# Synthetic data generation with probabilistic Bayesian Networks

Grigoriy Gogoshin[*1a], Sergio Branciamore[1b], and Andrei S. Rodin[*1c]

[*]Corresponding authors

[1]Department of Computational and Quantitative Medicine, Beckman Research Institute, and Diabetes and Metabolism Research Institute, City of Hope National Medical Center, 1500 East Duarte Road, Duarte, CA 91010 USA

[a]ggogoshin@coh.org

[b]sbranciamore@coh.org

[c]arodin@coh.org

**Abstract**

Bayesian Network (BN) modeling is a prominent and increasingly popular computational systems biology method. It aims at constructing probabilistic networks from the big heterogeneous biological data that reflect the underlying networks of biological relationships. Currently, there is a variety of strategies for evaluating the BN methodology performance, ranging from utilizing the artificial benchmark datasets and models, to the specialized biological benchmark datasets, to the simulation studies that generate synthetic data from the predefined network models. The latter is arguably the most comprehensive approach; however, existing implementations are typically limited by their reliance on the SEM (structural equation modeling) framework, with many explicit and implicit assumptions that might not be realistic in a typical biological data analysis scenario. In this study, we develop an alternative, purely probabilistic, simulation framework that is a more appropriate fit with the real biological data and biological network models. In conjunction, we also expand on our current understanding of the theoretical notions of causality and dependence / conditional independence in BNs and Markov Blankets within.

## 1 Introduction

A descriptive / predictive modeling activity that is especially suited to the systems biology approach to the data analysis (of big heterogeneous data) is dependency modeling, wherein a biological network, or a graphical representation of a biological model (genotypes, phenotypes, metabolites, endpoints, other -omics variables and their interrelationships), is constructed from the ”flat” data. (Subsequently, specific components of biological networks can be evaluated using traditional / parametric statistical criteria, and sub-networks can be selected for further biological hypothesis generation and testing). This activity is also known as causal discovery (inference) in graphical models. Bayesian Networks (BNs)-based dependency modeling is an established computational biology tool that has been rapidly gaining acceptance in big biological data analysis (Branciamore et al. 2018; Cooper et al. 2015; Gogoshin, Boerwinkle, and Rodin 2017; Jiang, Barmada, and Visweswaran. 2010; Lan et al. 2016; Neapolitan, Xue, and Jiang 2014; Needham et al. 2007; Pe'er 2005; Piatetsky-Shapiro and Tamayo 2003; Qi, Li, and Cheng 2014; Rodin et al. 2005; Rodin et al. 2012; Sedgewick et al. 2019; Sherif, Zayed, and Fakhr 2015; Wang, Audenaert, and Michoel 2019; Yin et al. 2015; Zeng, Jiang, and Neapolitan 2016; Ziebarth, Bhattacharya, and Cui 2013; Zhang and Shi 2017; Zhang et al. 2017; Zhang et al. 2014). Comprehensive treatments of BN methodology, and probabilistic networks (PNs) in general, can be found in numerous textbooks (Pearl 1988; Pearl 2009; Russell and Norvig. 2009; Spirtes, Glymour, and Scheines 2000) and reviews (Daly, Shen, and Aitken 2011; Glymour, Zhang, and Spirtes 2019; Heckerman, Geiger, and Chickering 1995; Heckerman 1995; Spirtes and Zhang 2016; Zhang et al. 2018).

Most of the recent work in the field has been aimed at improving the BN modeling scalability, handling mixed data types (including various types of biological data), incorporating latent variables, and developing more robust software interfaces and visualization options (Andrews, Ramsey, and Cooper 2018; Andrews, Ramsey, and Cooper 2019; Chen et al. 2019; Hong et al. 2018; Jabbari et al. 2017; Ogarrio, Spirtes, and J 2016; Raghu et al. 2018; Ramsey et al. 2017; Sedgewick et al. 2016; Spirtes and Zhang 2016; Xing et al. 2017; Xing et al. 2018; Yu, Liu, and Li 2019; Zhang et al. 2018; Zhang et al. 2019). In view of these developments, the ability to objectively assess the performance of BN modeling is of the utmost importance. Currently, there are three principal venues for accomplishing this: (i) using well-established (in machine learning community) predefined benchmark models/datasets, such as Alarm or Asia, (ii) using various specialized biologically-oriented benchmark datasets, both real and simulated, such as DREAM (Eicher et al. 2019; Wang, Audenaert, and Michoel 2019; Xing et al. 2017), and (iii) developing more-or-less realistic simulation frameworks (Andrews, Ramsey, and Cooper 2018; Han et al. 2017; Ellis and Wong 2008; Isci et al. 2011; Tasaki et al. 2015; Zhang et al. 2019). The first approach is necessarily limited and does not generalize to the modern high-dimensional biological data. The later two strategies have their obvious pros and cons; a useful discussion can be found in (Wang, Audenaert, and Michoel 2019). In this study, we concentrate on the third approach, namely constructing robust, generalizable, and mathematically rigorous simulation frameworks. Currently, these frameworks typically involve (i) specifying the synthetic network structures, and (ii) utilizing linear SEM (structural equation modeling) methodology to generate the actual synthetic data (Isci et al. 2011; Tasaki et al. 2015; Zhang et al. 2019).

In the broader context of heterogeneous biological data and networks, it is important to distinguish between the notions of causality, correlation, association and dependence / conditional independence. Establishing causality (Cooper et al. 2015; Glymour, Zhang, and Spirtes 2019; Pearl 2009; Zhang et al. 2018) is usually perceived as the ultimate goal in the biological network analyses (Ramsey et al. 2017; Zeng, Jiang, and Neapolitan 2016; Zhang et al. 2019). However, establishing (and even defining) causality is often equivocal, for both theoretical (Glymour, Zhang, and Spirtes 2019; Zhang et al. 2018) and practical/biological reasons (e.g., ambiguity of the causal interpretation of the relationship between two correlated gene expression measurements, or two single nucleotide polymorphisms in linkage disequilibrium) (Tasaki et al.

2015). Of course, there are advantages to viewing certain biological network models (or their parts) through the lens of directional causality: first, directional causality often fits the ground-truth biological reality well (to use an obvious example, genotypes ”cause” phenotypes, but not the other way around); second, existing BN algorithmic machinery largely relies on the concept of DAGs, or directed acyclic graphs. And yet, at the most basic level (e.g., physical / chemical interaction of two biological molecules), it is unclear if it is possible, or even desirable, to impose a directional causation label on what is essentially a dependence relationship that we attempt to infer from observing a statistical correlation in the joint probability distribution. The concept of ”cause - effect” may have more to do with rationalization in seeking implicational chains rather than reflecting the totality of any given interaction. Therefore, we have chosen to frame this study predominantly around the notion of dependence / conditional independence, which is a measure of relevance that needs not invoke causality considerations to make inferences about the biological reality (however, ancestor/descendant relationships in DAGs are taken into consideration if and when necessary).

Regardless of whether we are focusing on causation or dependence relationships, SEM might not be the most appropriate tool for generating synthetic data in the biological BN context. Leaving aside the issues of acyclicity (and directionality in general), and assumptions of linearity and normality (which can be dealt with, if imperfectly), there is a fundamental issue of observability. Indeed, consider a typical biological dataset subject to BN modeling — real biological datasets include only comparatively small sets of variables which are observable under the confines of specific biomedical studies (experiments). These variables cannot be expected to be in any particular categorizable relationship known or suspected a priori. The most one can postulate about a real dataset is that the variables in it may be dependent or independent, and certainly not that these variables belong to a carefully selected group of key components adequately describing dynamics of a biological system/model in question. Therefore, synthetic datasets generated via SEM are fundamentally different from the real biological datasets. Instead, we propose using PNs as the general abstraction for the synthetic data generation purposes. Such an abstraction, built around an analytically defined distribution induced by a given graphical model coupled with the random model structure generation, is immediately interpretable as the joint probability distribution, while simultaneously featuring inferential characteristics encoded in conditional independencies. Of course, building a simulation framework around PNs in a mathematically rigorous fashion is not a trivial undertaking. Below we detail the considerations involved in building such a framework (Methods, Proofs and Results: Sections 2-3), ascertain the congruence between the ”forward” process (simulations) and ”inverse” process (actual reconstruction of the networks and Markov Blankets within from the data) in Methods, Proofs and Results: Section 4, and discuss simulation sample size considerations in Results: Section 5.

## 2 Methods, Proofs and Results: Synthetic data generation with probabilistic networks

Synthetic data generation is a ”forward problem” of sampling from a distribution described by the model under consideration. (Similarly, ”inverse problem” is the BN modeling per se, i.e., reconstructing a BN from the observed data).

Since we are predominantly interested in performance testing of probabilistic inference over Markov Blankets of particular variables in the network, as well as in assessing the quality of extraction of structural parameters of a model from the data induced by it, the data synthesis algorithm should be able to generate samples from the arbitrarily complex models.

The analytically defined distribution induced by a given model has a factorizable joint probability that is no more complex than the basic chain rule, but given the set of further simplifications dictated by conditional independencies encoded in the model in question, the complexity of the factorization may be significantly lower than the worst case scenario.

In the most reduced form the chain factorization of the joint probability of a given model with an appropriate indexing of variables is

$$P(\bigcap_{i} x_i) = P(\bigcap_{i>0} x_i)P(x_0|\bigcap_{i>0} x_i) = P(\bigcap_{i>1} x_i)P(x_1|\bigcap_{i>1})P(x_0|\bigcap_{i>0} x_i) = \ldots = \prod_{i} P(x_i|\bigcap_{j>i} x_j) \tag{1}$$

This product of conditional probabilities that represents the joint probability for the model provides a way to generate sample components serially from local conditional probabilities instead of direct sampling from the n-dimensional joint distribution, which is an important consideration, because populating low probability fine event structures by direct sampling is difficult given a finite sample size. In fact, this difficulty grows with the dimensionality of the model, so that as the number of variables grows, many joint events quickly become intractable even for binary variables. Therefore, being able to rely on local sampling of conditional probability distributions is a key factor in successfully synthesizing components of an adequate n-dimensional sample that actually reflects the fine structure of the joint distribution. If, however, care is not taken to obtain such a level of resolution in the joint distribution, then the resulting synthetic sampling is likely to run into source uniqueness problems; i.e. the same data could be obtained from different source distributions, making the source distinction all but impossible.

In order to utilize the resolution provided by the expansion above, we need to find a way to methodically obtain conditional distributions for the dependent variables. Consider elementary mechanics for discretely distributed variables (we will only look at binary distributions, but the results readily generalize to higher arity):

In the linear algebra sense, the joint distribution of two variables $x$ and $y$ can be represented as a matrix of joint events, and is readily decomposed by the chain law into an inner product of conditional distribution and unconditional distribution, as follows:

$$P(x \cap y) = \begin{pmatrix} P(x_0 \cap y_0) & P(x_0 \cap y_1) \\ P(x_1 \cap y_0) & P(x_1 \cap y_1) \end{pmatrix} = \begin{pmatrix} P(x_0|y_0)P(y_0) & P(x_0|y_1)P(y_1) \\ P(x_1|y_0)P(y_0) & P(x_1|y_1)P(y_1) \end{pmatrix} = P(x|y) \cdot \mathrm{diag}(P(y)) \tag{2}$$

where individual events are notated as indexed variables and $P(y) = (P(y_0), P(y_1))^T$.

The marginalization operation for the lexicographic order provided above is given by

$$P(x) = P(x \cap y) \cdot \mathbf{1} = P(x|y) \cdot P(y) \tag{3}$$

and

$$P(y) = P(x \cap y)^T \cdot \mathbf{1} = P(y \cap x) \cdot \mathbf{1} = P(y|x) \cdot P(x) \tag{4}$$

where $\mathbf{1} = (1, 1)^T$. Note that here the change of the order of arguments in the joint probability operator implies the transposition operation, i.e. $P(x \cap y)^T = P(y \cap x)$.

We can now derive the condition necessary for generating conditional distributions for *dependent* variables. A formal criterion for dependence is obtained by negation of the independence criterion

$$\exists y_i \quad (P(x|y = y_i) \neq P(x)) \tag{5}$$

i.e. the existence of at least one dependent event is sufficient to establish dependence.

However, the unconditional distribution of $x$, that will necessarily be dependent on $y$, is not given a priori. Furthermore, the above condition gives no prescription with regard to consistent random construction of the appropriate objects (or their placement in the formal probabilistic reasoning system).

This can be remedied by the derivation of a criterion for dependence which is agnostic about the unconditional distribution of $x$, since from the perspective of data synthesis $P(x|y)$ could be a matrix defined independently of $P(y)$, which would then induce $P(x \cap y)$ along with $P(x)$ through the chain rule relationship. A well-defined rule along these lines should (i) aid in automation of generation of conditional distributions, (ii) reduce the number of parameters needed to synthesize a random network to the number of nodes and their arity, and (iii) allow to span arbitrarily defined collections of random networks with relative ease (by direct sampling).

Consider the following observation

**Lemma 1.** *Let $x$ be a random variable with $n$ outcomes, then the columns of conditional distribution $P(x|y)$ are members of $n$-simplex $S \subset \mathbb{R}^{n+1}$.*

*Proof.* Trivially, the elements of any column of a conditional distribution $p(x|y)$ sum up to 1. Hence, since the variable $x$ has $n$ outcomes, all columns are members of the set defined by

$$S = \{(v_1, \ldots, v_n) : \sum_{i=1}^{n} v_i = 1, \quad v_i \geq 0, \quad \forall i\} \tag{6}$$

which is an $n$-simplex, and a subset of $\mathbb{R}^{n+1}$ □

To simplify the notation and to emphasize the fact that the conditional distribution $P(x|y)$ can be defined independently, let $D$ be an $n \times m$ linear operator whose columns are members of the $n$-simplex $S$, so that $P(x) = D \cdot P(y)$ given some $P(y)$.

**Lemma 2.** *If the column rank of $D$ is at least two, then $x$ defined by $P(x) = D \cdot P(y)$ is conditionally dependent on $y$.*

*Proof.* Suppose the rank of $D$ is at least two, then there is at least one linearly independent column, i.e.

$$\exists y_i, \quad \forall \alpha_k \geq 0, \quad \left( D_i \neq \sum_{k \neq i} D_k \alpha_k \right) \tag{7}$$

where $D_i$ is the $i$-th column of $D$. This implies that if for every index $k \neq i$ the coefficients are given by $\alpha_k = \frac{P(y_k)}{1-P(y_i)}$, then

$$D_i(1 - P(y_i)) \neq \sum_{k \neq i} D_k P(y_k) \tag{8}$$

Rearranging the terms yields

$$\exists y_i, \quad (D_i \neq D \cdot P(y)) \tag{9}$$

which is a restatement of the dependence criterion. Hence, there is at least one dependent event, making $x$ dependent on $y$. □

This lemma already guarantees that any matrix of appropriate size with rank of at least two should be sufficient to generate a conditionally dependent pair of variables. This fact, however, does not elucidate additional constraints an arbitrary conditional dependence may impose, and does little to provide control over the structure/nature of a synthetically constructed dependence relationship.

The following observations helps to rectify this lack of perspective:

**Lemma 3.** *Let $x$ be a random variable with $n$ outcomes, dependent on a random variable $y$ with $m$ outcomes; then*

- *The convex hull $C$ formed by the columns of $D$ has at least two vertices;*
- *Rank of $D$ must be at least two;*
- *$P(x)$ lies in the convex hull $C$ formed by the columns of $D$;*
- *The number of vertices of convex hull $C$ corresponds to the number of conditionally dependent events.*

*Proof.* Suppose $x$ is conditionally dependent on $y$, then

$$\exists y_i, \quad (D_i \neq D \cdot P(y)) \tag{10}$$

where $D_i$ is the $i$-th column of $D$, because $P(x) = D \cdot P(y)$. Rearranging the terms gives

$$D_i(1 - P(y = y_i)) \neq \sum_{k \neq i} D_k P(y = y_k) \tag{11}$$

for some arbitrary $P(y)$. Assuming that $P(y = y_i) \neq 1$, this yields

$$D_i \neq \sum_{k \neq i} D_k \alpha_k \tag{12}$$

with coefficients defined on the probability simplex characterized by

$$\sum_{k \neq i} \alpha_k = 1, \quad \alpha_k \geq 0, \forall k \neq i \tag{13}$$

In other words, in order for $x$ to be conditionally dependent on $y$, for any $D$ defined independently of $P(y)$, there must be at least one column in $D$ that must not be a convex combination of other columns. This column is, therefore, a vertex of a convex hull $C \subset S$ formed by all convex combinations of columns in $D$ characterized by

$$\sum_{i=1}^{m} D_i \alpha_i, \quad \sum_{i=1}^{m} \alpha_i = 1, \quad \alpha_i \geq 0, \forall i \tag{14}$$

The above implies the following:

- the convex hull contains more than one member, and, therefore, has at least two vertices;
- because there are at least two vertices, this polytope must be embedded in at least a 2-simplex, and, therefore, in at least $\mathbb{R}^2$, hence the the linear operator $D$ must be at least of rank two;
- the unconditional distribution $P(x)$ always lies in this convex hull, because $P(x) = D \cdot P(y)$ for any independently defined $D$ and $P(y)$;
- every conditionally dependent event of $y$ corresponds to a vertex of the convex hull that envelops the columns of $D$.

□

Some of the useful immediate consequences of the above can be summarized in the following corollary

**Corollary 1.** *Let $x$ be a random variable with $n$ outcomes and $y$ be a random variable with $m$ outcomes define by $P(x) = D \cdot P(y)$, then*

- *$x$ is conditionally dependent on $y$ if and only if the rank of the linear opeartor $D$ is at least two.*
- *Let $d$ be the number of dependent ancestor events that the linear operator $D$ of the size $n \times m$ and $rank = r$ encodes; then $r \leq d \leq m$.*
- *For a non-trivial pair $x$ and $y$, $D$ satifying $P(x) = D \cdot P(y)$ is not unique, and forms an equivalence class of geometrically similar constellations of points in the simplex $S$ of varying size induced by uniform scaling, and anchored at $x$.*

The obtained results suggest a concrete recipe for constructing random conditional dependence distributions in a formally sound way, while making sure that all relevant relationship properties are accounted for.

First, we observe that the columns of $D$ must be sampled from the n-simplex $S$. If sampled uniformly, this is equivalent to sampling from a Dirichlet distribution with the appropriate parameters. The objective of constructing random dependence relationships, however, has more to do with sampling polytopes of different shapes and locations from the simplex $S$ (see Corrolary 1 above), rather than its individual elements, something uniform sampling may not be able to address.

From this perspective, it may, after all, be more practical to first select a pair of unconditional distributions of $x$ and $y$, so that the convex hull corresponding to $D$ is anchored before the shape of the polytope is decided upon.

Second, we observe that the numerical structure of the linear operator $D$ is secondary to the properties of the convex hull formed by its columns, because it reflects the qualitative structure of the event space of the conditional dependence relationship. In particular, manipulation of the properties of this convex polytope allows to choose the proportion of dependent and independent events in the event space.

Conditional distributions of higher ancestral complexity, i.e. $P(x|\bigcap Y_k)$, can be treated the same way as a pairwise distribution $P(x|y)$, if we observe that $\bigcap Y_k$ can be viewed as a single variable over the combinatorically prescribed joint events. In this situation, the only specification that is different from the already considered scenario with the pairwise conditional distribution is that the joint events are not given *a priori*, but rather have to be derived as a function of the number and arity of ancestor variables in the DAG (for two binary ancestor variables, there will be 4 possible events, etc.)

Of some practical interest (to the data synthesis task), the minimum requirement for conditional dependence is that the rank of the linear operator is at least two under all circumstances (as stated in the Corollary 1). This observation makes realizing random dependency relationships, characterized by the minimal requirement only, seemingly very simple. It also further reinforces the notion (discussed in the preceding section) that probabilistic relationships are, by their very nature, fundamentally "generic" / implicit.

On the other hand, characterization as weakly constraining as that, leaves open the question as to how much of probabilistic dependency relationship landscape could be reliably covered relying on this device alone, and what is the most typical relationship that such a characterization might return.

Importantly, the observed correspondence between the geometrical object hidden in the linear operator $P(x|y)$, and conditional dependence relationship that it represents, fully addresses the raised concerns (this said, any developments that go beyond the introduction of the above ideas, such as uniform sampling of polytopes from the encompassing simplex, is out of scope of this work). We should emphasize here that our goal is not so much the "proper" generation of typical linear operators associated with conditional dependency, but rather the complete justification and technical tooling of the data synthesis method where, starting with the unconditional distribution of the set of root nodes, the whole joint probability distribution of the network can be sequentially generated, given any arbitrary DAG structure.

(Note that the DAG structure is equivalent to an adjacency $(0, 1)$-matrix where the acyclicity requirements necessarily dictates triangular shape and zero main diagonal. Hence, in order to produce arbitrary DAG structures, it is sufficient to generate random triangular adjacency matrices.)

In conclusion, after having defined all the necessary parameters, the sample construction can be carried out starting with the sampling of the set of root nodes, and proceeding sequentially forward following the rule that, given a particular ancestral sample $y^*$, the joint probability distribution for the downstream node combined with the ancestral sample is given by

$$P(x \cap y = y^*) = P(x|y = y^*)P(y = y^*) \tag{15}$$

i.e. for each sample $y = y^*$ the sample $x = x^*$ will be obtained with probability $P(x = x^* \cap y = y^*)$. This serialized construction of samples will produce a single row of data with values corresponding to nodes in the probabilistic DAG. This row will then constitute a single sample from the given DAG. Repeating this procedure as necessary will generate a dataset of prescribed size. We discuss sample size considerations below in Results: Section 5.

## 3 Methods, Proofs and Results: Random graph distribution

In order to make error estimates in structure recovery for the inverse solver (actual BN reconstruction algorithm), the forward solver (synthetic data generator) needs to not only synthesize data from a given model, but also be able to generate random model structures (joint probability factorizations) that the data is sampled from. This can be realized by producing random DAG adjacency matrices, or, more specifically, triangular extractions from symmetric adjacency matrices to account for the acyclicity of structures.

Thus, synthesized random structures should be distributed in a prescribed manner, a crucial characteristic without which error estimation will necessarily lead to unreliable results. Therefore, we need to establish some of the basic properties of DAG structure distributions.

The discrete nature of graph configurations implies nonuniform distribution across various graph densities for combinatorical reasons. For a graph of $n$ variables the maximum density (maximum number of edges) possible is given by

$$D = \frac{n(n-1)}{2} \tag{16}$$

,

Then the total count $C$ of possible graph configurations with a density $d < D$ is given by the binomial coefficient

$$C(d) = \frac{D!}{d!(D-d)!} \tag{17}$$

which implies that structures with density around $\frac{1}{2}D$ are more numerous than symmetrically higher or lower density structures. The total number of possible configurations of any density is

$$\sum_{k=0}^{D} C(k) = 2^D \tag{18}$$

Let $\delta$ be the density valued function over graph structures. Then the probability that a graph $\mathcal{G}$ has density $\delta(\mathcal{G}) = d$ is

$$P(\delta(\mathcal{G}) = d) = \frac{C(d)}{2^D} \tag{19}$$

The joint probability of obtaining a configuration $\mathcal{G}$ with density $d$ is therefore

$$P(\mathcal{G} \cap (\delta(G) = d)) = P(\mathcal{G}|\delta(G) = d)P(\delta(\mathcal{G}) = d) = \frac{1}{C(d)}\frac{C(d)}{2^D} = \frac{1}{2^D} \tag{20}$$

where the conditional probability of $\mathcal{G}$ given the prescribed density is

$$P(\mathcal{G}|\delta(\mathcal{G}) = d) = \frac{1}{C(d)} \tag{21}$$

Of particular interest is the expression for $P(\delta(\mathcal{G}) = d)$ since it implies that structures with medium density are much more likely than any others. This effect will not only bias the error estimation accuracy, but can also hinder the performance of a naive stochastic structure recovery method that searches the state space by evaluation only, not accounting for the fact that medium density states will occur more frequently.

Technically, this could be remedied by a forced uniform sampling across various density groups, increasing the sampling rate over lower and higher density groups relative to the medium density groups or, similarly, by decreasing the sampling rate over the medium density groups. However, this may be impractical, because even for a problem of moderate size, the number of possible configurations at the extremes of the distribution is sharply limited relative to medium density graphs, which would then make it virtually impossible to maintain constant sampling error. For a $16$-node structure the above translates to $120$ configurations with one edge, $840261910995$ configurations with 8 edges, and in the order of $10^{34}$ medium density 60-edge configurations. Therefore, if a $10\%$ sampling rate is to be maintained at any cost, then the number of samples across all density groups is in the order of $10^{35}$ (with most of the computational load in the medium density groups). Therefore, we conclude that in practice, instead of pursuing a uniform sampling error, one should aim to address this situation analytically.

## 4 Methods, Proofs and Results: Forward inference over Markov Blankets

It is important to assure the congruence of independence / probabilistic inference mechanisms in the simulation framework (as described above) and during the BN construction process. In this section, we will dissect the ”localized” probabilistic inference within a BN by utilizing a concept of Markov Blanket. By definition, Markov Blanket $M$ of a particular variable $x$ renders such "center" variable conditionally independent from the rest of the BN (i.e. BN sans the Markov Blanket), with the "periphery" variables of $M$ completely determining the state of the center variable assuming their states are known. This naturally leads to the problem of estimating conditional probability of $x$ as a function of states of periphery variables in $M - x$.

$$P(x|M-x) = \frac{P(M)}{P(M-x)} \tag{22}$$

The computational difficulties here are the same as with the estimation of any sufficiently complex joint probability - often we would run out of accuracy due to the sampling error before we can estimate most joint events unless all the variables are defined analytically, and the dimensionality of the computation for such an estimation grows superexponentially with the number of variables. However, these difficulties can be mitigated by relying on the structural information encoded in the BN. If correct, the simplifications introduced into factorization reduce the computational burden and sampling error to the point of making the desired estimation not only feasible, but also more reliable. And although some of the difficulties alleviated are traded for the multiplicative propagation of error, the end result is still preferable to simply running out of samples to estimate joint events.

Let $M$ be Markov Blanket of $x$ and let $a(z) \subset M$ denote the ancestor set of $z \in M$ in $M$ characterized by the following statement

$$\forall z \in M(a(z) \subset M) \quad \wedge \quad \forall s \subset a(z)\{P(z|a(z)) \neq P(z|a(z)-s)\} \tag{23}$$

where $s$ is any subset of the ancestor set. Further, let $d(z) \subset M$ be the descendant set of $z \in M$ characterized by the following

$$y \in d(z) \iff z \in a(y) \tag{24}$$

And, for notational convenience, let

$$ad(x) = a(d(x)) - x - a(x) - d(x) \tag{25}$$

so that $ad(x)$ represents the set of ancestors of members of $d(x)$ excluding $x$ itself that are neither direct ancestors or descendants of $x$ themselves, i.e.

$$x \notin ad(x), \quad ad(x) \cap a(x) = \varnothing, \quad ad(x) \cap d(x) = \varnothing \tag{26}$$

Then the joint probability of MB of $x$ can be factored as follows

$$P(M) = P(x, a(x), d(x), ad(x)) = P(a(x))P(x|a(x)) \prod_{y_i \in d(x)} P(y_i|a(y_i)) \prod_{z_j \in ad(x)} P(z_j|a(z_j)) \tag{27}$$

The above equation can be derived through the following scheme

$$\begin{aligned}
&P(x, a(x), d(x), ad(x)) = P(x, a(x), ad(x))P(d(x)|x, a(x), ad(x)) \\
&= P(a(x))P(x, ad(x)|a(x))P(d(x)|x, a(x), ad(x)) \\
&= P(a(x))P(x|a(x))P(ad(x)|x, a(x))P(d(x)|x, a(x), ad(x)) \\
&= P(a(x))P(x|a(x))P(d(x)|a(d(x)))P(ad(x)|a(x))
\end{aligned} \tag{28}$$

Methodical application of chain rule to the remaining complex conditional probability terms yields the result in (27). Also, note that

$$ad(x) \cap d(x) = \varnothing \implies P(ad(x)|x, a(x)) = P(ad(x)|a(x)) \tag{29}$$

and that

$$a(d(x)) \cap d(x) \neq \varnothing \implies \exists y \in d(x)(a(y) \cap d(x) \neq \varnothing) \tag{30}$$

Of course, one could try using any arbitrary factorization for the same purpose, but unless that factorization happens to be in the same class of equivalence with all the conditional independence simplifications in place, the arbitrary structure will necessarily be less optimal than the factorization found through the structural optimization process.

Equation (27) provides direct access to all the joint probabilities for any instantiation of variables in $M$, allowing to proceed with the estimation of the desired conditional probability

$$P(x = x_i|M - x) = \frac{P(x = x_i, M - x)}{\sum_j P(x = x_j, M - x)} \tag{31}$$

marginalizing over $x$ in the denominator for any given state $x_i$ of $x$ in the numerator. Undoubtedly, the division operation itself introduces further numerical error into the estimate, but the classical numerical techniques are applicable in this situation.

Different types of inference are possible under the same assumptions, utilizing the same set of equations. For example, one could assess the probability of a periphery variable $z \in M$ being in a state $z_i$ in order for the center variable $x \in M$ to be observed in a state $x_j$:

$$P(z = z_i|x = x_j, M - x - z) = \frac{P(x = x_j, z = z_j, M - x - z)}{\sum_k P(x = x_j, z = z_k, M - x - z)} \tag{32}$$

In essence, the above type of inquiry allows to make a judgment about the most likely scenario that caused the $x = x_j$ event, and could be extended to more than one periphery variable.

Furthermore, the idea that periphery variables can predict the state of the center variable with a prescribed accuracy can also be utilized in structural optimization during the BN structure search. Local prediction accuracy as an objective optimization criterion is inherently practical in that it attempts to maximize the utility of the result rather than a set of more abstract notions associated with information properties.

## 5 Results: Sample size considerations

In practice, it is important to evaluate just how many samples should be generated (as prescribed in Section 2 above) for the estimation of the joint probability to be sufficiently robust. As a representative example, let us considered a network of eight variables, with arity of three and average connection density of 0.8. This network can assume $3^8 = 6561$ possible configurations, and it is straightforward to analytically compute the probabilities of all configurations. Subsequently, we can evaluate the effect of sample size in estimating the "true" distribution. Figure 1 shows the analytical and estimated distribution for all possible configurations in our example, for four (increasing) sample size values. To quantify the differences between the distributions, we further used Jensen-Shannon divergence (JSD), a distributional divergence metric (Figure 2). These results confirm that our procedures converge to zero distributional divergence with the increasing sample size. This, of course, is not unexpected — the real, practical, question is how to estimate the necessary minimal sample size for each novel simulation experiment/dataset. An obvious prescription would be to carry out the analysis along the lines of Figures 1-2 de novo for each new simulation experiment/dataset encountered by the investigators, while keeping in mind the investigators' hardware and software resources and time constraints.

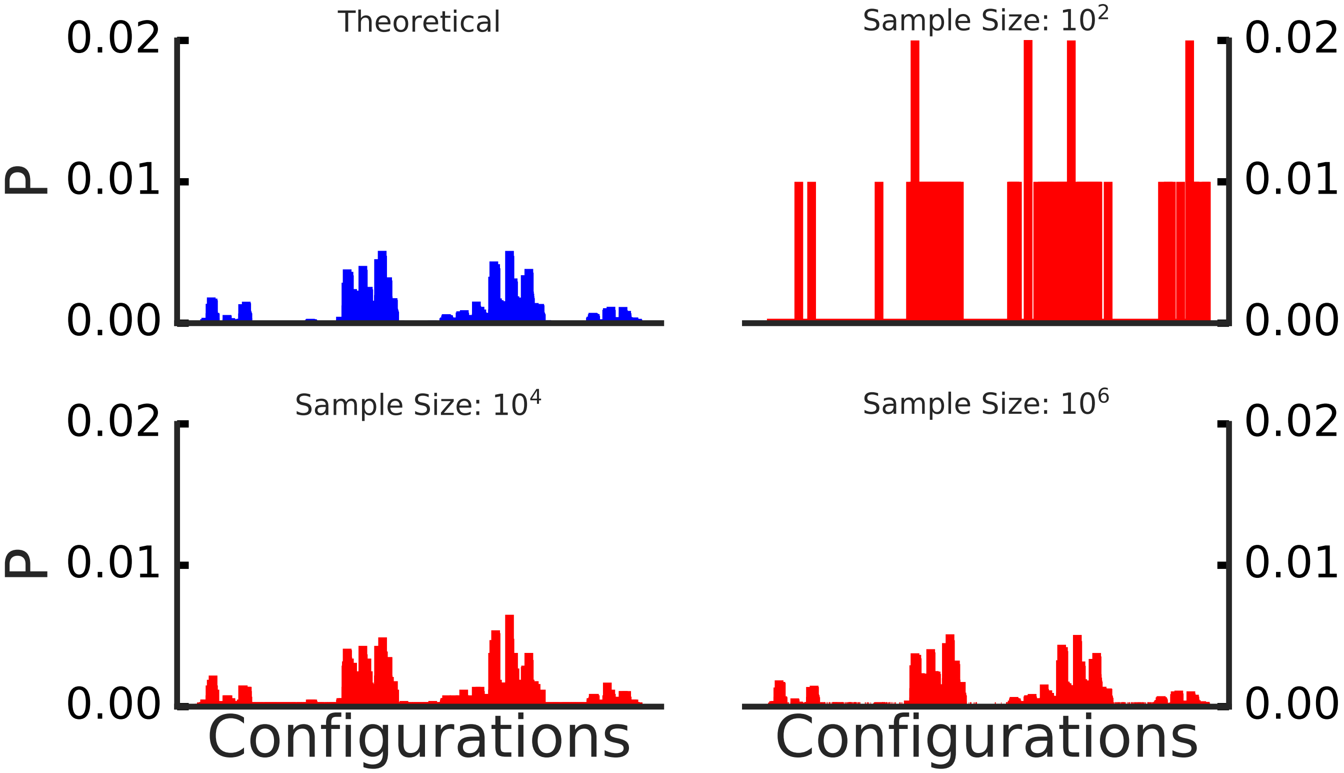

Figure 1: Comparison of the analytical and estimated sampled distributions of all possible configurations of a given 8-nodes random graph with average variable connection density and arity being 0.8 and 3, respectively. The analytical distribution is shown in top left panel (blue); distributions estimated using sample size ($10^2$, $10^4$, $10^6$) are shown in red.

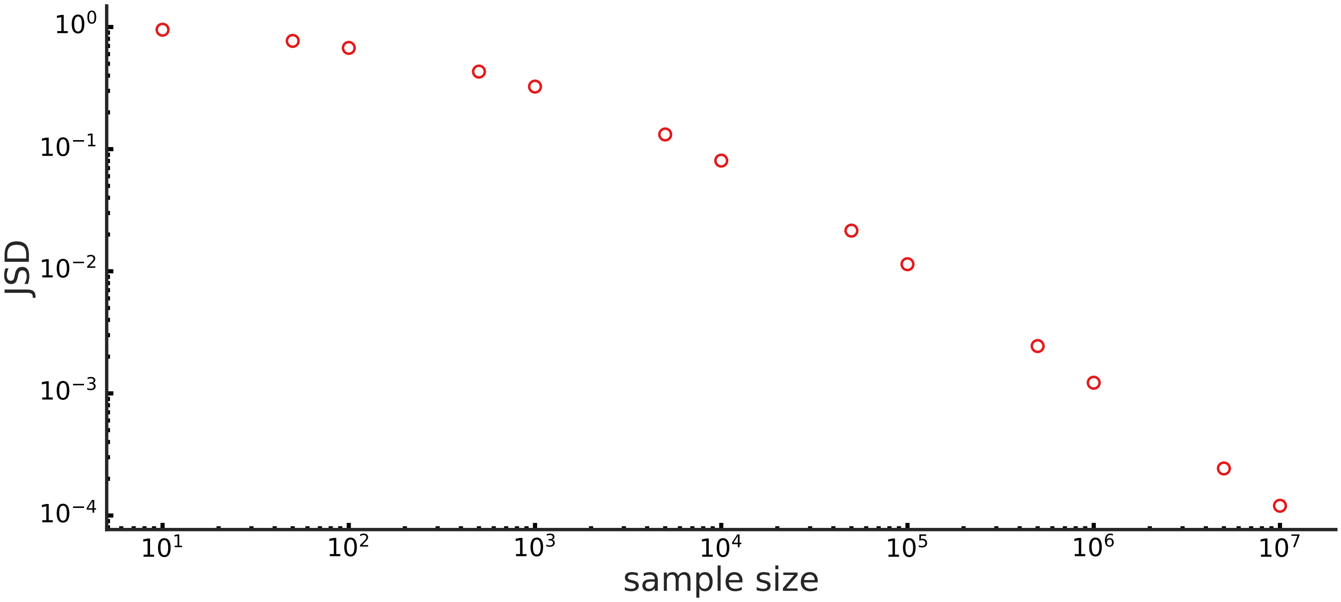

Figure 2: Jensen-Shannon divergence (JSD) between the analytical and estimated sampled distributions of all possible configurations of a given 8-nodes random graph with average variable connection density and arity being 0.8 and 3 respectively. JSD, shown as a function of sample size, is obtained by averaging 100 replicas for each sample size value.

## 6 Discussion and Conclusions

Simulation studies based on synthetic data generation are ubiquitous in machine learning / computer science domain. They are instrumental in objectively assessing the performance of descriptive/predictive modeling methods, such as BNs. However, comprehensive and realistic simulations frameworks are comparatively underdeveloped in the context of network-centered systems biology methodology. Most of the BN simulation studies there rely on SEM approximations — not necessarily a good fit with the biological reality. In this study, we have outlined theoretical and methodological considerations behind devising a more realistic, fully probabilistic, synthetic data generation framework for the biological BNs.

At this time, we have developed algorithms and software for generating synthetic data for randomly generated discrete-variables probabilistic DAGs, as part of our BNOmics BN modeling software package (Gogoshin, Boerwinkle, and Rodin 2017). The software is freely available from the authors, as well as from the bitbucket open source distributary (https://bitbucket.org/77D/bnomics). In future, we intend to expand it to the mixed data types (continuous/discrete variable mix, specifically).

Two outstanding issues, relevant to both the BN reconstruction, and synthetic data generation / BN performance evaluation, are (i) whether the Markov Blankets (within the BNs) can in principle be recovered with consistency (provided that the forward generation and backward reconstruction processes are sufficiently aligned, as in our proposed framework), and (ii) whether it is possible to incorporate a strict, rigorous definition of dependence / conditional independence in the BN reconstruction process, as opposed to concentrating on establishing causation from correlation. We are cautiously optimistic on both counts, but more investigation is needed, and will be forthcoming.

It remains to note that much of the work detailed in this communication has been spurred by the ongoing collaborative immuno-oncology study that involves comparative BN analyses of multi-dimensional FACS (Fluorescence-activated cell sorting) and other immuno-oncology datasets obtained from the gastrointestinal and breast cancer patients undergoing immunotherapy treatments. In it, we aim to construct and compare/contrast BNs representing immune signaling network states before and after therapy, in responders and non-responders. In future, we plan to generate synthetic datasets that hew closely to the real, observed FACS and other immuno-oncology datasets — this will allow us to rigorously validate the reconstructed immune signalling BNs and corresponding biological results. It is our strongly considered opinion that each and every BN analysis should, ideally, be accompanied by a corresponding simulation study built around the simulated data that is as close to the actual biological data under consideration as realistically possible.

## 7 Acknowledgements

The authors are grateful to Arthur D. Riggs, Russell C. Rockne, Peter P. Lee and Amanda J. Myers for many stimulating discussions and useful suggestions on the applicability of BN methodology to a wide variety of biomedical data.This work was supported by the NCI Cancer Biology System Consortium U01CA23221601 grant (to A.S.R.) and NIH NCI award P30CA033572. This work was also supported by the Susumu Ohno Chair in Theoretical and Computational Biology (held by A.S.R.), a Susumu Ohno Distinguished Investigator fellowship (to G.G.), and City of Hope funds (to G.G., S.B., and A.S.R.). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

## 8 Author Disclosure Statement

The authors declare that no competing financial interests exist.

## 9 Data Availability Statement

Relevant code and software are available, as part of the BNOmics package, at https://bitbucket.org/77D/bnomics, as well as directly from the authors.

## References

Andrews, B., J. Ramsey, and G. F. Cooper (2018). "Scoring Bayesian Networks of Mixed Variables". In: *Int. J. Data Sci. Anal.* 6.1, pp. 3–18. DOI: 10.1007/s41060-017-0085-7..

— (2019). "Learning High-dimensional Directed Acyclic Graphs with Mixed Data-types". In: *Proc. Mach. Learn. Res.* 104, pp. 4–21.

Branciamore, S. et al. (2018). "Intrinsic properties of tRNA molecules as deciphered via Bayesian Network and distribution divergence analysis". In: *Life* 8.1, E5. DOI: 10.3390/life8010005..

Chen, J. et al. (2019). "shinyBN: an online application for interactive Bayesian network inference and visualization". In: *BMC Bioinformatics.* 20.1, p. 711. DOI: 10.1186/s12859-019-3309-0..

Cooper, G. F. et al. (2015). "The center for causal discovery of biomedical knowledge from big data". In: *J. Am. Med. Inform. Assoc.* 22.6, pp. 1132–6. DOI: 10.1093/jamia/ocv059..

Daly, R., Q. Shen, and S. Aitken (2011). "Review: learning bayesian networks: approaches and issues". In: *The Knowl. Eng. Rev.* 26.2, 99157.

Eicher, T. et al. (2019). "Challenges in proteogenomics: a comparison of analysis methods with the case study of the DREAM proteogenomics sub-challenge". In: *BMC Bioinformatics.* 20.Suppl 24, p. 669. DOI: 10.1186/s12859-019-3253-z..

Ellis, B. and W. H. Wong (2008). "Learning causal Bayesian network structures from experimental data". In: *Journal of the American Statistical Association* 103.482, 778789.

Glymour, C., K. Zhang, and P. Spirtes (2019). "Review of Causal Discovery Methods Based on Graphical Models". In: *Front. Genet.* 10, p. 524. DOI: 10.3389/fgene.2019.00524..

Gogoshin, G., E. Boerwinkle, and A. S. Rodin (2017). "New algorithm and software (BNOmics) for inferring and visualizing Bayesian networks from heterogeneous big biological and genetic data". In: *J. Comp. Bio.* 24.4, pp. 340–356. DOI: 10.1089/cmb.2016.0100..

Han, S. et al. (2017). "An efficient Bayesian approach for Gaussian Bayesian network structure learning". In: *Commun. Stat. Simul. Comput.* 46.7, pp. 5070–5084. DOI: 10.1080/03610918.2016.1143103..

Heckerman, D. (1995). "Tutorial on Learning with Bayesian Networks". In: *Microsoft Research.*

Heckerman, D., D. Geiger, and D. Chickering (1995). "Learning Bayesian networks: The Combination of Knowledge and Statistical Data". In: *Machine Learning* 20, pp. 197–243.

Hong, Y. et al. (2018). "Causal Discovery Combining K2 with Brain Storm Optimization Algorithm". In: *Molecules* 16.7, E1729. DOI: 10.3390/molecules23071729..

Isci, S. et al. (2011). "Pathway analysis of high-throughput biological data within a Bayesian network framework". In: *Bioinformatics* 27.12, pp. 1667–74. DOI: 10.1093/bioinformatics/btr269..

Jabbari, F. et al. (2017). "Discovery of Causal Models that Contain Latent Variables through Bayesian Scoring of Independence Constraints". In: *Mach. Learn. Knowl. Discov. Databases* 2017, pp. 142–157. DOI: 10.1007/978-3-319-71246-8_9..

Jiang, X., M. M. Barmada, and S. Visweswaran. (2010). "Identifying genetic interactions in genome-wide data using Bayesian networks". In: *Genet. Epidemiol.*

Lan, Z. et al. (2016). "Bayesian network feature finder (BANFF): an R package for gene network feature selection". In: *Bioinformatics.* 32.23, pp. 3685–3687.

Neapolitan, R., D. Xue, and X. Jiang (2014). "Modeling the altered expression levels of genes on signaling pathways in tumors as causal bayesian networks". In: *Cancer Inform* 13, pp. 77–84.

Needham, C. J. et al. (2007). "A primer on learning in Bayesian networks for computational biology". In: *PLoS Comput. Biol.* 3.8.

Ogarrio, J. M., P. Spirtes, and Ramsey J (2016). "A Hybrid Causal Search Algorithm for Latent Variable Models". In: *JMLR Workshop Conf. Proc.* 52, pp. 368–379.

Pearl, J. (1988). "Probabilistic Reasoning in Intelligent Systems". In: *Morgan Kaufmann, San Mateo, CA.*

Pearl, J (2009). *Causality.*

Pe'er, D. (2005). "Bayesian network analysis of signaling networks: a primer". In: *Sci STKE (281):l4.*

Piatetsky-Shapiro, G. and P. Tamayo (2003). "Microarray Data Mining: Facing the Challenges". In: *ACM SIGKDD 5(2).*

Qi, Q., J. Li, and J. Cheng (2014). "Reconstruction of metabolic pathways by combining probabilistic graphical model-based and knowledge-based methods". In: *BMC Proc.* 8.Suppl 6 Proceedings of the Great Lakes Bioinformatics Confer, S5.

Raghu, V. K. et al. (2018). "Comparison of strategies for scalable causal discovery of latent variable models from mixed data". In: *Int. J. Data Sci. Anal.* 6.1, pp. 33–45. DOI: 10.1007/s41060-018-0104-3..

Ramsey, J. et al. (2017). "A million variables and more: the Fast Greedy Equivalence Search algorithm for learning high-dimensional graphical causal models, with an application to functional magnetic resonance images". In: *Int. J. Data Sci. Anal.* 3.2, pp. 121–129. DOI: 10.1007/s41060-016-0032-z..

Rodin, A. et al. (2005). "Mining genetic epidemiology data with Bayesian networks: Application to ApoE gene variants and plasma lipid levels". In: *Journal of Computational Biology* 12, pp. 1–11.

Rodin, A. S. et al. (2012). "Exploring Genetic Epidemiology Data with Bayesian Networks". In: *Handbook of Statistics, Elsevier B.V.;* 28, pp. 479–510.

Russell, S. and P. Norvig. (2009). "Artificial Intelligence: A Modern Approach (3rd Edition)". In: *Prentice Hall.*

Sedgewick, A. J. et al. (2016). "Learning mixed graphical models with separate sparsity parameters and stability-based model selection". In: *BMC Bioinformatics* 17.Suppl 5, p. 175. DOI: 10.1186/s12859-016-1039-0..

Sedgewick, A. J. et al. (2019). "Mixed Graphical Models for Integrative Causal Analysis with Application to Chronic Lung Disease Diagnosis and Prognosis". In: *Bioinformatics* 35.7, pp. 1204–1212. DOI: 10.1093/bioinformatics/bty769..

Sherif, F. F., N. Zayed, and M. Fakhr (2015). "Discovering Alzheimer Genetic Biomarkers Using Bayesian Networks". In: *Adv. Bioinformatics* 2015, p. 639367. DOI: 10.1155/2015/639367.

Spirtes, P., C. Glymour, and R. Scheines (2000). "Causation, Prediction, and Search". In: *The MIT Press, 2nd Edition.*

Spirtes, P. and K. Zhang (2016). "Causal discovery and inference: concepts and recent methodological advances". In: *Appl. Inform. (Berl).* 3, p. 3.

Tasaki, S. et al. (2015). "Bayesian network reconstruction using systems genetics data: comparison of MCMC methods". In: *Genetics* 199.4, pp. 973–89. DOI: 10.1534/genetics.114.172619..

Wang, L., P. Audenaert, and T. Michoel (2019). "High-Dimensional Bayesian Network Inference From Systems Genetics Data Using Genetic Node Ordering". In: *Front. Genet.* 10, p. 1196. DOI: 10.3389/fgene.2019.01196..

Xing, L. et al. (2017). "An improved Bayesian network method for reconstructing gene regulatory network based on candidate auto selection". In: *BMC Genomics.* 18.Suppl 9, p. 844. DOI: 10.1186/s12864-017-4228-y..

Xing, L. et al. (2018). "Gene Regulatory Networks Reconstruction Using the Flooding-Pruning Hill-Climbing Algorithm". In: *Genes (Basel).* 9.7, E342. DOI: 10.3390/genes9070342..

Yin, W. et al. (2015). "From genome-scale data to models of infectious disease: A Bayesian network-based strategy to drive model development". In: *Math. Biosci.* 270.Pt B, pp. 156–168.

Yu, K., L. Liu, and J. Li (2019). "Learning Markov Blankets from Multiple Interventional Data Sets". In: *IEEE Trans. Neural Netw. Learn. Syst.* DOI: 10.1109/TNNLS.2019.2927636..

Zeng, Z., X. Jiang, and R. Neapolitan (2016). "Discovering causal interactions using Bayesian network scoring and information gain". In: *BMC Bioinformatics* 17.1, p. 221. DOI: 10.1186/s12859-016-1084-8..

Zhang, K. et al. (2018). "Learning causality and causality-related learning: some recent progress". In: *Natl. Sci. Rev.* 5.1, pp. 26–29. DOI: 10.1093/nsr/nwx137..

Zhang, L. et al. (2019). "bAIcis: A Novel Bayesian Network Structural Learning Algorithm and Its Comprehensive Performance Evaluation Against Open-Source Software". In: *J. Comput. Biol.* DOI: 10.1089/cmb.2019.0210..

Zhang, Q. and X. Shi (2017). "A mixture copula Bayesian network model for multimodal genomic data". In: *Cancer Inform* 16. DOI: 10.1177/1176935117702389.

Zhang, X. et al. (2014). "Integrative Bayesian variable selection with gene-based informative priors for genome-wide association studies". In: *BMC Genet. 15:130. PMID: 25491445.*

Zhang, X. et al. (2017). "Analysis of high-resolution 3D intrachromosomal interactions aided by Bayesian network modeling". In: *Proc. Natl. Acad. Sci. USA* 114.48, E10359–E10368. DOI: 10.1073/pnas.1620425114,.

Ziebarth, J. D., A. Bhattacharya, and Y. Cui (2013). "Bayesian Network Webserver: a comprehensive tool for biological network modeling". In: *Bioinformatics,* 29.21, pp. 2801–3.