A Framework for Fault Diagnosis using Continuous Bayesian Network and Causal Inference

(Supplementary Material)

Asif Hanif [1]

S1. Logistic Mixture Loss Function

Assuming a random variable X has $logistic$ distribution;

$$\begin{aligned} X &\sim logistic(\mu, s) \\ F_X(x) &= \text{Pr}[X \leq x] \\ &= sigmoid(\frac{x-\mu}{s}) \\ &= \sigma(\frac{x-\mu}{s}) \end{aligned} \tag{S1.1}$$

where F_X is cumulative distribution function (CDF) of *logistic* PDF and Pr denotes probability. A bin around data sample x is defined as an interval $[x-\epsilon\ ,\ x+\epsilon]$ where $\epsilon > 0$ is half of the bin's total width. Probability of X taking values in the interval/bin $[x-\epsilon\ ,\ x+\epsilon]$ is computed as follows;

$$\begin{aligned} \text{Pr}_{\text{bin}} &= \text{Pr}[\ (x-\epsilon) \ \leq \ X \ \leq \ (x+\epsilon)\ ] \\ &= \text{Pr}[\ X \ \leq \ (x+\epsilon)\ ] \ - \ \text{Pr}[\ X \ \leq \ (x-\epsilon)\ ] \\ &= \sigma(\frac{x+\epsilon-\mu}{s}) - \sigma(\frac{x-\epsilon-\mu}{s}) \end{aligned} \tag{S1.2}$$

We are assuming M components in logistic mixture. If $\mu_{[m]}, s_{[m]}$ are parameters of m_{th} component, probability of X taking values in the interval $[x-\epsilon\ ,\ x+\epsilon]$ under m_{th} mixture component, i.e. $\text{Pr}_{x[m]}$, is given by;

$$\begin{aligned} \text{Pr}_{x[m]} &= \text{Pr}[\ (x-\epsilon) \ \leq \ X \ \leq \ (x+\epsilon)\ ]_{m_{\text{th}} \text{ mixture component}} \\ &= \sigma(\frac{x+\epsilon-\mu_{[m]}}{s_{[m]}}) \ - \ \sigma(\frac{x-\epsilon-\mu_{[m]}}{s_{[m]}}) \end{aligned} \tag{S1.3}$$

Note: In notation $\text{Pr}_{x[m]}$, m is being used as an index and should not be confused with probability *event*.

Collectively, M mixture components will assign following probability to interval/bin $[x-\epsilon, x+\epsilon]$;

$$\begin{aligned} \text{Pr}_x \ &= \ \text{Pr}[\ (x-\epsilon) \ \leq \ X \ \leq \ (x+\epsilon)\ ]_{\text{all mixture components}} \\ &= \ \sum_{m=1}^{M} \omega_{[m]} \ . \ \text{Pr}_{x[m]} \end{aligned} \tag{S1.4}$$

where M is the total number of mixture components and $\omega_{[m]}$ is the weight of m_{th} mixture component $(\omega_{[m]} \geq 0 \ \forall \ m, \ \sum_{m=1}^{M} \omega_{[m]} = 1)$. Pr_x is being obtained by taking convex combination of $\{\text{Pr}_{x[1]}, \text{Pr}_{x[2]}, \cdots, \text{Pr}_{x[M]}\}$.

[1]Department of Electrical Enginerring, Information Technology University, Lahore, Pakistan. Correspondence to: Asif Hanif <asif.hanif@itu.edu.pk, asif.hanif@outlook.com>.

For graphical illustration of Eq. S1.4, refer to Figure 1 in which probabilities have been highlighted as shaded regions.

Finally, loss is defined as sum of negative log-probabilities;

$$loss(\boldsymbol{\mu}, \boldsymbol{s}, \boldsymbol{\omega}; \mathcal{D}) \;=\; \sum_{k=1}^{K} -\log(\Pr_{x_k}) \tag{S1.5}$$

where $\mathcal{D}$ is the set of K data samples of random variable X, x_k is k_{th} sample of X and $\Pr_{x_k}$ is probability assigned to interval $[x_k - \epsilon, x_k + \epsilon]$ by mixture of logistic PDFs. log represents natural logarithm.

Our aim is to optimize following program using end-to-end training of neural network;

$$\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{s}}, \hat{\boldsymbol{\omega}} = \underset{\boldsymbol{\mu}, \boldsymbol{s}, \boldsymbol{\omega}}{\operatorname{argmin}} \;\; loss(\boldsymbol{\mu}, \boldsymbol{s}, \boldsymbol{\omega}; \mathcal{D}) \tag{S1.6}$$

where $\boldsymbol{\mu}, \boldsymbol{s}, \boldsymbol{\omega} = \{\, \mu[m], s[m], \omega[m] \,\}_{m=1}^{M}$ are parameterized by neural network's weights and biases. Program in Eq. S1.6 is equivalent to minimization of local negative log-likelihood function.

Minimization of sum of negative-log probabilities makes intuitive sense. When neural network returns probability density function that is away from ground truth[1], low probability will be assigned to data and loss will be high. Thus, it will force neural network to update weights (through backpropagation) such that density parameters converge to the values that assign high probability to data.

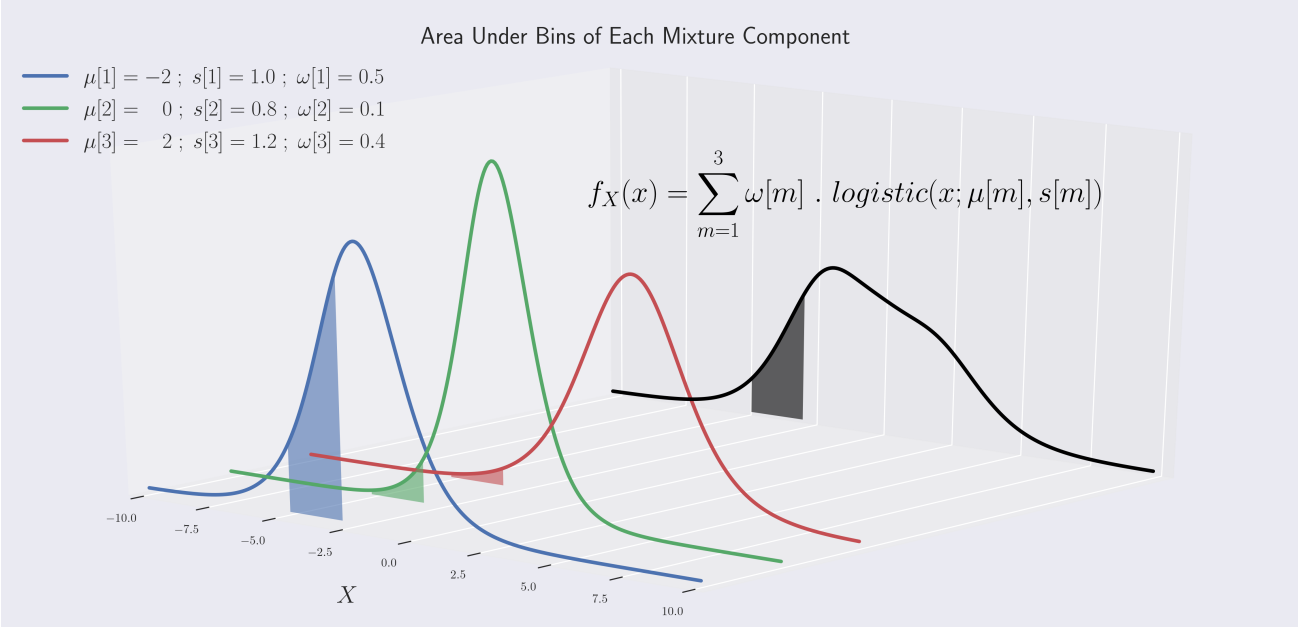


Figure 1. Logistic Mixture Loss: While calculating loss, area under the bins of each mixture component is computed. Each bin is centered around data sample x. If estimated density parameters $\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{s}}, \hat{\boldsymbol{\omega}}$ are close to ground truth, combined area of bins for all data samples will be maximum.

S2. Interpretation of $do()$ Operator

Notation $do(\ X\ )$ transforms a Bayesian network $\mathcal{G}$ into modified Bayesian network $\tilde{\mathcal{G}}$ by removing all the directed edges leading into node X in $\mathcal{G}$ (see Figure 2 for visualization). There are multiple ways to specify marginal PMF/PDF of X in $\tilde{\mathcal{G}}$.

$\mathcal{G}$: structure of Bayesian network **before** $do()$ operation

$\tilde{\mathcal{G}}$: structure of Bayesian network **after** $do()$ operation in $\mathcal{G}$

[1] parameters of actual probability density function are not given to neural network during training

Following notations precisely describe how marginal PMF/PDF of a node can be specified via $do()$ operation in $\mathcal{G}$.

[1] $\boldsymbol{do}\big(\ X = \hat{x}\ \big)$

Random variable X could be discrete or continuous. In both cases, $\boldsymbol{do}\big(\ X = \hat{x}\ \big)$ indicates that X has been made independent of its parent nodes (i.e. there is no arrow leading into X in modified Bayesian network $\tilde{\mathcal{G}}$).

If X is discrete node and $\hat{x}$ is one particular state of X , modified PMF (P_X) in $\tilde{\mathcal{G}}$ is defined as follows;

$$\boldsymbol{do}\big(\ X = \hat{x}\ \big)$$

$$P_X[x] = \begin{cases} 1 & \text{when } x = \hat{x} \\ 0 & \text{otherwise} \end{cases}$$

If X is continuous node and $\hat{x} \in$ support(X), $\boldsymbol{do}\big(\ X = \hat{x}\ \big)$ indicates that value of all of its samples drawn from $\tilde{\mathcal{G}}$ remains equal to $\hat{x}$.

[2] $\boldsymbol{do}\big(\ P_X = \vec{\theta}\ \big)$

This notation assumes X to be a **discrete** node. It removes all the edges leading towards X in $\mathcal{G}$. Removal of incoming edges makes X independent of its parents.

If $\vec{\theta}$ denotes user-specified or data-driven parameters of marginal distribution of X, modified PMF (P_X) in $\tilde{\mathcal{G}}$ is defined as follows;

$$\boldsymbol{do}\big(\ P_X = \vec{\theta}\ \big)$$

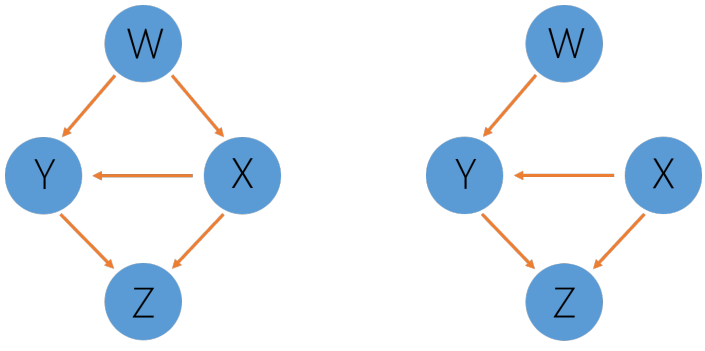
$$P_X[x] = \vec{\theta}[x] \quad \text{for each } x \in \{\text{discrete states of } X\}$$

Note: Here elements of vector $\vec{\theta}$ are being indexed by discrete states of X.

[3] $\boldsymbol{do}\big(\ f_X = f(\vec{\theta})\ \big)$

This notation assumes X to be a **continuous** node. It removes all the edges leading towards X in $\mathcal{G}$. Removal of incoming edges makes X independent of its parents and its marginal PDF in $\tilde{\mathcal{G}}$ is set to $f(\vec{\theta})$. Here $f(\vec{\theta})$ could be any arbitrary probability density function parameterized by values in vector $\vec{\theta}$.

For example, node X is a child of W in $\mathcal{G}$ (shown in Figure 2[a]). If conditional PDF of X in $\mathcal{G}$ is assumed to be $f_{X\ |\ (W=w)} = \mathcal{N}(\mu = 3 + w^2\ ,\ \sigma = 2)$, then $\boldsymbol{do}\big(\ f_X = \mathcal{N}(\mu_{new}, \sigma_{new})\ \big)$ indicates that there is no edge from W to X in modified Bayesian network $\tilde{\mathcal{G}}$ (see Figure 2[b]) and modified PDF of X becomes $f_X = \mathcal{N}(\mu = \mu_{new}\ ,\ \sigma = \sigma_{new})$



(a) $\mathcal{G}$: before $\boldsymbol{do}(X)$ (b) $\tilde{\mathcal{G}}$: after $\boldsymbol{do}(X)$

Figure 2. Bayesian network before and after $\boldsymbol{do}(X)$. **(a)** Original Bayesian network $\mathcal{G}$. **(b)** Modified Bayesian network $\tilde{\mathcal{G}}$ obtained after performing $\boldsymbol{do}(X)$ in $\mathcal{G}$. There is no arrow leading into node X in $\tilde{\mathcal{G}}$.

S3. Root Cause Analysis (RCA)

For RCA, we used thirteen-node continuous Bayesian network (shown in Figure 7). RCA results on 12 faulty data sets have been shown in Figure 3. Each faulty data set was obtained by intervening on a particular node's PDF in data generating process. For instance, data used for results shown in first subplot of Figure 3 was generated by modifying PDF of node X_1 in such a way that $Fault$ condition is triggered. We define $Fault$ condition on node X_{13} i.e. $Fault : X_{13} > 120$

When actual root-cause/*source* node X_s is directly connected to *sink* node X_t, RCA seems to be easy and provides confident results. For example, take the case when $X_s = X_5$. In this case, X_5 is the only node whose interventional probability is high. On the other hand, if X_s is not directly connected to X_t, one can face difficulties. Consider the case when $X_s = X_7$. Please note that interventional probabilities of X_7 and its descendants $\{X_{10}, X_{11}, X_{12}\}$ are significant as compared to that of other nodes. However, interventional probability of X_7 is not maximum. We suggest to use threshold to filter out the nodes with significant interventional probabilities and then explore their levels in graph. Among the filtered-out nodes, declare the node having highest level as root-cause/*source* node. Since in our case X_7 is at uppermost level when compared with levels of $\{X_{10}, X_{11}, X_{12}\}$, therefore,we can safely declare it as X_s.

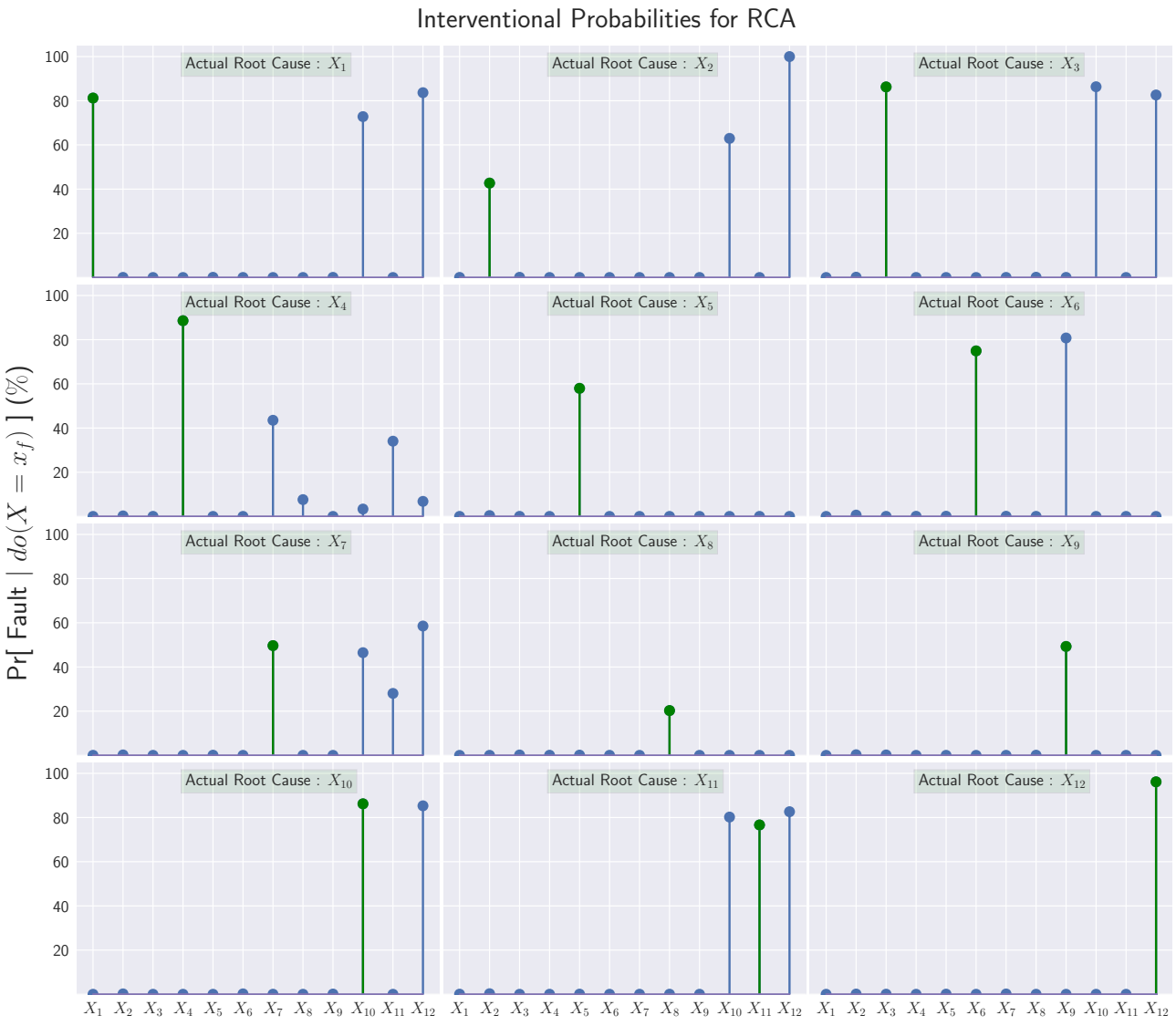


Figure 3. RCA results using 12 faulty data sets. X-axis of each subplot is labeled with the names of nodes and value on Y-axis (against the node X_i) can be interpreted as probability of $Fault$ being caused by node X_i. In each subplot, node corresponding to green stem-line represents **actual** root-cause/*source* node.

S4. Most Influential Path (MIP)

Identification of most influential[2] path between *source* node X_s and *sink* node X_t in a graph can be posed as the following program;

$$\boldsymbol{p}^* = \underset{\substack{\boldsymbol{p}_k \\ k\in\{1,2,\cdots,K\}}}{\operatorname{argmax}} \quad \Pr_{\mathcal{G}_k}[\, Fault \mid do(X_s = x_f)\,] \tag{S4.1}$$

where x_f represents value of X_s in faulty data. $\Pr_{\mathcal{G}_k}[.]$ notation indicates that probability is being found with respect to adjusted Bayesian network $\mathcal{G}_k$.

Following are the details to obtain an adjusted/path-specific Bayesian network $\mathcal{G}_k$ by modifying original Bayesian network $\mathcal{G}$ for an arbitrary path $\boldsymbol{p}_k$ between any two nodes. This method is slightly adapted version of (Pearl, 2013).

1. Assuming there are K different paths $\{\boldsymbol{p}_j\}_{j=1}^K$ from *source* node X_s to *sink* node X_t in $\mathcal{G}$. If one wants to find interventional effect of X_s on X_t only through k_{th} path $\boldsymbol{p}_k$, all other paths ($\{\boldsymbol{p}_j\}_{j=1}^K$ where $j \neq k$) need to be *de-activated* in graph.
2. To deactivate the non-relevant paths ($\{\boldsymbol{p}_j\}_{j=1}^K$ where $j \neq k$), a new adjusted Bayesian network is constructed– denoted as $\mathcal{G}_k$
3. Adjusted Bayesian network $\mathcal{G}_k$ is obtained by modifying the structure of $\mathcal{G}$. It involves **inclusion of new nodes** (that are being referred to as **StarNodes** here) and **removal of some edges**.
4. Steps to transform original Bayesian network $\mathcal{G}$ into path-specific adjusted Bayesian network $\mathcal{G}_k$ have been given in Table. 1.

Notation	Explanation
X_i	i_{th} node in $\mathcal{G}$
$pa(X_i)$	Parents of node X_i in $\mathcal{G}$
$pa(X_i)_{\boldsymbol{p}_k}$	Members of $pa(X_i)$ having an edge towards X_i in path $\boldsymbol{p}_k$
$pa(X_i)_{\overline{\boldsymbol{p}}_k}$	Members of $pa(X_i)$ having **no** edge towards X_i in path $\boldsymbol{p}_k$
$pa(X_i)^*_{\overline{\boldsymbol{p}}_k}$	**(StarNodes)** Same as the members of $pa(X_i)_{\overline{\boldsymbol{p}}_k}$ but with star (*) in their superscript
$pa(X_i)_{\mathcal{G}_k}$	$\{pa(X_i)_{\boldsymbol{p}_k}, pa(X_i)^*_{\overline{\boldsymbol{p}}_k}\}$ = New parents of node X_i in adjusted Bayesian network $\mathcal{G}_k$

Table 1. To construct an adjusted Bayesian network $\mathcal{G}_k$ for a path $\boldsymbol{p}_k$ between any two nodes, steps (row-wise) shown in this table are repeated for each node in $\mathcal{G}$.

After repeating steps (shown in Table 1) for each node in $\mathcal{G}$, one can form a path-specific adjusted Bayesian network $\mathcal{G}_k$. $\mathcal{G}_k$ contains an augmented set of nodes i.e. $\{X_i\ ,\ pa(X_i)^*_{\overline{\boldsymbol{p}}_k}\}_{i=1}^N$. In $\mathcal{G}_k$, updated parent set of node X_i is $\{pa(X_i)_{\boldsymbol{p}_k}, pa(X_i)^*_{\overline{\boldsymbol{p}}_k}\}$ and each **StarNode** i.e. $\{pa(X_i)^*_{\overline{\boldsymbol{p}}_k}\}_{i=1}^N$, has no parents and appear as *root* node in $\mathcal{G}_k$. Marginal PDF of each **StarNode** X_i^* in $\mathcal{G}_k$ is estimated using *healthy*[3] data of corresponding node X_i.

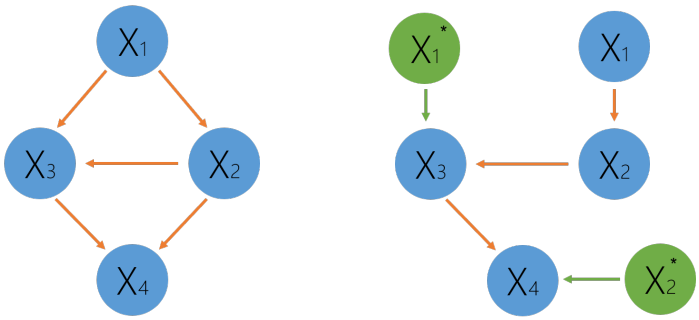
To illustrate the construction of an adjusted Bayesian network, an example has been given in Figure 4 and Table 2.

[2] influence of a path in triggering $Fault$ condition

[3] Here *healthy* data refers to the samples collected (from original Bayesian network $\mathcal{G}$) under normal or no-fault condition

i	X_i	$pa(X_i)$	$pa(X_i)_{\boldsymbol{p}_k}$	$pa(X_i)_{\overline{\boldsymbol{p}}_{\boldsymbol{k}}}$	$pa(X_i)^*_{\overline{\boldsymbol{p}}_{\boldsymbol{k}}}$	$pa(X_i)_{\mathcal{G}_k}$
1	X_1					
2	X_2	X_1	X_1			X_1
3	X_3	X_1, X_2	X_2	X_1	X_1^*	X_1^*, X_2
4	X_4	X_2, X_3	X_3	X_2	X_2^*	X_2^*, X_3

Table 2. This table shows construction of adjusted Bayesian network $\mathcal{G}_k$ for path $\boldsymbol{p}_k = \{X_1 \to X_2 \to X_3 \to X_4\}$ in graph $\mathcal{G}$ (shown in Figure. 4(a)). $\mathcal{G}_k$ is shown in Figure 4(b).



(a) $\mathcal{G}$ (original Bayesian network)

(b) $\mathcal{G}_k$ (path-specific adjusted Bayesian network)

Figure 4. Formation of adjusted Bayesian network $\mathcal{G}_k$ for path $\boldsymbol{p}_k = \{X_1 \to X_2 \to X_3 \to X_4\}$ **(a)** Original Bayesian network ($\mathcal{G}$). There are three directed paths from X_1 to X_4. We want to find interventional effect of X_1 on X_4 through path $\boldsymbol{p}_k = \{X_1 \to X_2 \to X_3 \to X_4\}$ **(b)** Adjusted Bayesian network $\mathcal{G}_k$ for path $\boldsymbol{p}_k = \{X_1 \to X_2 \to X_3 \to X_4\}$. Other two paths have been *de-activated* using star nodes $\{X_1^*, X_2^*\}$.

S5. Data Generating Process

We use forward/ancestral sampling to draw samples from a Bayesian network $\mathcal{G}$. In this process, values of *root* nodes are generated using their marginal PDFs. Afterwards, samples of descendants of *root* nodes are drawn using conditional PDFs successively.

$$x_i \sim f(X_i \mid pa(X_i))$$

Here X_i refers to random variable associated with node i and x_i is the realization of $X_i \mid pa(X_i)$ where $i \in \{1, 2, \cdots, N\}$ with N being the total number of nodes in the Bayesian network. Moreover, $f(X_i \mid pa(X_i))$ represents PDF of $X_i \mid pa(X_i)$.

While generating test data from the following processes, parameters of PDFs are also saved for qualitative and quantitative comparison with the predictions.

S5.1. Four Node Bayesian Network

Data generating process of a continuous Bayesian network (shown in Figure 5) over four nodes is given below. Sample of each random variable is being drawn from a distribution having tri-modal logistic PDF. Marginal histogram of data of each node is shown in Figure 6.

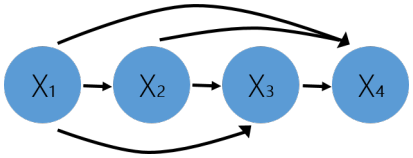


Figure 5. A fully-connected Bayesian network over four *continuous* nodes.

Draw Sample from $f(X_1)$

Mixture Component	Parameters
1^{st}	$\mu_{1[1]} = -10 \quad ; \quad s_{1[1]} = 0.5 \quad ; \quad \omega_{1[1]} = 0.5$
2^{nd}	$\mu_{1[2]} = 0 \quad ; \quad s_{1[2]} = 0.5 \quad ; \quad \omega_{1[2]} = 0.3$
3^{rd}	$\mu_{1[3]} = 10 \quad ; \quad s_{1[3]} = 0.5 \quad ; \quad \omega_{1[3]} = 0.2$

$$f(X_1) = \sum_{m=1}^{3} \omega_{1[m]} \,.\, logistic(\, \mu_{1[m]}, s_{1[m]} \,)$$

$$x_1 \sim f(X_1)$$

Draw Sample from $f(X_2 \mid X_1 = x_1)$

Mixture Component	Parameters
1^{st}	$\mu_{2[1]} = \sqrt{\lvert x_1 \rvert} \quad ; \quad s_{2[1]} = \lvert x_1 \rvert + 0.1 \quad ; \quad \omega_{2[1]} = 0.4$
2^{nd}	$\mu_{2[2]} = x_1 + 10 \quad ; \quad s_{2[2]} = \sqrt{\lvert x_1 \rvert + 0.1} \quad ; \quad \omega_{2[2]} = 0.2$
3^{rd}	$\mu_{2[3]} = 2x_1 - 10 \quad ; \quad s_{2[3]} = \sqrt{\lvert x_1 \rvert + 0.1} \quad ; \quad \omega_{2[3]} = 0.4$

$$f(X_2 \mid X1 = x_1) = \sum_{m=1}^{3} \omega_{2[m]} \,.\, logistic(\, \mu_{2[m]}, s_{2[m]} \,)$$

$$x_2 \sim f(X_2 \mid X1 = x_1)$$

Draw Sample from $f(X_3 \mid (X_1 = x_1,\ X_2 = x_2)\,)$

Mixture Component	Parameters
1^{st}	$\mu_{3[1]} = \sqrt{\lvert x_1 \rvert} \quad ; \quad s_{3[1]} = \sqrt{\lvert x_1 \rvert + 0.1} \quad ; \quad \omega_{3[1]} = 0.3$
2^{nd}	$\mu_{3[2]} = 2x_1 + 10 \quad ; \quad s_{3[2]} = \sqrt{\lvert x_1 \rvert + 0.1} \quad ; \quad \omega_{3[2]} = 0.4$
3^{rd}	$\mu_{3[3]} = x_1 + \sin(x_2) - 10 \quad ; \quad s_{3[3]} = \sqrt{\lvert x_2 \rvert + 0.1} \quad ; \quad \omega_{3[3]} = 0.3$

$$f(X_3 \mid (X_1 = x_1,\ X_2 = x_2)\,) = \sum_{m=1}^{3} \omega_{3[m]} \,.\, logistic(\, \mu_{3[m]}, s_{3[m]} \,)$$

$$x_3 \sim f(X_3 \mid (X_1 = x_1,\ X_2 = x_2)\,)$$

Draw Sample from $f(X_4 \mid (X_1 = x_1,\ X_2 = x_2,\ X_3 = x_3)\,)$

Mixture Component	Parameters
1^{st}	$\mu_{4[1]} = \sqrt{\lvert x_1 + \cos(x_3^2) \rvert} \quad ; \quad s_{4[1]} = \sqrt{\lvert x_1 + x_2 \rvert + 0.1} \quad ; \quad \omega_{4[1]} = 0.3$
2^{nd}	$\mu_{4[2]} = \sin(2x_2) - x_1 + 15 \quad ; \quad s_{4[2]} = \sqrt{\lvert x_1 - x_3 \rvert + 0.1} \quad ; \quad \omega_{4[2]} = 0.4$
3^{rd}	$\mu_{4[3]} = \cos(x_1) + \sin(x_3) - 15 \quad ; \quad s_{4[3]} = \sqrt{\lvert x_2 - x_3 \rvert + 0.1} \quad ; \quad \omega_{4[3]} = 0.3$

$$f(X_4 \mid (X_1 = x_1,\ X_2 = x_2,\ X_3 = x_3)\,) = \sum_{m=1}^{3} \omega_{4[m]} \,.\, logistic(\, \mu_{4[m]}, s_{4[m]} \,)$$

$$x_4 \sim f(X_4 \mid (X_1 = x_1,\ X_2 = x_2,\ X_3 = x_3)\,)$$

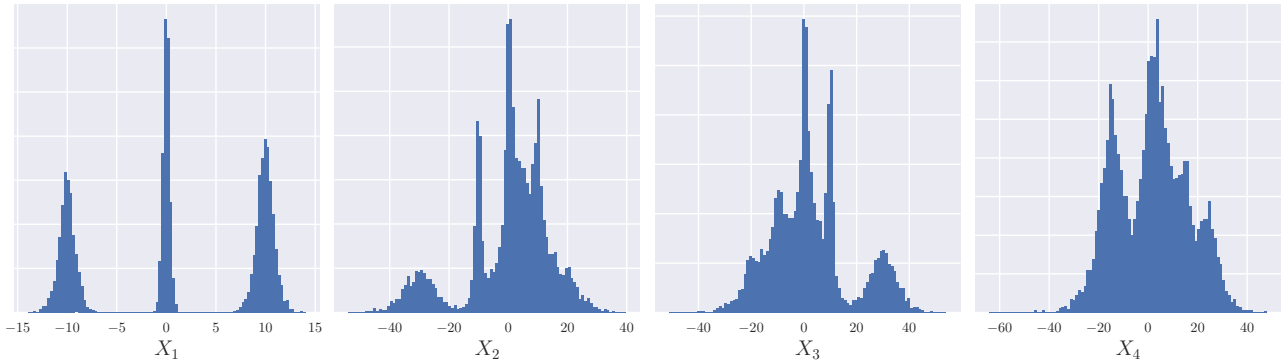


Figure 6. Marginal histogram of data of each node in four-node continuous Bayesian network (shown in Figure 5).

S5.2. Thirteen Node Bayesian Network

Data generating process of a continuous Bayesian network (shown in Figure 7) over thirteen nodes is given below. Marginal histogram of data of each node is shown in Figure 8.

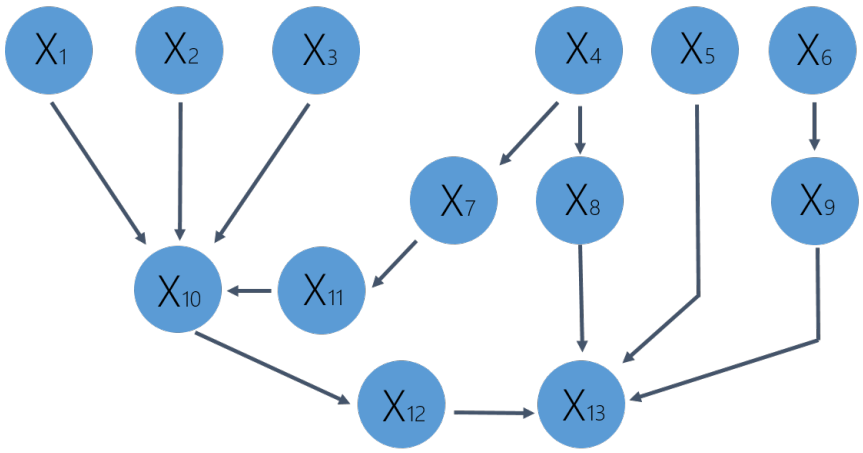


Figure 7. A Bayesian network over thirteen *continuous* nodes.

$$
\begin{aligned}
X_1 &\sim 0.5\ logistic(10, 1) + 0.5\ logistic(20, 1) \\
X_2 &\sim 0.5\ logistic(\ 15,\ 2) + 0.5\ logistic(\ 25,\ 1) \\
X_3 &\sim logistic(\ 10,\ 2) \\
X_4 &\sim logistic(\ 8,\ 2) \\
X_5 &\sim logistic(\ 15,\ 2) \\
X_6 &\sim logistic(\ 10,\ 2) \\
X_7 \mid (X_4 = x_4) &\sim logistic(\ 2\sqrt{|x_4|},\ 1) \\
X_8 \mid (X_4 = x_4) &\sim logistic(\ |x_4|^{0.7},\ 0.5) \\
X_9 \mid (X_6 = x_6) &\sim logistic(\max(x_6,\ 0),\ 0.5) \\
X_{10} \mid (X_1 = x_1,\ X_2 = x_2,\ X_3 = x_3,\ X_{11} = x_{11}) &\sim logistic(x_1 - x_2 + 1.1x_3 + 1.2x_{11},\ 1) \\
X_{11} \mid (X_7 = x_7) &\sim logistic(\ 3|x_7|^{0.8},\ 1) \\
X_{12} \mid (X_{10} = x_{10}) &\sim logistic(0.5x_{10},\ 0.5) \\
X_{13} \mid (X_5 = x_5,\ X_8 = x_8,\ X_9 = x_9,\ X_{12} = x_{12}) &\sim logistic(\ |-2.1x_5 + 2x_8 + 3x_9 + 5x_{12}\ |^{0.92},\ 0.5)
\end{aligned}
$$

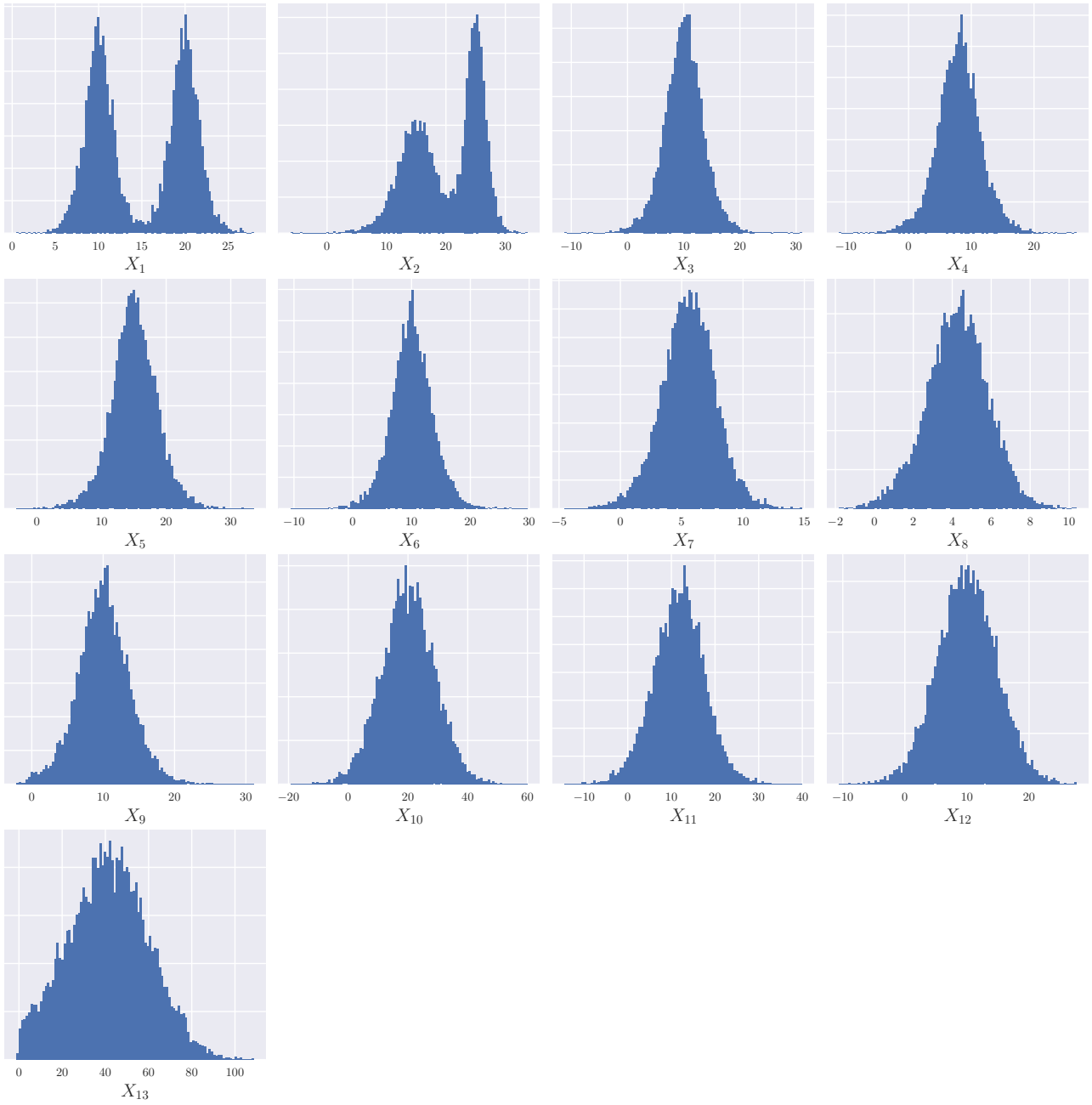


Figure 8. Marginal histogram of data of each node in thirteen-node continuous Bayesian network (shown in Figure 7).

S6. Derivation of Inference Query

Assume a Bayesian network $\mathcal{G}$ having N ***continuous*** nodes/random-variables[4] $\boldsymbol{X} = \{X_1, X_2, \cdots, X_N\}$ that are topologically-sorted and follow breadth-first-search(BFS) ordering. Joint PDF over N nodes is denoted as $f(X_1, X_2, \cdots, X_N)$. Let $K = \{1, 2, \cdots, N\}$ be an index set for the members of $\boldsymbol{X}$. One can compute joint probability of a set of random variables taking values in their corresponding intervals i.e. Pr$[\ \bigcap_{j \in J}(x_j^{'} \leq X_j \leq x_j^{''})\ ]$ by integrating joint PDF over the intervals. Here, $J \subseteq K$, $x_j^{'}$ and $x_j^{''}$ are the lower and upper limits of node X_j values.

[4]We will use term ***node*** and ***random-variable*** interchangeably.

$$\text{Query} = \Pr[\bigcap_{j\in J}(x_j^{'} \le X_j \le x_j^{''})\,] \tag{S6.1}$$
$$= \int\cdots\int_{\mathcal{T}} f(X_1, X_2, \cdots, X_N)\,\mathrm{d}x_{_N}\mathrm{d}x_{_{N-1}}\cdots\mathrm{d}x_{_3}\mathrm{d}x_{_2}\mathrm{d}x_{_1}$$

where $\mathcal{T}$ is a N-*dimensional* hyperrectangle (a region in N-*dimensional* real space), defined as a Cartesian-product of real-line intervals: $\mathcal{T} = (x_{_N}^{'}, x_{_N}^{''}) \times (x_{_{N-1}}^{'}, x_{_{N-1}}^{''}) \times \cdots \times (x_{_1}^{'}, x_{_1}^{''}) \subseteq \mathbb{R}^N$. For any $k \notin J$, the corresponding interval $(x_k^{'}\, x_k^{''})$ is assumed to be $(-\infty, +\infty)$.

Considering joint PDF $f(X_1, X_2, \cdots, X_n)$ is *Markov relative* (Pearl, 2009) to $\mathcal{G}$, it can be written as product of conditional PDFs.

$$= \int\cdots\int_{\mathcal{T}} \prod_{k=1}^{N} f(X_{_k} \mid \mathrm{pa}(X_{_k}))\,\mathrm{d}x_{_N}\mathrm{d}x_{_{N-1}}\cdots\mathrm{d}x_{_3}\mathrm{d}x_{_2}\mathrm{d}x_{_1}$$

If $q = \max(J)$, X_q is a node (among the nodes indexed by J) at deepest level in graph $\mathcal{G}$. All the nodes after X_q i.e. $\{X_{q+1}, X_{q+2}, \cdots, X_N\}$ will be marginalized-out in previous equation. In the following equations, $\mathcal{R}$ and $\mathcal{Q}$ are the sub-regions of $\mathcal{T}$ that are defined as Cartesian-product of real-line intervals:
$\mathcal{R} = (x_{_q}^{'}, x_{_q}^{''}) \times (x_{_{q-1}}^{'}, x_{_{q-1}}^{''}) \times \cdots \times (x_{_1}^{'}, x_{_1}^{''}) \subseteq \mathbb{R}^q \quad , \quad \mathcal{Q} = (x_{_{q-1}}^{'}, x_{_{q-1}}^{''}) \times (x_{_{q-2}}^{'}, x_{_{q-2}}^{''}) \times \cdots \times (x_{_1}^{'}, x_{_1}^{''}) \subseteq \mathbb{R}^{q-1}$

$$= \int\cdots\int_{\mathcal{R}} \prod_{k=1}^{q} f(X_{_k} \mid \mathrm{pa}(X_{_k}))\,\mathrm{d}x_{_q}\cdots\mathrm{d}x_{_3}\mathrm{d}x_{_2}\mathrm{d}x_{_1}$$
$$= \int\cdots\int_{\mathcal{Q}} \prod_{k=1}^{q-1} f(X_{_k} \mid \mathrm{pa}(X_{_k})) \cdot \left(\int_{x_q^{'}}^{x_q^{''}} f(X_q \mid \mathrm{pa}(X_q))\,\mathrm{d}x_{_q}\right)\mathrm{d}x_{_{q-1}}\cdots\mathrm{d}x_{_3}\mathrm{d}x_{_2}\mathrm{d}x_{_1}$$
$$= \int\cdots\int_{\mathcal{Q}} \prod_{k=1}^{q-1} f(X_{_k} \mid \mathrm{pa}(X_{_k})) \cdot \Big(\Pr[(x_q^{'} \le X_q \le x_q^{''}) \mid \mathrm{pa}(X_q)]\Big)\,\mathrm{d}x_{_{q-1}}\cdots\mathrm{d}x_{_3}\mathrm{d}x_{_2}\mathrm{d}x_{_1}$$
$$= \int\cdots\int_{\mathcal{Q}} f(X_1, X_2, \cdots, X_{q-1}) \cdot \Big(\Pr[(x_q^{'} \le X_q \le x_q^{''}) \mid \mathrm{pa}(X_q)]\Big)\,\mathrm{d}x_{_{q-1}}\cdots\mathrm{d}x_{_3}\mathrm{d}x_{_2}\mathrm{d}x_{_1}$$
$$= \int\cdots\int_{\mathcal{Q}} f(\boldsymbol{X}_{_{<q}}) \cdot \Big(\Pr[(x_q^{'} \le X_q \le x_q^{''}) \mid \mathrm{pa}(X_q)]\Big)\,\mathrm{d}x_{_{q-1}}\cdots\mathrm{d}x_{_3}\mathrm{d}x_{_2}\mathrm{d}x_{_1}$$
$$= \int_{-\infty}^{+\infty}\cdots\int f(\boldsymbol{X}_{_{<q}}) \cdot \Big(\mathbf{1}_{\mathcal{Q}}(\boldsymbol{q})\,.\,\Pr\big[(x_q^{'} \le X_q \le x_q^{''}) \mid \mathrm{pa}(X_q) = \pi_q(\boldsymbol{q})\big]\Big)\,\mathrm{d}x_{_{q-1}}\cdots\mathrm{d}x_{_3}\mathrm{d}x_{_2}\mathrm{d}x_{_1}$$
$$= \mathbf{E}_{f(\boldsymbol{X}_{_{<q}})}\Big[\mathbf{1}_{\mathcal{Q}}(\boldsymbol{q}) \;\cdot\; \Pr\big[(x_q^{'} \le X_q \le x_q^{''}) \mid \mathrm{pa}(X_q) = \pi_q(\boldsymbol{q})\big]\Big] \tag{S6.2}$$

where $f(\boldsymbol{X}_{_{<q}})$ is joint PDF over nodes $\{X_1, X_2, \cdots, X_{q-1}\}$. $\boldsymbol{q} \in \mathbb{R}^{q-1}$ is drawn from $f(\boldsymbol{X}_{_{<q}})$. $\pi_q()$ is a function that extracts values of node X_q's parents from input vector $\boldsymbol{q}$.

$$\boldsymbol{q} \;\sim\; f(\boldsymbol{X}_{_{<q}}) \quad ; \quad \mathbf{1}_{\mathcal{Q}}(\boldsymbol{q}) = 1 \text{ when } \boldsymbol{q} \in \mathcal{Q} \quad ; \quad \mathbf{1}_{\mathcal{Q}}(\boldsymbol{q}) = 0 \text{ when } \boldsymbol{q} \notin \mathcal{Q}$$

Using the same procedure described previously, one can derive expression for conditional probability query of the form $\Pr\Big[\bigcap_{i\in I}(x_i^{'} \le X_i \le x_i^{''}) \;\Big|\; \bigcap_{j\in J}(x_j^{'} \le X_j \le x_j^{''})\Big]$ as follows;

$$Conditional\ Query = \Pr\left[\bigcap_{i\in I}(x_i^{'} \le X_i \le x_i^{''}) \,\middle|\, \bigcap_{j\in J}(x_j^{'} \le X_j \le x_j^{''})\right] \tag{S6.3}$$

$$= \frac{\Pr\left[\bigcap_{i\in I}(x_i^{'} \le X_i \le x_i^{''}) \ \text{AND} \ \bigcap_{j\in J}(x_j^{'} \le X_j \le x_j^{''})\right]}{\Pr\left[\bigcap_{j\in J}(x_j^{'} \le X_j \le x_j^{''})\right]}$$

$$= \frac{\mathbf{E}_{f(\boldsymbol{X}_{<p})}\left[\mathbf{1}_{\mathcal{P}}(\boldsymbol{p}) \cdot \Pr\left[(x_p^{'} \le X_p \le x_p^{''}) \mid \text{pa}(X_p) = \pi_p(\boldsymbol{p})\right]\right]}{\mathbf{E}_{f(\boldsymbol{X}_{<q})}\left[\mathbf{1}_{\mathcal{Q}}(\boldsymbol{q}) \cdot \Pr\left[(x_q^{'} \le X_q \le x_q^{''}) \mid \text{pa}(X_q) = \pi_q(\boldsymbol{q})\right]\right]} \tag{S6.4}$$

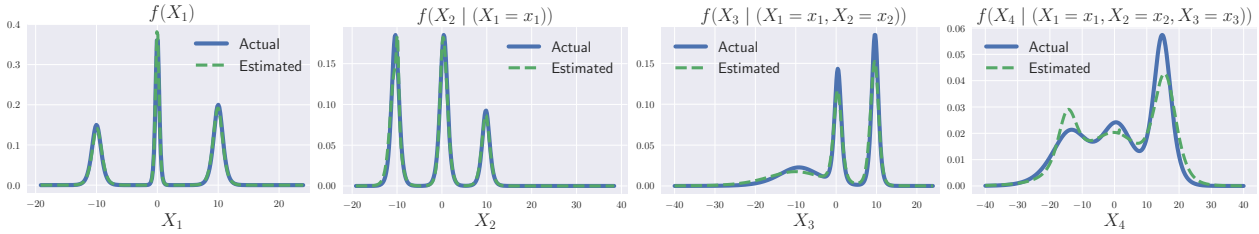
where $p = \max(I \cup J)$ and $q = \max(J)$ i.e. X_p and X_q are two nodes at the deepest level/s (among the nodes indexed by $(I \cup J)$ and J respectively) in $\mathcal{G}$.

$\mathcal{P}$ and $\mathcal{Q}$ are the regions in $\mathbb{R}^{p-1}$ and $\mathbb{R}^{q-1}$ defined by Cartesian-product of intervals $\{(x_k^{'}\ x_k^{''})\}_{k=1}^{p-1}$ and $\{(x_k^{'}\ x_k^{''})\}_{k=1}^{q-1}$ respectively. It must be noted that for any $k \notin (I \cup J)$, the corresponding interval $(x_k^{'}\ x_k^{''})$ is assumed to be $(-\infty, +\infty)$.

$\boldsymbol{p} \in \mathbb{R}^{p-1}$, $\boldsymbol{q} \in \mathbb{R}^{q-1}$ are drawn from joint PDFs $f(\boldsymbol{X}_{<p})$ and $f(\boldsymbol{X}_{<q})$ respectively. $\pi_p()$ is a function that extracts values of node X_p's parents from input vector $\boldsymbol{p}$. Expectations in Eq. S6.4 can be approximated using *Monte Carlo* method.

S7. Parameter Learning Results

The following results were obtained by training a ***single masked*** neural network with 6 layers, each layer comprising of 30 units.



(a) (b) (c) (d)

Figure 9. Actual and predicted PDFs associated with the nodes of Bayesian network shown in Figure 5. Visual inspection indicates that predictions are quite close to actual parameters. For quantitative comparison, we use TVD between two distributions.

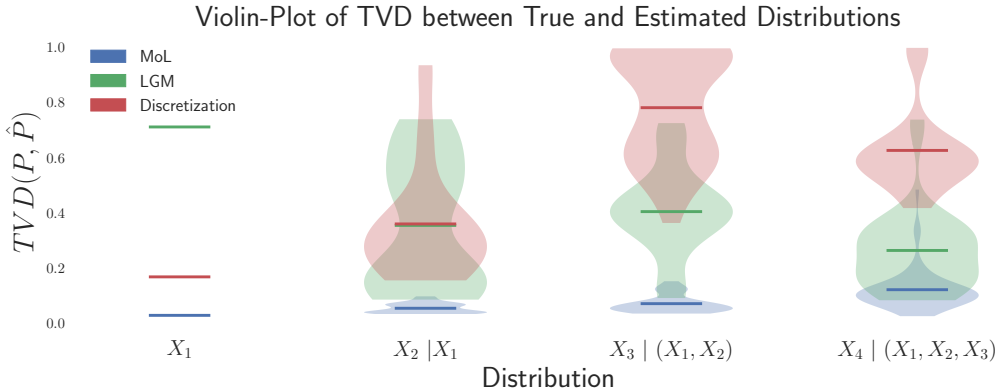


Figure 10. Violin-plot of TVD vs. each node in Bayesian network. Horizontal bar represents mean value of corresponding kernel density plot.

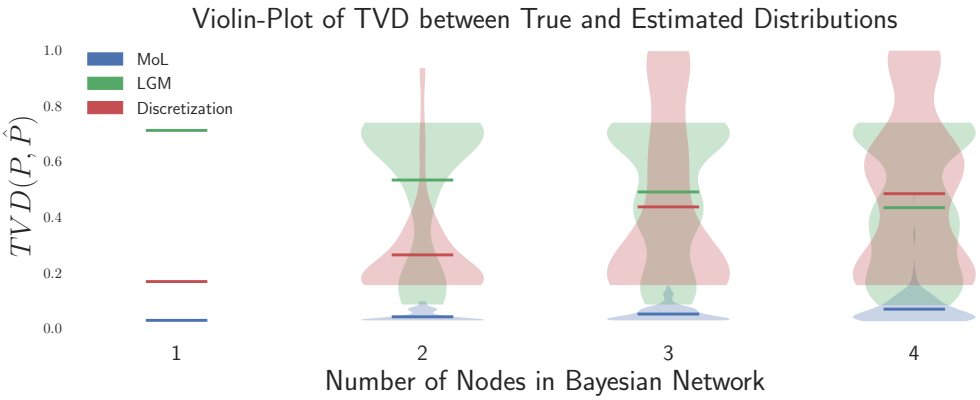


Figure 11. Violin-plot of TVD vs. number of nodes in Bayesian network. Horizontal bar represents mean value of corresponding kernel density plot.

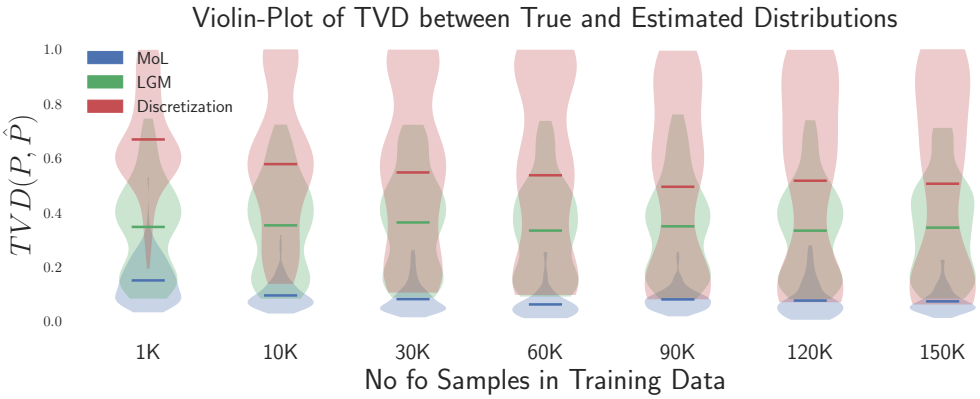


Figure 12. Violin-plot of TVD vs. number of training samples. Horizontal bar represents mean value of corresponding kernel density plot.

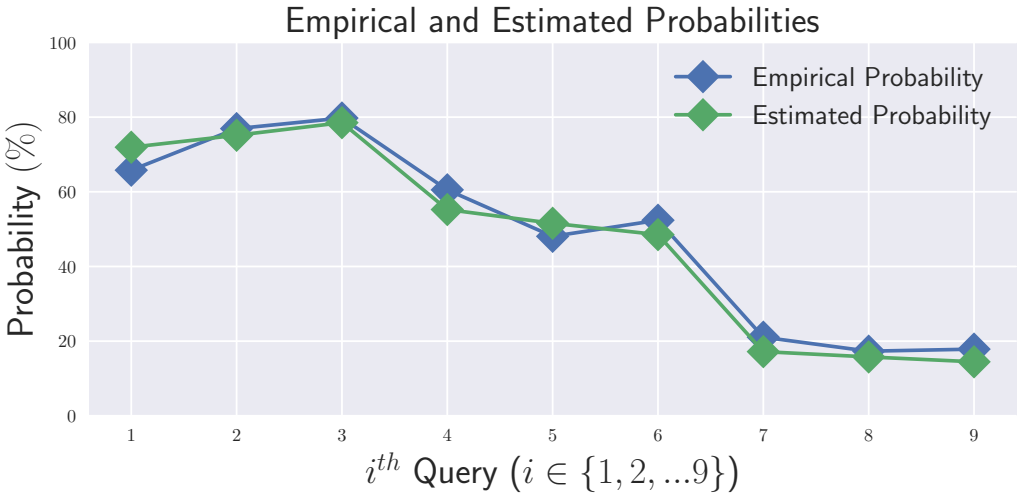


Figure 13. Empirical and estimated probabilities of nine inference queries e.g. 9_{th} query is $\Pr[(-30 < X_4 < 1)\&(-15 < X_3 < 0) \mid (-15 < X_1 < 15)\&(-15 < X_2 < 15)]$.

References

Pearl, J. *Causality*. Cambridge university press, 2009.

Pearl, J. Direct and indirect effects. *arXiv preprint arXiv:1301.2300*, 2013.