.

The optimization of MRF assigns weights to the feature functions according to their actual prevalence in the MRF graph. In the following paragraphs, we discuss the proposed feature functions.

1. We define two feature functions for the set of singleton cliques. These two feature functions capture the two events, when $X_{ij}=1$ and $X_{ij}\neq 1$.We write a feature function $f(X_{ij},1)$ which equals to 1 when X_{ij} = 1 and 0 otherwise. For notational convenience, we also denote this feature function by $\zeta(X_{ij})$ with the same semantics as that of $f(X_{ij},1)$. We define another feature function $f(X_{ij},t)$, where $t\in\{0,2,3\}$, which equals to 1 when X_{ij} equals to t and 0 otherwise. We define two separate feature functions to capture these two events, as their frequencies are not equal.
2. Let us consider a sequence of four genes g_1, g_2, g_3 and g_5 in Figure 2(a). Consider the X_{23} variable in the MRF graph that consists of the states of g_2 and g_3. X_{13} is a neighbor of X_{23} in MRF graph as g_1 is an incoming neighbor of g_3 in the gene network. Similarly, X_{25} is a neighbor of X_{23} as g_5 is an outgoing neighbor of g_3. So, if S_1 equals to S_2 then X_{23} = X_{13}. Similarly if S_3 equals to S_5 then X_{23} = X_{25}. We capture these events in two feature functions for X_{ij} based on the incoming neighbors of g_i and the outgoing neighbors of g_j.

   - **Incoming neighbors of g_i:** Let us denote the incoming neighbors of g_i with $In(g_i)$. We write a feature function $f(X_{kj},X_{ij})$, $\forall k$, $g_k\in In(g_i)$. $f(X_{kj},X_{ij})$ = 1 if S_i = S_k and W_{ki} = W_{ij} = 1. Otherwise, $f(X_{kj},X_{ij})$ = 0.
   - **Outgoing neighbors of g_j:** Let us denote the outgoing neighbors of g_j as $Out(g_j)$. We define a feature function $f(X_{ip},X_{ij})$, $\forall p$, $g_p\in Out(g_j)$. $f(X_{ip},X_{ij})$ = 1 if S_p = S_j and W_{jp} = W_{ij} = 1. Otherwise, $f(X_{ip},X_{ij})$ = 0.

In the last two feature functions, X_{kj} or X_{ip} may not represent any interactions from the extended gene network when W_{kj} = 0 or W_{ip} = 0 respectively. We represent them by rectangles in Figure 2(b).

After embedding these feature functions in the potential functions the joint prior distribution of $\mathcal{X}$ is,

$$\begin{aligned}p(\mathcal{X}|\theta_X)=\frac{1}{Z}exp(&\gamma_1\sum_{i,j,W_{ij}=1}\zeta(X_{ij})+\gamma_2\sum_{i,j,W_{ij}=1,t\in\{0,2,3\}}f(X_{ij},t)\\&+\gamma_3\sum_{i,j,k,W_{ij}=1,W_{ki}=1}f(X_{ij},X_{kj})\\&+\gamma_4\sum_{i,j,p,W_{ij}=1,W_{jp}=1}f(X_{ij},X_{ip}))\end{aligned} \quad (3)$$

Here we denote $\{\gamma_1,\gamma_2,\gamma_3,\gamma_4\}$ as the MRF parameters θ_X.

In the next section, we discuss how we define the objective function with respect the MRF. We also describe how we formulate the posterior probability density function for X_{ij}.

2.4 Approximation of the objective function

To optimize the objective function in Equation 1, we need to evaluate the prior density function $p(\mathcal{X}|\theta_X)$. In Equation 3, we defined this function as a combination of feature functions. This requires estimation of θ_X. Also, a maximum likelihood estimation of θ_X requires evaluation of the partition function Z in Equation 3. Evaluation of Z is intractable even for a small number of X_{ij}, as the number of terms in the summation is exponential in the number of X_{ij}. We use an approximate formula to solve this problem. A standard approximation scheme is pseudo-likelihood [5], where the objective function is the simple product of the conditional likelihood function of the X_{ij} variables. Geman et al. proved the consistency of the maximum pseudo-likelihood estimate [14].

From Equation 3 the conditional likelihood of a node X_{ij} is given by Equation 2 in Figure 3. We refer the reader to the Appendix I for the derivation of Equation 2.

In this framework of pseudo-likelihood, we can approximate the objective function of Equation 1 as the product of posterior density function of X_{ij} and optimize it using the ICM [6] algorithm given θ_X. We consider those X_{ij} when W_{ij} =1, as for other X_{ij} variables, g_i and g_j are not correlated. Thus, our objective function becomes,

$$F=\text{argmax}_{\mathcal{X}}(\prod_{i,j}F_{ij}) \quad (4)$$

We derive the posterior density function F_{ij} of X_{ij} as,

$$\begin{aligned}F_{ij}&=p(X_{ij}|\mathcal{X}-X_{ij},\theta_X,\theta_Y)\\&=\frac{p(Y_i,Y_j|X_{ij},X^*_{ij},\theta_Y)p(X_{ij}|\mathcal{X}-X_{ij},\theta_X)}{\sum_{X_{ij}\in\{0,1,2,3\}}p(Y_i,Y_j|X_{ij},X^*_{ij},\theta_Y)}\end{aligned} \quad (5)$$

We present the derivation of F_{ij} in Appendix III due to page limitation.

There are two different terms in objective function of Equation 4. We already discussed one, the conditional likelihood density function in Equation 2. We discuss the other, the data likelihood density function $p(Y_i, Y_j|X_{ij}, X^*_{ij}, \theta_Y)$ in the next section.

2.5 Calculation of likelihood density function

The next challenge in computing the posterior density function, is to define the data likelihood function (i.e. the function at the denominator of Equation 5). In this section, we describe how we derive this function.

We start by making a mild assumption that the expressions of a gene in its control or non-control groups follow normal distribution. Note that we can rewrite the derivation below with minor changes when the expressions of genes follow another distribution. To keep the discussion brief we will use normal distribution.

When a gene is equally expressed, all the data points in both control and non-control groups share a latent mean [25]. For a DE gene, the shared latent means in the two groups are different. So, for a DE gene the control and non-control groups should follow separate normal distribution. Let us denote a set of measurements for a gene g_i by $\mathbf{z_i} = \{z_{i1}, z_{i2}, \cdots, z_{iN}\}$ that follows a single Gaussian distribution. Let us denote the latent mean of $\mathbf{z_i}$ as μ and the standard deviation as σ. As different genes can have different average expression, we assume that μ follows a genome wise distribution [25] with mean μ_0 and standard deviation τ. μ varies between the control and non-control groups of a DE gene. Thus, for $\mathbf{z_i}$, the joint likelihood for the data points in that group is given by,

$$\begin{aligned}\mathcal{L} &= \int [\prod_{i=1}^{n} \mathcal{N}(z_i|\mu, \sigma^2)]\mathcal{N}(\mu|\mu_0, \tau^2)d\mu \\ &= \frac{\sigma}{(\sqrt{2\pi}\sigma)^n\sqrt{n\tau^2+\sigma^2}} exp(-\frac{\sum_i z_i^2}{2\sigma^2} - \frac{\mu_0^2}{2\tau^2})\cdot \\ &\quad exp(\frac{\frac{\tau^2 n^2 \bar{z}^2}{\sigma^2} + \frac{\sigma^2\mu_0^2}{\tau^2} + 2n\bar{z}\mu_0}{2(n\tau^2+\sigma^2)})\end{aligned} \tag{6}$$

The reader can find the derivation of Equation 6 at Demichelis et al [12].

If a gene is DE, its expression measurements in control and non-control groups follow separate distributions. On the other hand, for equally expressed genes, all the measurements in both the groups share the same mean.The joint data likelihood for a DE gene is given by,

$$\mathcal{L}_{\text{DE}}(g_i) = p(\mathbf{y_i}|\mu_0, \sigma^2, \tau^2)p(\mathbf{y_i}^{'}|\mu_0, \sigma^2, \tau^2) \tag{7}$$

Similarly, for EE genes it is given by,

$$\mathcal{L}_{\text{EE}}(g_i) = p(\mathbf{y_i} \cup \mathbf{y_i}^{'}|\mu_0, \sigma^2, \tau^2) \tag{8}$$

Now we are ready to derive the joint likelihood distribution for different values of X_{ij}. Let us denote the set of parameters $\{\mu, \sigma, \tau\}$ by θ_Y.

Case 1. ($X_{ij} = 1$) In this case, both g_i and g_j are DE. We define the neighbors of S_i by $S_i^* = \{S_k|X_{ik}, X_{ki} \in X^*_{ij}\}$. We substitute X_{ij} with the set $\{S_i, S_j\}$, where S_i and S_j denote the states of gene g_i and g_j respectively. We substitute $X_{ij} \cup X^*_{ij}$ by $\{S_i, S_j, S_i^*, S_j^*\}$. So, for X_{ij} = 1, $p(Y_i, Y_j|X_{ij} = 1, X^*_{ij}, \theta_Y) = p(Y_i|S_i = DE, \theta_Y)p(Y_j|S_j = DE, \theta_Y) = \mathcal{L}_{\text{DE}}(g_i)\mathcal{L}_{\text{DE}}(g_j)$.

Case 2. ($X_{ij} = 0$) Here, both g_i and g_j are EE. As earlier, we substitute $\{X_{ij}, X^*_{ij}\}$ with $\{S_i, S_j, S_i^*, S_j^*\}$. From Equation 8 we obtain the likelihood density function as, $p(Y_i, Y_j|X_{ij} = 0, X^*_{ij}, \theta_Y) = \mathcal{L}_{\text{EE}}(g_i)\mathcal{L}_{\text{EE}}(g_j)$.

Case 3. (X_{ij} = 2 or X_{ij} = 3) In this case only one of the genes is DE. From Equation 7 and 8 for X_{ij} =2 we obtain the likelihood density function as, $p(Y_i, Y_j|X_{ij} = 2, X^*_{ij}, \theta_Y) = \mathcal{L}_{\text{DE}}(g_i)\mathcal{L}_{\text{EE}}(g_j)$. Similarly for X_{ij} =3 we obtain the likelihood density function as, $p(Y_i, Y_j|X_{ij} = 3, X^*_{ij}, \theta_Y) = \mathcal{L}_{\text{EE}}(g_i)\mathcal{L}_{\text{DE}}(g_j)$ We defer the detailed derivation to the Appendix IV.

A special case arises when g_i is the metagene, i.e. g_R. We assume that $\mathcal{L}_{\text{DE}}(g_R)$ = 1 and $\mathcal{L}_{\text{EE}}(g_R)$ = 0. Thus, the likelihood of the metagene given than S_R is DE equals to 1 and 0 otherwise.

2.6 Objective function optimization

So far, we have described how we compute the posterior density function. The final challenge is to find the values of the hidden variables that maximize the objective function (Equation 4). We develop an iterative algorithm to address this challenge.

At each iteration, we first estimate the hyper-parameters of joint conditional density function and joint likelihood density function based on the estimated value of $\mathcal{X}$ in the previous iteration. Next, based on the estimated hyper-parameters, we estimate $\mathcal{X}$.

The joint likelihood density function is non-convex in terms of the parameters $\theta_Y = \{\mu_0, \sigma, \tau\}$. Also, the joint conditional density is non-convex in terms of $\theta_X = \{\gamma_1, \gamma_2, \gamma_3, \gamma_4\}$. We use a global optimization method called differential evolution [37] to optimize both of them. To optimize the objective function we employ the ICM algorithm described by Besag [6]. Briefly, our iterative algorithm works as follows.

1. Obtain an initial estimate of $\mathcal{S}$ variables. In our implementation we use student's t-test assuming the data follows normal distribution. We use 5% confidence interval for this purpose.
2. Estimate parameters θ_Y that maximizes the joint data likelihood function,

$$\begin{aligned}&\prod_{X_{ij}, W_{ij}=1} p(Y_i, Y_j|X_{ij}, X^*_{ij}, \theta_Y) \\ &= \prod_{\nu\in\{0,1,2,3\}} \prod_{X_{ij}=\nu, W_{ij}=1} p(Y_i, Y_j|A, X_{ij} = \nu, \theta_Y)\end{aligned}$$

   We implement this step using Differential Evolution, which is similar to genetic algorithm.
3. Calculate an estimate of parameters θ_X that maximizes the conditional prior density function by $\prod_{X_{ij}, W_{ij}=1} p(X_{ij}|\mathcal{X} - \{X_{ij}\}, \theta_X)$. We also implement this step using Differential Evolution.
4. Carry out a single cycle of ICM using the current estimate of $\mathcal{S}$, θ_X and θ_Y. For all S_i, maximize $\prod_{X_{mn}} p(X_{mn}|\mathcal{X} - X_{mn}, \mathcal{Y}, \theta_X, \theta_Y)$ when $X_{mn} \in \{X_{i*}, X_{*i}\}$ and W_{mn} =1.
5. Go to step 2 for a fixed number of cycles or until $\mathcal{X}$ converges to a certain predefined value.

We optimize the objective function in terms of the S_i ($1 \leq i \leq M$) variables instead of X_{ij} variables. So, in step 4, we maximize the product of the conditional density functions of all the X_{ij} variables given by $\prod_{X_{mn}} p(X_{mn}|\mathcal{X} - X_{mn}, \mathcal{Y}, \theta_X, \theta_Y)$, $X_{mn} \in \{X_{ij}|i = m \text{ or } j = n\}$, $W_{mn} = 1$.

Table 1. List of top 25 genes that are mostly affected by external perturbation. The dataset was generated using 10 Gy ionizing radiation over immortalized B cells obtained from 155 members of 15 Centre d'tude du Polymorphisme Humain (CEPH) Utah pedigrees [11]. Genes are tabulated row-wise, in increasing order of ranking.

PGF	IL8RB	FOSL1	F2R	PPM1D
MDM2	CDKN1A	TNC	PLXNB2	EPHA2
DDB2	TP53I3	PLK1	TNFSF9	ADRB2
MAP3K12	JUN	SORBS1	LRDD	MDM2
SDC1	MYC	PRKAB1	EI24	DDIT4

3 EXPERIMENTS

In this section we discuss the experiments we conducted to evaluate the quality of our method. We implemented our method in MATLAB and Java. We obtained the code of Differential Evolution from the `http://www.icsi.berkeley.edu/~storn/code.html`. We compared our method with SSEM [9] as SSEM is one of the most recent methods that can be used to solve the problem considered in this paper. We obtained SSEM from `http://gardnerlab.bu.edu/SSEMLasso`. We ran our code on a cluster that consists of AMD Opteron 2.4 Ghz and Intel Core 2 Duo 2.4 Ghz with 4GB memory on every machine.

Dataset We use the data set collected by Smirnov et al. [34] for the real microarray data. The dataset was generated using 10 Gy ionizing radiation over immortalized B cells obtained from 155 members of 15 Centre d'tude du Polymorphisme Humain (CEPH) Utah pedigrees [11]. Microarray snapshots were obtained at 0th hour (i.e., before the radiation) and 2 and 6 hours after the radiation. We adapt the time series data to create the control and non-control data for our experiments. We use the data before radiation as control data. For the non-control data we calculate the expected expressions of a gene at each points after the radiation. We select the one with higher absolute difference from the expected expression of control data for that gene.

We also collect 24,663 genetic interactions from the 105 regulatory and signaling pathways of KEGG database [24]. Overall 2,335 genes belong to at least one pathway in KEGG. We consider only the genes that take part in the gene networks in our model.

3.1 Evaluation of biological significance

In this experiment, we investigate the biological significance of the genes that our method detects as the primarily affected ones. We train our method on the dataset collected by Smirnov et al. [34]. After optimization we rank the genes in descending order of the data likelihood of the DE genes with the perturbation metagene. We tabulate top 25 genes from the rank in Table 1.

Nine out of the ten highest ranked genes have significant biological evidence that they are impacted by radiation. Imaoka et al. [21] compared the gene expression between normal mammary glands to spontaneous and γ-radiation induced cancerous glands of rat. The PGF (parental growth factor) gene showed differential expression in both spontaneous and irradiated carcinomas. Nagtegaal et al. [30] applied radiation to human rectal adenocarcinoma and compared the gene response to that of normal tissues. The cytokines and receptor IL8RB showed differential expression between normal and irradiated rectal tissues. Amundson et al. [1] administered γ-radiation to p-53 wild type ML-1 human myeloid cell line. FOSL1 (known by FRA1 that time) showed differential expression as the stress response. Lin et al. [26] applied ionizing radiation on human lymphoblastoid cells. F2R, a coagulation factor II receptor, was upregulated in that experiment. Jen et al. [23] investigated the effect of ionizing radiation on the transcriptional response of lymphoblastoid cells in time series microarray experiments. PPM1D, a gene related to DNA repair, showed response to both 3Gy and 10Gy radiation. Wu et al. [39] conducted a high dose UV radiation experiment to observe the relation between MDM2 gene on p53 gene. Their experiment revealed that initially both protein and mRNA level of MDM2 increases in a p53 independent manner, which clearly substantiated the direct effect of radiation on MDM2. Jakob et al. [22] irradiated human fibroblasts with accelerated lead ions. Confocal microscopy discovered a single, bright focus of CDKN1A protein in the nuclei of human fibroblast within 2 minutes after radiation. Rieger et al. [32] applied both ultra violet and infrared radiation on fifteen human cell lines and observed that PLXNB2 was up-regulated for both kind of radiations. Zhang et al. [40] reported that EPHA2 worked as an essential mediator of UV-radiation induced apoptosis.

The above discussion demonstrates that our method can find the primarily affected genes from a perturbation microarray data accurately.

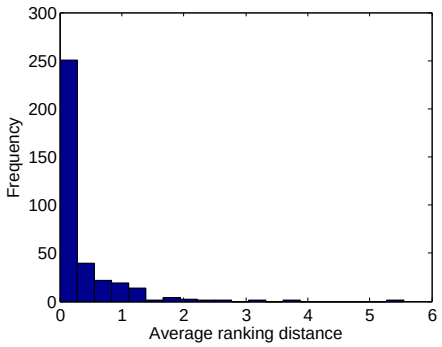


Fig. 4. Frequency of average distance of rankings over training and testing data. The figure shows that the difference is very close to zero. This suggests that our method can rank the probabilistic effect of the incoming neighbors of the genes with great precision. The average difference between the ranks obtained in the training and the testing data is less than one position in 92.7% of the cases.

3.2 Evaluation of the rankings of neighbor genes

Recall that our goal is to find the primarily affected genes. We achieve this objective by computing the probability of each gene, including the metagene to contribute the alterations in the expression of every other gene. In this experiment, we evaluate our success in terms of how accurately we rank the contribution probabilities of the genes as follows.

We divide the dataset of 155 samples into training and testing set in 2:1 ratio. For each DE gene, we sort its incoming DE neighbors in decreasing order of their data likelihood probability with respect to the outgoing neighbor. For example, let us assume g_1 is DE. It has four incoming DE neighbors g_2, g_3, g_4 and g_R where g_R is the metagene. Let NL_{ij} denotes the normalized likelihood function $\frac{p(Y_i,Y_j|\{X_{ij}\},X^*_{ij},\theta_Y)}{\sum_{X_{ij}\in\{0,1,2,3\}} p(Y_i,Y_j|X_{ij},X^*_{ij},\theta_Y)}$ of X_{ij}. For instance, If $NL_{R1} \geq NL_{41} \geq NL_{21} \geq NL_{31}$, then the sorted list is $\{g_R, g_4, g_2, g_3\}$. We denote the sorted list as a ranking of the incoming DE neighbors. Let us denote the position of a gene g_i in the ranking of g_j for training data $\rho_{g_j}(g_i)$. We create another set of rankings from the

testing data likelihood probability. Let us denote the position of g_i in the ranking of g_j from testing data by $\rho'_{g_j}(g_i)$. For a gene g_j we define the *average ranking distance* between training and testing data as $\delta(g_j) = \frac{\sum_{g_i \in IN(g_j)} abs(\rho_{g_j}(g_i) - \rho'_{g_j}(g_i))}{|IN(g_j)|}$, where $IN(g_j)$ is the set of incoming DE neighbors for g_i, abs(.) denotes the absolute value and $|IN(g_j)|$ stands for the cardinality of $IN(g_j)$.

We calculated the average ranking distance for all the genes that have incoming neighbors apart from the metagene. We repeat the experiments three times with a different set of training and testing data. We create a histogram for the average differences from the three experiments in Figure 4.

Figure 4 shows that the difference is very close to zero. This suggests that our method can rank the probabilistic effect of the incoming neighbors of the genes with great precision. The average difference between the ranks obtained in the training and the testing data is less than one position in 92.7% of the cases. This implies that our method can accurately identify the primary cause of DE genes.

3.3 Evaluation of the accuracy of our method

The experiments over the real dataset suggest the validity of our model. Two questions however follow from these experiments. (1) What are the limitations of our method? In other words, when does our method work accurately? (2) Does our method distinguish the primary affected genes from the others?

To answer these questions we conducted experiments on synthetic datasets to observe the performance of our method in a controlled manner.

Synthetic data generation We generate the data in the presence of a hypothetical perturbation to simulate the real dataset. We use the gene network derived from KEGG first to select a random gene from the network and denote it as a primarily affected DE gene. We traverse the ancestors in a breadth fast manner. For each of the ancestor, we made it a secondarily affected DE gene with a probability of $1 - (1 - q)^{\eta}$, where η is the number of incoming DE neighbors. Here q is the probability that a gene is DE due to a DE predecessor. We repeat these steps to create the desired number of primarily affected genes. After the classification of the genes we create control and non-control data for each of them for over N patients. We first create data for the control group for a gene with mean μ_c and standard deviation σ. We sample the mean from another Gaussian distribution $N(\mu_c|\mu_0, \tau)$. For EE gene both the groups follow the same distribution. For DE gene we separate the mean μ_{nc} of gene expression between control and non-control group by a fixed amount. We keep the separation in primarily affected genes higher than that of the secondarily affect genes. In our experiment q = 0.4, N = 200, μ_0 = 7, τ = 2.9 and σ = 0.87.

Detection of primarily affected genes In this experiment our goal is to detect the primarily affected genes by the perturbation using the synthetic dataset. After optimization, we rank all the DE genes in descending order of the normalized data with respect to metagene g_R. Let the set of true primarily affected genes be PA. Let RG be the set of first $|PA|$ genes from the rank, where $|PA|$ is the cardinality of PA. We define accuracy as $\frac{|PA \cap RG|}{|RG|}$.

In our experiments we keep μ_c fixed at σ and vary difference $|\mu_{nc} - \mu_c|$ between 0 to σ to produce different dataset. We calculate the accuracy for each of the $|\mu_{nc} - \mu_c|$ and plot the calculated accuracy. We repeat the entire experiments twice with $|PA|$ = 10 and 54. The total number of DE genes is 159 and 400 respectively.

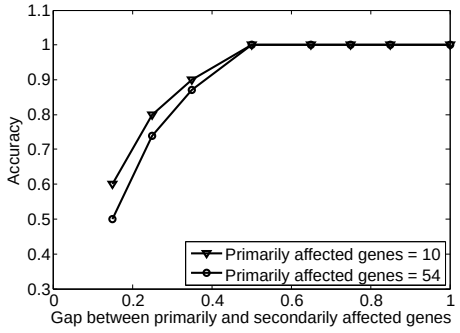


Fig. 5. Accuracy of our method over synthetic dataset. The X axis is in terms of σ. For instance, 0.4 means that the difference between the mean of control and non-control groups of a gene is 0.4 $\times\sigma$. The figure shows that we reach 100% accuracy with even a difference of $0.5 \times \sigma$. Compared to the standard deviation of noise (i.e. σ) of the data this difference is quite small. This testifies that our model is efficient to differentiate between primarily and secondarily affected genes with very small difference of the two groups.

Figure 5 shows that we reach 100% accuracy with even a difference of $0.5 \times \sigma$. Compared to the standard deviation of noise (i.e. σ) of the data this difference is quite small. This testifies that our model is efficient to differentiate between primarily and secondarily affected genes with very small difference of the two groups. Also, the number of total DE genes were 159 and 400 respectively, which were quite high compared to the number of corresponding primarily affected genes (i.e. 10 and 54 respectively). So, our method is able to eliminate most of the Secondarily impacted genes in most of the cases. As the number of primarily affected genes decrease from 54 to 10 the accuracy of our method goes higher.

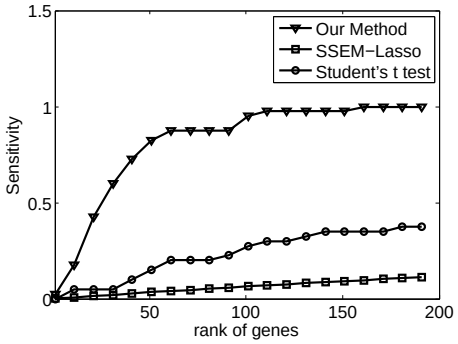
3.4 Comparison to other methods

In this section, we compare the accuracy of our method to that of SSEM and a simpler method *Student's t test*.

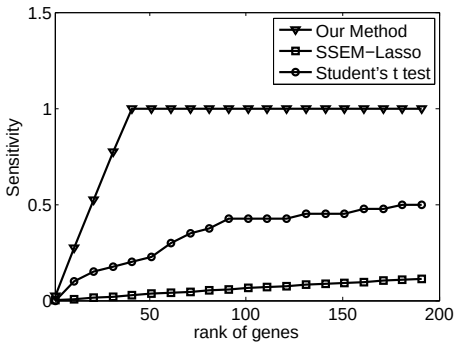
Synthetic data generation: We simulated real perturbation events to prepare synthetic data with known primarily and secondarily affected genes in a controlled setting. We generated this dataset by simulating the real dataset to maximum possible extent. We used the gene networks from KEGG database. First we decide on a random set of primarily and secondarily affected genes based on the steps described in Section 3.3. We use the control part of the real dataset in Smirnov et al. [34] as the control part of our synthetic dataset. To generate the non-control dataset, we traverse each of the genes that participate in the gene networks. Suppose, for a gene g_i, the mean and standard deviation of its expression in the control dataset are given by μ_{i_c} and σ_{i_c} respectively. If the gene is EE we generate its non-control data points from the a normal distribution given by the parameters (μ_{i_c}, $\sigma^2_{i_c}$). If the gene is DE, we use the same variance as that of the control group. However, we use a different mean. For the primarily and secondarily affected genes we use $\mu_{i_c} \pm d_p$ and $\mu_{i_c} \pm d_s$ respectively, where $d_p > d_s$.

To summarize, we use the same variance in the non-control group as that in the control group. However, for an affected gene we change the value of the mean in the non-control group from that in the control group. For a primarily affected gene we applied a higher deviation of mean than that of the secondarily affected genes.

Experimental setup: Given an input dataset, using each of the three methods, we ranked all the genes. Highly ranked genes have higher chance of being a primarily affected gene according to each method. We explain how we do the ranking in the following.



(a) Gap = 0.2 $\times\sigma$



(b) Gap = 0.6 $\times$ σ

Fig. 6. Comparison of our method to SSEM and t-test. The number of primarily affected genes is 50. The gap between the mean of primarily affected and secondarily affected genes are 0.2 to 0.6 $\times\sigma$, where sigma is estimated from the real dataset. The figures indicate that our method outperforms SSEM and t-test.

- **Our method:** We sort the genes in decreasing order of joint likelihood with the metagene. A higher joint likelihood implies a higher chance of being primarily affected.
- **SSEM:** We train SSEM on the control dataset, where it learns the correlation between the genes. We test SSEM on the non-control dataset, where it produces a rank for each single data point.
- **Student's t test:** We used the function called $ttest$ from MATLAB. We apply it on every individual gene, where it takes control and non-control dataset as input and produces a p-value as output. By default, null hypothesis is that "the differences of two input data set are a random sample from a normal distribution with mean 0 and unknown variance, against the alternative that the mean is not 0". Thus, the null hypothesis corresponds that the gene is EE. So a substantially lower p-value implies a higher chance of being primarily affected. We performed the test on all the genes and rank them according the increasing order of p-values.

Let us assume the set of primarily affected genes as PG and first k elements of the ranking as RG_k. We define the sensitivity of the ranking at position k by $\eta_k = \frac{|PG \cap RG_k|}{|PG|}$. Thus, a higher value of η_k denotes a higher sensitivity. We prepare a sensitivity vector $\{\eta_1, \eta_2, \cdots \eta_{|R|}\}$, by arraying the sensitivity of a ranking at all the positions of the ranks. Here, $|R|$ denotes the cardinality of the ranking. For SSEM we obtain a sensitivity vector for every data points in the non-control dataset. We create a consolidated sensitivity vector by averaging them.

Results: We conducted experiments by for $\frac{d_s - d_p}{\sigma} = \{0.2, 0.4, 0.6, 0.8, 1.0, 1.2, 1.5, 1.75\}$, number of primarily affected genes = $\{10, 50\}$ and number of data points = $\{10, 20, 40, 60, 80, 100, 125, 155\}$. Here, σ corresponds to the standard deviation of the expressions of genes in the dataset. However, due to space limitation we discuss only two of them in this paper (see Figure 6). The results we discuss correspond to the case when we have 40 primarily affected genes and 155 data points. The results of the other experiments are similar to those in Figure 6(b).

Figures 6(a)and 6(b) show the sensitivity of the three methods when $(d_s - d_p) = 0.2 \times\sigma$ and 0.6 $\times\sigma$ respectively. The former one corresponds to the computationally harder case as the difference between the control and non-control datasets is small. As the gap between d_s and d_p increases identifying primarily affected genes becomes easier.

Form the figure, we observe that our method is significantly more accurate than the other two methods for all datasets consistently. It reaches 100% accuracy (i.e., it can find all the 50 primarily affected genes) in the top 150 ranked genes in when the gap is small and in the top 50 genes as the gap increases to 0.6 $\times\sigma$. The results were similar for larger gap values (results not shown). The t test reaches around 40% and 50% sensitivity at 200 ranking position respectively. SSEM's sensitivity is below 0.25 for all experiments even within the top 200 positions.

We believe that there are two major factors for the success of our method over the competing methods among other. First, our method can successfully incorporate the gene interactions using MRFs while others ignore this informations. Second, our method is capable of dealing with both large and small number of primarily affected genes while other methods' performance deteriorates as this number grows. In real perturbation experiments, often multiple genes are primarily affected. Thus, we conclude that our method is suitable for real perturbation experiments.

4 CONCLUSION

In this paper, we considered the problem of identifying primarily affected genes in the presence of an external effect that can perturb the expressions of genes. We assumed that we were given the expression measurements of a set of genes before and after the application of an external perturbation. We developed a new probabilistic method to quantify the cause of differential expression of each gene. Our method considers the possible gene interactions in regulatory and signaling networks, for a large number of perturbations. It uses a Bayesian model with the help of Markov Random Fields to capture the dependency between the genes. It also provides the underlying distribution of the impact with confidence interval.

Our experiments on both real and synthetic datasets demonstrated that our method could find primarily affected genes with high accuracy. Our method was 100% accurate when the difference between expected expressions of primarily and secondarily affected genes is at least half of the standard deviation of the gene expressions. It achieved significantly better accuracy than two competing methods, namely SSEM and the student's t test method.

Our experiments suggest that our method is applicable to real applications as it works well when the number of primarily affected genes grows. SSEM can be more appropriate for small scale experiments, where a few genes are primarily affected. When

confronted with a dataset with higher number of affected genes and multiple replications, SSEM is negatively impacted by the variance of the genes over those replications. Hence, it fails to model the patterns that exists over the entire dataset The efficiency of our method justifies the use of gene networks in our methods.

Our method produces a probability distribution rather than a fixed binary decision. The major advantage of this approach is that it augment every decision with a range, and hence endows it with a confidence. A distribution is most of the time more useful, as is it models the very stochastic nature of gene interactions.

REFERENCES

[1]SA. Amundson, M. Bittner, and Y. Chen et al. Fluorescent cDNA microarray hybridization reveals complexity and heterogeneity of cellular genotoxic stress responses. *Oncogene*, 18(24):3666–72, 1999.

[2]Ferhat Ay, Fei Xu, and Tamer Kahveci. Scalable steady state analysis of boolean biological regulatory networks. *PloS one*, 4(12), 2009.

[3]KA. Baggerly, KR. Coombes, and KR. Hess et al. Identifying differentially expressed genes in cDNA microarray experiments. *J Comput Biol*, 8(6):639–59, 2001.

[4]Julian Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society*, 36(2):192–236, 1974.

[5]Julian Besag. Efficiency of pseudolikelihood estimation for simple Gaussian fields. *Biometrika*, 64(3):616–618, 1977.

[6]Julian Besag. On the Statistical Analysis of Dirty Pictures. *Journal of the Royal Statistical Society*, 48(3):259–302, 1986.

[7]Christopher M. Bishop. *Pattern Recognition and Machine Learning* . Springer, 2007.

[8]RY. Cheng, A. Zhao, and WG. Alvord et al. Gene expression dose-response changes in microarrays after exposure of human peripheral lung epithelial cells to nickel(II). *Toxicol Appl Pharmacol*, 191(1):22–39, 2003.

[9]EJ. Cosgrove, Y. Zhou, TS. Gardner, and ED. Kolaczyk. Predicting gene targets of perturbations via network-based filtering of mRNA expression compendia. *Bioinformatics*, 24(21):2482–90, 2008.

[10]J. Courcelle, A. Khodursky, and B. Peter et al. Comparative gene expression profiles following UV exposure in wild-type and SOS-deficient Escherichia coli. *Genetics*, 158(1):41–64, 2001.

[11]J. Dausset and Others. Centre d'etude du polymorphisme humain (CEPH): collaborative genetic mapping of the human genome. *Genomics*, 6:575–577, 1990.

[12]F. Demichelis, P. Magni, and P. Piergiorgi et al. A hierarchical Nave Bayes Model for handling sample heterogeneity in classification problems: an application to tissue microarrays. *BMC Bioinformatics*, 7:514, 2006.

[13]D. di Bernardo, MJ. Thompson, and TS. Gardner et al. Chemogenomic profiling on a genome-wide scale using reverse-engineered gene networks. *Nat Biotechnol*, 23(3):377–83, 2005.

[14]S. Geman and C. Graffigne. Markov random field image models and their applications to computer vision. *Proceedings of the International Congress of Mathematics: Berkley*, pages 1496–1517, 1987.

[15]G. Giaever, P. Flaherty, and J. Kumm et al. Chemogenomic profiling: identifying the functional interactions of small molecules in yeast. *Proc Natl Acad Sci U S A*, 101(3):793–8, 2004.

[16]G. Giaever, DD. Shoemaker, and TW. Jones et al. Genomic profiling of drug sensitivities via induced haploinsufficiency. *Nat Genet*, 21(3):278–83, 1999.

[17]D. Hamelinck, H. Zhou, and L. Li et al. Optimized normalization for antibody microarrays and application to serum-protein profiling. *Mol Cell Proteomics*, 4(6):773–84, 2005.

[18]JM. Hammersley and P. Clifford. Markov fields on finite graphs and lattices. *Unpublished manuscript*, 1971.

[19]TR . Hughes, MJ. Marton, and AR. Jones et al. Functional discovery via a compendium of expression profiles. *Cell*, 102(1):109–126, 2000.

[20]T. Ideker, V. Thorsson, and JA. Ranish et al. Integrated genomic and proteomic analyses of a systematically perturbed metabolic network. *Science*, 292(5518):929–34, 2001.

[21]T. Imaoka, S. Yamashita, and M. Nishimura et al. Gene expression profiling distinguishes between spontaneous and radiation-induced rat mammary carcinomas. *J Radiat Res (Tokyo)*, 49(4):349–60, 2008.

[22]B. Jakob, M. Scholz, and G. Taucher-Scholz. Immediate localized CDKN1A (p21) radiation response after damage produced by heavy-ion tracks. *Radiat Res*, 154(4):398–405, 2000.

[23]KY. Jen and VG. Cheung. Transcriptional response of lymphoblastoid cells to ionizing radiation. *Genome Res*, 13(9):2092–100, 2003.

[24]M. Kanehisa and S. Goto. KEGG: kyoto encyclopedia of genes and genomes. *Nucleic Acids Res*, 28(1):27–30, 2000.

[25]CM. Kendziorski, MA. Newton, H. Lan, and MN. Gould. On parametric empirical Bayes methods for comparing multiple groups using replicated gene expression profiles. *Stat Med*, 22(24):3899–914, 2003.

[26]R. Lin, Y. Sun, and C. Li et al. Identification of differentially expressed genes in human lymphoblastoid cells exposed to irradiation and suppression of radiation-induced apoptosis with antisense oligonucleotides against caspase-4. *Oligonucleotides*, 17(3):314–26, 2007.

[27]PY. Lum, CD. Armour, and SB. Stepaniants et al. Discovering Modes of Action for Therapeutic Compounds Using a Genome-Wide Screen of Yeast Heterozygotes. *Cell*, 116(1):5–7, 2004.

[28]MJ. Marton, JL. DeRisi, and HA. Bennett et al. Drug target validation and identification of secondary drug target effects using DNA microarrays. *Nat Med*, 4(11):1293–301, 1998.

[29]GL. Miklos and R. Maleszka. Microarray reality checks in the context of a complex disease. *Nat Biotechnol*, 22(5):615–21, 2004.

[30]ID. Nagtegaal, CG. Gaspar, and LT. Peltenburg et al. Radiation induces different changes in expression profiles of normal rectal tissue compared with rectal carcinoma. *Virchows Arch*, 446(2):127–35, 2005.

[31]AB. Parsons, RL. Brost, and H. Ding et al. Integration of chemical-genetic and genetic interaction data links bioactive compounds to cellular target pathways. *Nat Biotechnol*, 22(1):62–9, 2004.

[32]KE. Rieger and G. Chu. Portrait of transcriptional responses to ultraviolet and ionizing radiation in human cells. *Nucleic Acids Res*, 32(16):4786–803, 2004.

[33]Yishai Shimoni, Gilgi Friedlander, and Guy Hetzroni et al. Regulation of gene expression by small non-coding RNAs: a quantitative view. *Mol Syst Biol.*, 3:138, 2007.

[34]DA. Smirnov, M. Morley, and E. Shin et al. Genetic analysis of radiation-induced changes in human gene expression. *Nature*, 459(7246):587–91, 2009.

[35]Bin Song, I. Esra Buyuktahtakin, Sanjay Ranka, and Tamer Kahveci. Manipulating the steady state of metabolic pathways . *IEEE TCBB*.

[36]Le Song, Mladen Kolar, and Eric P. Xing. KELLER: estimating time-varying interactions between genes. *Bioinformatics*, 25(12):i128–136, June 2009.

[37]R. Storn and K. Price. Differential Evolution A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *Journal of Global Optimization*, 11(4):341–359, 1997.

[38]KK. Tsai, EY. Chuang, JB. Little, and ZM. Yuan. Cellular mechanisms for low-dose ionizing radiation-induced perturbation of the breast tissue microenvironment. *Cancer Res*, 65(15):6734–44, 2005.

[39]L. Wu and AJ. Levine. Differential regulation of the p21/WAF-1 and mdm2 genes after high-dose UV irradiation: p53-dependent and p53-independent regulation of the mdm2 gene. *Mol Med*, 3(7):441–51, 1997.

[40]G. Zhang, CN. Njauw, JM. Park, C. Naruse, M. Asano, and H. Tsao. EphA2 is an essential mediator of UV radiation-induced apoptosis. *Cancer Res*, 68(6):1691–6, 2008.

APPENDIX

I CONDITIONAL LIKELIHOOD

From Equation 3 the conditional likelihood of a node X_{ij} is,

$$p(X_{ij}|\mathcal{X}-X_{ij},\theta_X)=\frac{exp(\gamma_1\zeta(X_{ij})+\gamma_2\sum_{t\in\{0,2,3\}}f(X_{ij},t)+\gamma_3\sum_{k,W_{ki}=1}f(X_{ij},X_{kj})+\gamma_4\sum_{p,W_{jp}=1}f(X_{ij},X_{ip}))}{\sum_{X_{ij}\in\{0,1,2,3\}}exp(\gamma_1\zeta(X_{ij})+\gamma_2\sum_{t\in\{0,2,3\}}f(X_{ij},t)+\gamma_3\sum_{k,W_{ki}=1}f(X_{ij},X_{kj})+\gamma_4\sum_{p,W_{jp}=1}f(X_{ij},X_{ip}))} \tag{9}$$

We provide the derivation of Equation 9 in Appendix II. As the number of incoming and outgoing neighbors can vary for a gene we modify version of Equation 2 to normalize γ_3 and γ_4 as,

$$p(X_{ij}|\mathcal{X}-X_{ij},\theta_X)=\frac{exp(\gamma_1\zeta(X_{ij})+\gamma_2\sum_{t\in\{0,2,3\}}f(X_{ij},t)+\gamma_3\frac{\sum_{k,W_{ki}=1}f(X_{ij},X_{kj})}{|N_1(X_{ij})|}+\gamma_4\frac{\sum_{p,W_{jp}=1}f(X_{ij},X_{ip})}{|N_2(X_{ij})|})}{\sum_{X_{ij}\in\{0,1,2,3\}}exp(\gamma_1\zeta(X_{ij})+\gamma_2\sum_{t\in\{0,2,3\}}f(X_{ij},t)+\gamma_3\frac{\sum_{k,W_{ki}=1}f(X_{ij},X_{kj})}{|N_1(X_{ij})|}+\gamma_4\frac{\sum_{p,W_{jp}=1}f(X_{ij},X_{ip})}{|N_2(X_{ij})|})} \tag{10}$$

Here, $N_1(X_{ij})=\{X_{kj}|W_{ki}=1\}$ and $N_2(X_{ij})=\{X_{ip}|W_{jp}=1\}$ are the set of neighbors for X_{ij} and $|N_1(X_{ij})|$ is the cardinality of $N_1(X_{ij})$.

II DERIVATION OF $P(X_{IJ}|\mathcal{X}-\{X_{IJ},\theta_X\})$, W_{IJ} =1 :

$$\begin{aligned}
&p(X_{ij}|\mathcal{X}-X_{ij},\theta_X)\\
&=\frac{p(\mathcal{X},\theta_X)}{P(\mathcal{X}-X_{ij},\theta_X)}\\
&=\frac{p(\mathcal{X},\theta_X)}{\sum_{X_{ij}\in\{0,1,2,3\}}P(\mathcal{X}-X_{ij},X_{ij},\theta_X)}\\
&=\frac{A(X_{ij}).B}{(A(0)+A(1)+A(2)+A(3)).B}
\end{aligned} \tag{11}$$

Here, $A(X_{ij})$ is $exp(\gamma_1\zeta(X_{ij})+\gamma_2\sum_{t\in\{0,2,4\}}f(X_{ij},t)+\gamma_3\sum_{k,W_{ki}=1}f(X_{ij},X_{kj})+\gamma_4\sum_{p,W_{jp}=1}f(X_{ij},X_{ip}))$. B is given by $exp(\gamma_1\sum_{mn\neq ij,W_{mn}=1}\zeta(X_{mn})+\gamma_2\sum_{mn\neq ij,W_{mn}=1,t\in\{0,2,4\}}f(X_{mn},t)+\gamma_3\sum_{q,ij\neq mn,W_{mn}=1,W_{qm}=1}f(X_{mn},X_{qn})$
$+\gamma_4\sum_{l,ij\neq mn,W_{mn}=1,W_{nl}=1}f(X_{mn},X_{ml}))$. We cancel B from numerator and denominator and the density function simplifies to,

$$\begin{aligned}
&f(X_{ij},1)p(X_{ij}|\mathcal{X}-X_{ij},\theta_X)\\
&=\frac{exp(\gamma_1\zeta(X_{ij})+\gamma_2\sum_{t\in\{0,2,4\}}f(X_{ij},t)+\gamma_3\sum_{k,W_{ki}=1}f(X_{ij},X_{kj})+\gamma_4\sum_{p,W_{jp}=1}f(X_{ij},X_{ip}))}{\sum_{X_{ij}\in\{0,1,2,3\}}exp(\gamma_1\zeta(X_{ij})+\gamma_2\sum_{t\in\{0,2,4\}}f(X_{ij},t)+\gamma_3\sum_{k,W_{ki}=1}f(X_{ij},X_{kj})+\gamma_4\sum_{p,W_{jp}=1}f(X_{ij},X_{ip}))}
\end{aligned} \tag{12}$$

Here, we denote the prior density parameters $\{\gamma_1,\gamma_2,\gamma_3,\gamma_4\}$ by θ_X.

III DERIVATION OF F_{IJ} :

$$\begin{aligned}
&F_{ij}\\
&=p(X_{ij}|\mathcal{X}-X_{ij},\mathcal{Y},\theta_X,\theta_Y)\\
&=p(X_{ij}|\mathcal{X}-X_{ij},Y_i,Y_j,\theta_X,\theta_Y)\\
&=\frac{p(Y_i,Y_j,\mathcal{X}-X_{ij}-X^*_{ij},X_{ij},X^*_{ij},\theta_X,\theta_Y)}{p(Y_i,Y_j,\mathcal{X}-X_{ij}-X^*_{ij},X^*_{ij},\theta_X,\theta_Y)}\\
&=\frac{p(Y_i,Y_j,\mathcal{X}-X_{ij}-X^*_{ij}|X_{ij},X^*_{ij},\theta_X,\theta_Y)p(X_{ij},X^*_{ij},\theta_X,\theta_Y)}{p(Y_i,Y_j,\mathcal{X}-X_{ij}-X^*_{ij}|X^*_{ij},\theta_X,\theta_Y)p(X^*_{ij},\theta_X,\theta_Y)}\\
&=\frac{p(Y_i,Y_j|X_{ij},X^*_{ij},\theta_X,\theta_Y)p(\mathcal{X}-X_{ij}-X^*_{ij}|X_{ij},X^*_{ij},\theta_X,\theta_Y)p(X_{ij},X^*_{ij},\theta_X,\theta_Y)}{p(Y_i,Y_j|X^*_{ij},\theta_X,\theta_Y)p(\mathcal{X}-X_{ij}-X^*_{ij}|X^*_{ij},\theta_X,\theta_Y)p(X^*_{ij},\theta_X,\theta_Y)}\\
&=\frac{p(Y_i,Y_j|X_{ij},X^*_{ij},\theta_X,\theta_Y)p(\mathcal{X}-X_{ij}-X^*_{ij},X_{ij},X^*_{ij},\theta_X,\theta_Y)}{p(Y_i,Y_j|X^*_{ij},\theta_X,\theta_Y)p(\mathcal{X}-X_{ij}-X^*_{ij},X^*_{ij},\theta_X,\theta_Y)}\\
&=\frac{p(Y_i,Y_j|X_{ij},X^*_{ij},\theta_X,\theta_Y)p(\mathcal{X},\theta_X,\theta_Y)}{p(Y_i,Y_j|X^*_{ij},\theta_X,\theta_Y)p(\mathcal{X}-X_{ij},\theta_X,\theta_Y)}
\end{aligned}$$

$$
\begin{aligned}
&= \frac{p(Y_i, Y_j|X_{ij}, X^*_{ij}, \theta_X, \theta_Y)p(\mathcal{X}, \theta_X)p(\theta_Y)}{p(Y_i, Y_j|X^*_{ij}, \theta_X, \theta_Y)p(\mathcal{X} - X_{ij}, \theta_X)p(\theta_Y)} \\
&= \frac{p(Y_i, Y_j|X_{ij}, X^*_{ij}, \theta_X, \theta_Y)p(\mathcal{X}, \theta_X)}{p(Y_i, Y_j|X^*_{ij}, \theta_X, \theta_Y)p(\mathcal{X} - X_{ij}, \theta_X)} \\
&= \frac{p(Y_i, Y_j|X_{ij}, X^*_{ij}, \theta_X, \theta_Y)p(X_{ij}|\mathcal{X} - X_{ij}, \theta_X)p(\mathcal{X} - X_{ij}, \theta_X)}{p(Y_i, Y_j|X^*_{ij}, \theta_X, \theta_Y)p(\mathcal{X} - X_{ij}, \theta_X)} \\
&= \frac{p(Y_i, Y_j|X_{ij}, X^*_{ij}, \theta_X, \theta_Y)p(X_{ij}|\mathcal{X} - X_{ij}, \theta_X)}{p(Y_i, Y_j|X^*_{ij}, \theta_X, \theta_Y)} \\
&= \frac{p(Y_i, Y_j, X_{ij}, X^*_{ij}, \theta_X, \theta_Y)p(X^*_{ij}, \theta_X, \theta_Y)p(X_{ij}|\mathcal{X} - X_{ij}, \theta_X)}{p(X_{ij}, X^*_{ij}, \theta_X, \theta_Y)p(Y_i, Y_j, X^*_{ij}, \theta_X, \theta_Y)} \\
&= \frac{p(Y_i, Y_j, \theta_X|X_{ij}, X^*_{ij}, \theta_Y)p(X_{ij}, X^*_{ij}, \theta_Y)p(X^*_{ij}, \theta_X, \theta_Y)p(X_{ij}|\mathcal{X} - X_{ij}, \theta_X)}{p(X_{ij}, X^*_{ij}, \theta_X, \theta_Y)p(Y_i, Y_j, \theta_X|X^*_{ij}, \theta_Y)p(X^*_{ij}, \theta_Y)} \\
&= \frac{p(Y_i, Y_j|X_{ij}, X^*_{ij}, \theta_Y)p(\theta_X|X_{ij}, X^*_{ij}, \theta_Y)p(X_{ij}, X^*_{ij}, \theta_Y)p(X^*_{ij}, \theta_X, \theta_Y)p(X_{ij}|\mathcal{X} - X_{ij}, \theta_X)}{p(Y_i, Y_j|X^*_{ij}, \theta_Y)p(X_{ij}, X^*_{ij}, \theta_X, \theta_Y)p(\theta_X|X^*_{ij}, \theta_Y)p(X^*_{ij}, \theta_Y)} \\
&= \frac{p(Y_i, Y_j|X_{ij}, X^*_{ij}, \theta_Y)p(X_{ij}, X^*_{ij}, \theta_X, \theta_Y)p(X^*_{ij}, \theta_X, \theta_Y)p(X_{ij}|\mathcal{X} - X_{ij}, \theta_X)}{p(Y_i, Y_j|X^*_{ij}, \theta_Y)p(X_{ij}, X^*_{ij}, \theta_X, \theta_Y)p(X^*_{ij}, \theta_X, \theta_Y)} \\
&= \frac{p(Y_i, Y_j|X_{ij}, X^*_{ij}, \theta_Y)p(X_{ij}|\mathcal{X} - X_{ij}, \theta_X)}{p(Y_i, Y_j|X^*_{ij}, \theta_Y)} \\
&= \frac{p(Y_i, Y_j|X_{ij}, X^*_{ij}, \theta_Y)p(X_{ij}|\mathcal{X} - X_{ij}, \theta_X)}{\sum_{X_{ij} \in \{0,1,2,3\}} p(Y_i, Y_j|X_{ij}, X^*_{ij}, \theta_Y)}
\end{aligned}
\tag{13}
$$

In step 2 of the derivation, we substitute $\mathcal{Y}$ by Y_i and Y_j as X_{ij} is independent of all Y_k such that $k \neq i$ and $k \neq j$.

IV DERIVATION OF LIKELIHOOD DENSITY FUNCTIONS:

In this section we derive the data likelihood density functions. We use Equation 7 and Equation 8 as the data likelihood function for DE and EE respectively in these derivations. We start with proving that that Y_i and Y_j are independent of S^*_i and S^*_j given S_i and S_j. We represent the parameters $\{\mu_0, \sigma, \tau\}$ by θ_Y.

$$
\begin{aligned}
&p(Y_i, Y_j|S_i, S_j, S^*_i, S^*_j, \theta_Y) \\
&= \frac{p(Y_i, Y_j, S_i, S_j, S^*_i, S^*_j, \theta_Y)}{p(S_i, S_j, S^*_i, S^*_j, \theta_Y)} \\
&= \frac{p(Y_i, Y_j, S^*_i, S^*_j|S_i, S_j, \theta_Y)p(S_i, S_j, \theta_Y)}{p(S_i, S_j, S^*_i, S^*_j, \theta_Y)} \\
&= \frac{p(Y_i, Y_j|S_i, S_j, \theta_Y)p(S^*_i, S^*_j|S_i, S_j, \theta_Y)p(S_i, S_j, \theta_Y)}{p(S_i, S_j, S^*_i, S^*_j, \theta_Y)} \\
&= \frac{p(Y_i, Y_j|S_i, S_j, \theta_Y)p(S_i, S_j, S^*_i, S^*_j, \theta_Y)}{p(S_i, S_j, S^*_i, S^*_j, \theta_Y)} \\
&= p(Y_i, Y_j|S_i, S_j, \theta_Y)
\end{aligned}
\tag{14}
$$

We use this conditional independence in the following proofs.

Case 1. $X_{ij} = 1$ We can replace X_{ij} by the set $\{S_i, S_j\}$, where S_i and S_j denote the states of g_i and g_j. We define the neighbors of S_i by $S^*_i = \{S_k|X_{ik}, X_{ki} \in X^*_{ij}\}$. Thus, we can replace $X_{ij} \bigcup X_{ij*}$ by $\{S_i, S_j, S^*_i, S^*_j\}$. So, for X_{ij} = 1, we write,

$$
\begin{aligned}
&p(Y_i, Y_j|X_{ij} = 1, X^*_{ij}, \theta_Y) \\
&= p(Y_i, Y_j|S_i = DE, S_j = DE, S^*_i, S^*_j, \theta_Y) \\
&= p(Y_i, Y_j|S_i = DE, S_j = DE, \theta_Y) \\
&= p(Y_i|S_i = DE)p(Y_j|S_j = DE, \theta_Y) \\
&= \mathcal{L}_{\text{DE}}(g_i)\mathcal{L}_{\text{DE}}(g_j)
\end{aligned}
\tag{15}
$$

Case 2. $X_{ij} = 0$ As earlier, we shall replace $\{X_{ij}, X^*_{ij}\}$ by $\{S_i, S_j, S^*_i, S^*_j\}$.

$$\begin{aligned}
&p(Y_i, Y_j | X_{ij} = 0, X^*_{ij}, \theta_Y) \\
&= p(Y_i, Y_j | S_i = EE, S_j = EE, S^*_i, S^*_j, \theta_Y) \\
&= p(Y_i, Y_j | S_i = EE, S_j = EE, \theta_Y) \\
&= p(Y_i | S_i = EE, \theta_Y) p(Y_j | S_j = EE, \theta_Y) \\
&= \mathcal{L}_{\text{EE}}(g_i) \mathcal{L}_{\text{EE}}(g_j)
\end{aligned} \tag{16}$$

Case 3. $X_{ij} = 2$ In a similar fashion we can can write that,

$$\begin{aligned}
&p(Y_i, Y_j | X_{ij} = 2, X^*_{ij}, \theta_Y) \\
&= p(Y_i, Y_j | S_i = DE, S_j = EE, S^*_i, S^*_j, \theta_Y) \\
&= p(Y_i, Y_j | S_i = DE, S_j = EE, \theta_Y) \\
&= p(Y_i | S_i = DE) p(Y_j | S_j = EE) \\
&= \mathcal{L}_{\text{DE}}(g_i) \mathcal{L}_{\text{EE}}(g_j)
\end{aligned} \tag{17}$$

Case 4. $X_{ij} = 3$ Similarly,

$$\begin{aligned}
&p(Y_i, Y_j | X_{ij} = 3, X^*_{ij}, \theta_Y) \\
&= p(Y_i, Y_j | S_i = EE, S_j = DE, S^*_i, S^*_j, \theta_Y) \\
&= p(Y_i, Y_j | S_i = EE, S_j = DE, \theta_Y) \\
&= p(Y_i | S_i = EE) p(Y_j | S_j = DE, \theta_Y) \\
&= \mathcal{L}_{\text{EE}}(g_i) \mathcal{L}_{\text{DE}}(g_j)
\end{aligned} \tag{18}$$