Bayesuvius,
a small visual dictionary of Bayesian Networks

Robert R. Tucci
www.ar-tiste.xyz

January 4, 2021

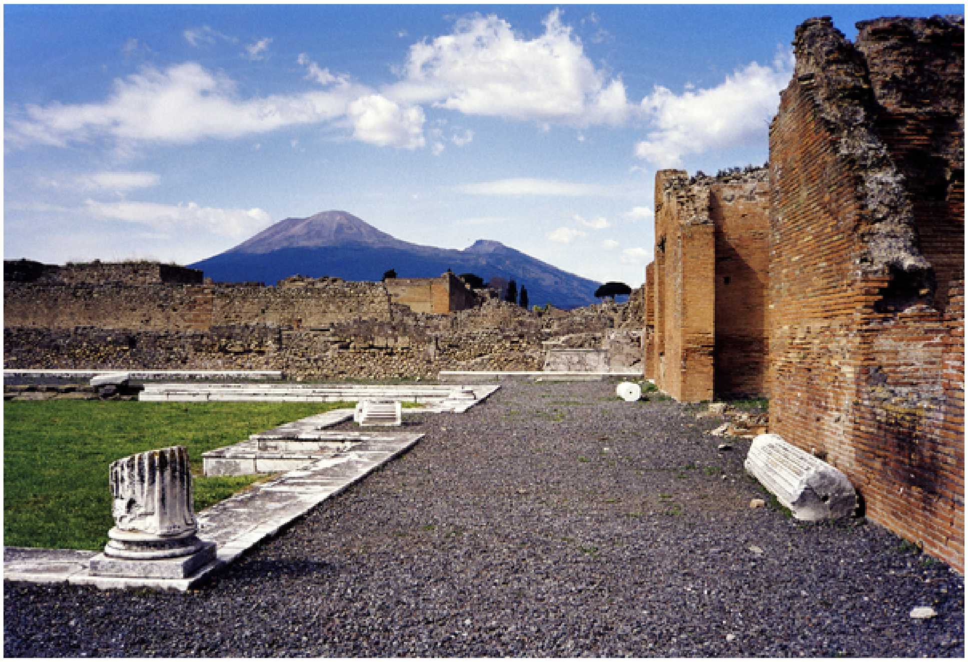


Figure 1: View of Mount Vesuvius from Pompeii

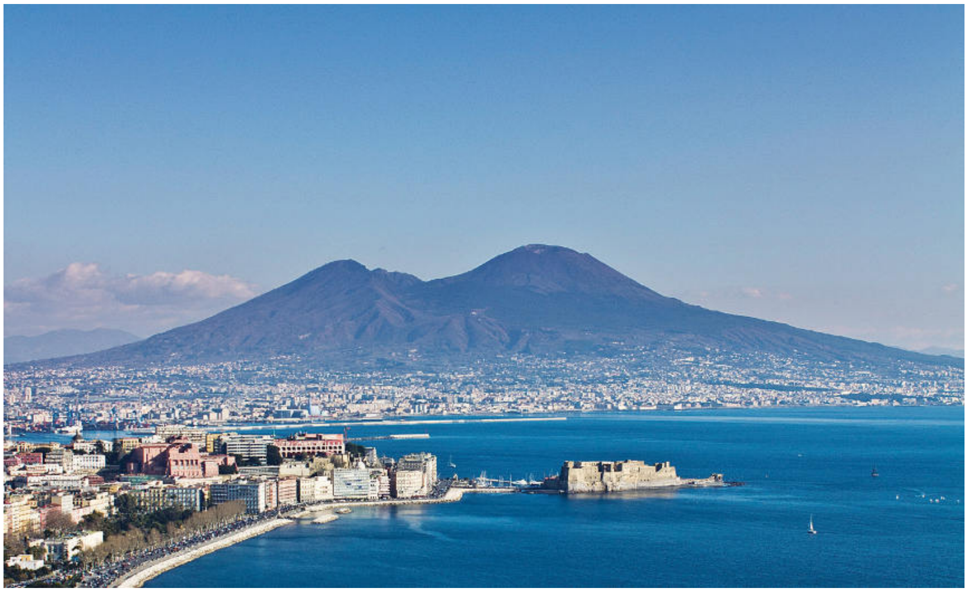


Figure 2: Mount Vesuvius and Bay of Naples

Contents

0.1 Foreword 5
0.2 Definition of a Bayesian Network 6
0.3 Notational Conventions and Preliminaries 8
0.4 Navigating the ocean of Judea Pearl's Books 16

1 ARACNE-structure learning 17

2 Backdoor Adjustment 19

3 Back Propagation (Automatic Differentiation) 23

4 Basic Curve Fitting Using Gradient Descent 30

5 Bell and Clauser-Horne Inequalities in Quantum Mechanics 32

6 Binary Decision Diagrams 33

7 Chow-Liu Trees and Tree Augmented Naive Bayes (TAN) 37

8 Counterfactual Reasoning 43

9 Decision Trees 50

10 Digital Circuits 53

11 Do-Calculus 55

12 D-Separation 65

13 Dynamic Bayesian Networks 68

14 Expectation Maximization 69

15 Front-door Adjustment 76

16 Generative Adversarial Networks (GANs) 78

17 Gaussian Nodes with Linear Dependence on Parents 83

18 Hidden Markov Model 86

19 Influence Diagrams & Utility Nodes 90

20 Junction Tree Algorithm 92

21 Kalman Filter 93

22 Linear and Logistic Regression 96

23 Linear Deterministic Bnets with External Noise 100

24 Markov Blankets 106

25 Markov Chain Monte Carlo (MCMC) 108

26 Markov Chains 117

27 Message Passing (Belief Propagation) 118

28 Missing Data, Imputation 148

29 Monty Hall Problem 154

30 Naive Bayes 156

31 Neural Networks 157

32 Noisy-OR gate 164

33 Non-negative Matrix Factorization 168

34 Observational Equivalence of DAGs 170

35 Potential Outcomes 173

36 Program evaluation and review technique (PERT) 174

37 Recurrent Neural Networks 179

38 Reinforcement Learning (RL) 188

39 Reliability Box Diagrams and Fault Tree Diagrams 197

40 Restricted Boltzmann Machines 205

41 Scoring the Nodes of a Learned Bnet 207

42 Simpson's Paradox 215

43 Structure and Parameter Learning for Bnets 219

44 Turbo Codes 224

45 Variational Bayesian Approximation 230

46 Zero Information Transmission (Graphoid Axioms) 235

Bibliography 238

0.2 Definition of a Bayesian Network

A **directed graph** $G = (V, E)$ consists of two sets, V and E. V contains the **vertices (nodes)** and E contains the **edges (arrows)**. An arrow $a \to b$ is an ordered pair (a, b) where $a, b \in V$.

The **parents** of a node x are those nodes a such that there are arrows $a \to x$. The **children** of a node x are those nodes b such that there are arrows $x \to b$. A **root node** is a node with no parents. A **leaf node** is a node with no children. The **neighbors** of a node x is the set of parents and children of x.

A **path** is a set of nodes that are connected by arrows, so that all nodes have 1 or 2 neighbors, but only two nodes (**open path**) or zero nodes (**closed path**) have only one neighbor. A **directed path** is a path in which all the arrows point in the same direction. A **loop** is a closed path;i.e., a path in which all nodes have exactly 2 neighbors. A **cycle** is a directed loop. A **Directed Acyclic Graph (DAG)** is a directed graph that has no cycles.

A **fully connected directed graph** is a directed graph in which every node has all other nodes as neighbors. Figs.3 and 4 show 2 different ways of drawing the same directed graph, a fully connected graph with 4 nodes. Note that a convenient way to label the nodes of a fully connected directed graph with N nodes is to point arrows from $\underline{x}_k$ to $\underline{x}_j$ where $j = 0, 1, 2, \dots, N-1$ and $k = j-1, j-2, \dots, 0$.

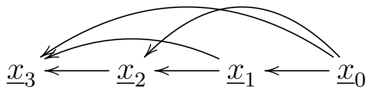


Figure 3: Fully connected directed graph with 4 nodes, drawn as a line.

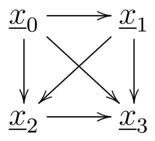


Figure 4: Fully connected directed graph with 4 nodes, drawn as a square.

A **connected graph** is a graph for which there is no way of separating the nodes into two sets so that there is no arrow from one set to the other. A **tree** is a directed graph in which all nodes have a single parent except for a single node called the "root" node which has no parents. A **polytree** is a DAG with no loops.

A **Bayesian network (bnet)** consists of a DAG and a **Transition Probability Matrix (TPM)** associated with each node of the graph. A TPM is often called a **Conditional Probability Table (CPT)**. The **structure** of a bnet is its DAG alone, sans the TPMs. The **skeleton** of a bnet is the undirected graph beneath the bnet's DAG.

In this book, random variables are indicated by underlined letters and their values by non-underlined letters. Each node of a bnet is labelled by a random variable. Thus, $\underline{x} = x$ means that node $\underline{x}$ is in state x.

Some sets of nodes associated with each node $\underline{a}$ of a bnet

- $ch(\underline{a})$ = children of $\underline{a}$.
- $pa(\underline{a})$ = parents of $\underline{a}$.
- $nb(\underline{a}) = pa(\underline{a}) \cup ch(\underline{a})$ = neighbors of $\underline{a}$.
- $de(\underline{a}) = \cup_{n=1}^{\infty} ch^n(\underline{a}) = ch(\underline{a}) \cup ch \circ ch(\underline{a}) \cup \ldots$, descendants of $\underline{a}$.
- $an(\underline{a}) = \cup_{n=1}^{\infty} pa^n(\underline{a}) = pa(\underline{a}) \cup pa \circ pa(\underline{a}) \cup \ldots$, ancestors of $\underline{a}$.

In this book, we will use $\underline{a}.$ to indicate a **multi-node (node set, node array)** $\underline{a}. = (\underline{a}_j)_{j=0,1,\ldots,na-1}$. We will often treat multinodes as if they were sets, and combine them with the usual set operators. For instance, for two multinodes $\underline{a}.$ and $\underline{b}.$, we define $\underline{a}. \cup \underline{b}.$, $\underline{a}. \cap \underline{b}.$, $\underline{a}. - \underline{b}.$ and $\underline{a}. \subset \underline{b}.$ in the obvious way. We will indicate a singleton set (single node multi-node) $\underline{a}. = \{\underline{a}\}$ simply by $\underline{a}. = \underline{a}$. For instance, $\underline{a}. - \underline{b} = \underline{a}. - \{\underline{b}\}$.

The TPM of a node $\underline{x}$ of a bnet is a matrix of probabilities $P(\underline{x} = x | pa(\underline{x}) = a.)$.

A bnet with nodes $\underline{x}.$ represents a probability distribution

$$P(x.) = \prod_j P(\underline{x}_j = x_j | (\underline{x}_k = x_k)_{k:\underline{x}_k \in pa(\underline{x}_j)}) \ . \tag{1}$$

Note that for a fully connected bnet with N nodes, Eq.(1) becomes

$$P(x.) = \prod_{j=0}^{N-1} P(x_j | (x_k)_{k=j-1,j-2,\ldots,0}) \ . \tag{2}$$

For example, if $N = 4$, Eq.(2) becomes

$$P(x_0, x_1, x_2, x_3) = P(x_3|x_2, x_1, x_0) P(x_2|x_1, x_0) P(x_1|x_0) P(x_0) \ . \tag{3}$$

We see that Eq.(2) is just the chain rule for conditional probabilities.

Given an arbitrarily large dataset of samples for the random variables $(\underline{x}_i)_{i=0,1,\ldots,N-1}$, there may be several bnets that fit the data well, but only one is used by Nature. That single one is called a **causal (or causally correct) Bayesian Network**. In this book, whenever we speak of causal issues, we will assume, often without mentioning it, that the correct bnet is being used.

0.3 Notational Conventions and Preliminaries

Some abbreviations frequently used throughout this book.

- bnet= B net= Bayesian Network
- CPT = Conditional Probabilities Table, same as TPM
- DAG = Directed Acyclic Graph
- i.i.d.= independent identically distributed.
- TPM= Transition Probability Matrix, same as CPT

Define $\mathbb{Z}, \mathbb{R}, \mathbb{C}$ to be the integers, real numbers and complex numbers, respectively.

For $a < b$, define $I_{\mathbb{Z}}$ to be the integers in the interval I, where $I = [a, b], [a, b), (a, b], (a, b)$ (i.e, I can be closed or open on either side).

$A_{>0} = \{k \in A : k > 0\}$ for $A = \mathbb{Z}, \mathbb{R}$.

Random variables will be indicated by underlined letters and their values by non-underlined letters. Each node of a bnet will be labelled by a random variable. Thus, $\underline{x} = x$ means that node $\underline{x}$ is in state x.

It is more conventional to use an upper case letter to indicate a random variable and a lower case letter for its state. Thus, $X = x$ means that random variable X is in state x. However, we have opted in this book to avoid that notation, because we often want to define certain lower case letters to be random variables or, conversely, define certain upper case letters to be non-random variables.

$P_{\underline{x}}(x) = P(\underline{x} = x) = P(x)$ is the probability that random variable $\underline{x}$ equals $x \in S_{\underline{x}}$. $S_{\underline{x}}$ is the set of states (i.e., values) that $\underline{x}$ can assume and $n_{\underline{x}} = |S_{\underline{x}}|$ is the size (aka cardinality) of that set. Hence,

$$\sum_{x \in S_{\underline{x}}} P_{\underline{x}}(x) = 1 \tag{4}$$

$$P_{\underline{x},\underline{y}}(x, y) = P(\underline{x} = x, \underline{y} = y) = P(x, y) \tag{5}$$

$$P_{\underline{x}|\underline{y}}(x|y) = P(\underline{x} = x|\underline{y} = y) = P(x|y) = \frac{P(x, y)}{P(y)} \tag{6}$$

Kronecker delta function: For x, y in discrete set S,

$$\delta(x, y) = \begin{cases} 1 \text{ if } x = y \\ 0 \text{ if } x \neq y \end{cases} \tag{7}$$

Dirac delta function: For $x, y \in \mathbb{R}$,

$$\int_{-\infty}^{+\infty} dx\ \delta(x - y) f(x) = f(y) \tag{8}$$

The TPM of a node of a bnet can be either a discrete or a continuous probability distribution. To go from continuous to discrete, one replaces integrals over states of a node by sums over new states, and Dirac delta functions by Kronecker delta functions. More precisely, consider a function $f : [a, b] \to \mathbb{R}$. Express $[a, b]$ as a union of small, disjoint (except for one point) closed sub-intervals (bins) of length Δx. Name one point in each bin to be the representative of that bin, and let $S_{\underline{x}}$ be the set of all the bin representatives. This is called discretization or binning. Then

$$\frac{1}{(b-a)}\int_{[a,b]} dx\; f(x) \to \frac{\Delta x}{(b-a)}\sum_{x\in S_{\underline{x}}} f(x) = \frac{1}{n_{\underline{x}}}\sum_{x\in S_{\underline{x}}} f(x)\ . \tag{9}$$

Both sides of last equation are 1 when $f(x) = 1$. Furthermore, if $y \in S_{\underline{x}}$, then

$$\int_{[a,b]} dx\; \delta(x-y) f(x) = f(y) \to \sum_{x\in S_{\underline{x}}} \delta(x,y) f(x) = f(y)\ . \tag{10}$$

Indicator function (aka Truth function):

$$\mathbb{1}(\mathcal{S}) = \left\{ \begin{array}{l} 1 \text{ if } \mathcal{S} \text{ is true} \\ 0 \text{ if } \mathcal{S} \text{ is false} \end{array} \right. \tag{11}$$

For example, $\delta(x, y) = \mathbb{1}(x = y)$.

$$\vec{x} = (x[0], x[1], x[2]\ldots, x[nsam(\vec{x}) - 1]) = x[:] \tag{12}$$

$nsam(\vec{x})$ is the number of samples of $\vec{x}$. $\underline{x}[\sigma] \in S_{\underline{x}}$ are i.i.d. (independent identically distributed) samples with

$$x[\sigma] \sim P_{\underline{x}} \ \text{ (i.e. } P_{\underline{x[\sigma]}} = P_{\underline{x}}\text{)} \tag{13}$$

$$P(\underline{x} = x) = \frac{1}{nsam(\vec{x})}\sum_{\sigma} \mathbb{1}(x[\sigma] = x) \tag{14}$$

Hence, for any $f : S_{\underline{x}} \to \mathbb{R}$,

$$\sum_{x} P(\underline{x} = x) f(x) = \frac{1}{nsam(\vec{x})}\sum_{\sigma} f(x[\sigma]) \tag{15}$$

If we use two sampled variables, say $\vec{x}$ and $\vec{y}$, in a given bnet, their number of samples $nsam(\vec{x})$ and $nsam(\vec{y})$ need not be equal.

$$P(\vec{x}) = \prod_{\sigma} P(x[\sigma]) \tag{16}$$

$$\sum_{\vec{x}} = \prod_{\sigma}\sum_{x[\sigma]} \tag{17}$$

$$\partial_{\vec{x}} = [\partial_{x[0]}, \partial_{x[1]}, \partial_{x[2]}, \ldots, \partial_{x[nsam(\vec{x})-1]}] \tag{18}$$

$$\begin{aligned} P(\vec{x}) &\approx [\prod_x P(x)^{P(x)}]^{nsam(\vec{x})} & (19) \\ &= e^{nsam(\vec{x})\sum_x P(x)\ln P(x)} & (20) \\ &= e^{-nsam(\vec{x})H(P_{\underline{x}})} & (21) \end{aligned}$$

$$f^{[1,\partial_x,\partial_y]}(x,y) = [f, \partial_x f, \partial_y f] \quad (22)$$

$$f^+ = f^{[1,\partial_x,\partial_y]} \quad (23)$$

For probabilty distributions $p(x), q(x)$ of $x \in S_{\underline{x}}$

- Entropy:
$$H(p) = -\sum_x p(x)\ln p(x) \geq 0 \quad (24)$$

- Kullback-Liebler divergence:
$$D_{KL}(p \parallel q) = \sum_x p(x)\ln\frac{p(x)}{q(x)} \geq 0 \quad (25)$$

- Cross entropy:
$$\begin{aligned} CE(p \to q) &= -\sum_x p(x)\ln q(x) & (26) \\ &= H(p) + D_{KL}(p \parallel q) & (27) \end{aligned}$$

Normal Distribution: $x, \mu, \sigma \in \mathbb{R}$, $\sigma > 0$

$$\mathcal{N}(x;\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad (28)$$

Uniform Distribution: $a < b$, $x \in [a, b]$

$$\mathcal{U}(x;a,b) = \frac{1}{b-a} \quad (29)$$

Expected Value and Variance

Given a random variable $\underline{x}$ with states $S_{\underline{x}}$ and a function $f : S_{\underline{x}} \to \mathbb{R}$, define

$$E_{\underline{x}}[f(\underline{x})] = E_{x\sim P(x)}[f(x)] = \sum_x P(x)f(x) \quad (30)$$

$$\begin{aligned} Var_{\underline{x}}[f(\underline{x})] &= E_{\underline{x}}\left[(f(\underline{x}) - E_{\underline{x}}[f(\underline{x})])^2\right] & (31) \\ &= E_{\underline{x}}[f(\underline{x})^2] - (E_{\underline{x}}[f(\underline{x})])^2 & (32) \end{aligned}$$

$$E[\underline{x}] = E_{\underline{x}}[\underline{x}] \tag{33}$$

$$Var[\underline{x}] = Var_{\underline{x}}[\underline{x}] \tag{34}$$

Conditional Expected Value

Given a random variable $\underline{x}$ with states $S_{\underline{x}}$, a random variable $\underline{y}$ with states $S_{\underline{y}}$, and a function $f : S_{\underline{x}} \times S_{\underline{y}} \to \mathbb{R}$, define

$$E_{\underline{x}|\underline{y}}[f(\underline{x},\underline{y})] = \sum_x P(x|\underline{y}) f(x,\underline{y}) \ , \tag{35}$$

$$E_{\underline{x}|\underline{y}=y}[f(\underline{x},y)] = E_{\underline{x}|y}[f(\underline{x},y)] = \sum_x P(x|y) f(x,y) \ . \tag{36}$$

Note that

$$\begin{aligned}
E_{\underline{y}}[E_{\underline{x}|\underline{y}}[f(\underline{x},\underline{y})]] &= \sum_{x,y} P(x|y)P(y)f(x,y) && (37)\\
&= \sum_{x,y} P(x,y)f(x,y) && (38)\\
&= E_{\underline{x},\underline{y}}[f(\underline{x},\underline{y})] \ . && (39)
\end{aligned}$$

Law of Total Variance

Claim 1 *Suppose $P : S_{\underline{x}} \times S_{\underline{y}} \to [0,1]$ is a probability distribution. Suppose $f : S_{\underline{x}} \times S_{\underline{y}} \to \mathbb{R}$ and $f = f(x,y)$. Then*

$$Var_{\underline{x},\underline{y}}(f) = E_{\underline{y}}[Var_{\underline{x}|\underline{y}}(f)] + Var_{\underline{y}}(E_{\underline{x}|\underline{y}}[f]) \ . \tag{40}$$

In particular,

$$Var_{\underline{x}}(x) = E_{\underline{y}}[Var_{\underline{x}|\underline{y}}(x)] + Var_{\underline{y}}(E_{\underline{x}|\underline{y}}[x]) \ . \tag{41}$$

proof:

Let

$$A = \sum_y P(y) \left(\sum_x P(x|y) f \right)^2 \ . \tag{42}$$

Then

$$\begin{aligned}
Var_{\underline{x},\underline{y}}(f) &= \sum_{x,y} P(x,y)f^2 - \left(\sum_{x,y} P(x,y) f \right)^2 && (43)\\
&= \left\{ \begin{array}{l} \sum_{x,y} P(x,y)f^2 - A \\ + \left(A - \left(\sum_{x,y} P(x,y) f \right)^2 \right) \end{array} \right. && (44)
\end{aligned}$$

$$
\begin{aligned}
E_{\underline{y}}[Var_{\underline{x}|\underline{y}}(f)] &= \sum_y P(y)\left(\sum_x P(x|y)f^2 - \left(\sum_x P(x|y)f\right)^2\right) && (45)\\
&= \sum_{x,y} P(x,y)f^2 - A && (46)
\end{aligned}
$$

$$
\begin{aligned}
Var_{\underline{y}}(E_{\underline{x}|\underline{y}}[f]) &= \sum_y P(y)\left(\sum_x P(x|y)f\right)^2 - \left(\sum_y P(y)\sum_x P(x|y)f\right)^2 && (47)\\
&= A - \left(\sum_{x,y} P(x,y)f\right)^2 && (48)
\end{aligned}
$$

QED

$\langle \underline{x}, \underline{y} \rangle$ notation, for covariances of any two random variables $\underline{x}, \underline{y}$.

Mean value of $\underline{x}$

$$\langle \underline{x} \rangle = E_{\underline{x}}[\underline{x}] \tag{49}$$

Signed distance of $\underline{x}$ to its mean value

$$\Delta \underline{x} = \underline{x} - \langle \underline{x} \rangle \tag{50}$$

Covariance of $(\underline{x}, \underline{y})$

$$\langle \underline{x}, \underline{y} \rangle = \langle \Delta \underline{x} \Delta \underline{y} \rangle = Cov(\underline{x}, \underline{y}) \tag{51}$$

Variance of $\underline{x}$

$$Var(\underline{x}) = \langle \underline{x}, \underline{x} \rangle \tag{52}$$

Standard deviation or $\underline{x}$

$$\sigma_{\underline{x}} = \sqrt{\langle \underline{x}, \underline{x} \rangle} \tag{53}$$

Correlation of $(\underline{x}, \underline{y})$

$$\rho_{\underline{x},\underline{y}} = \frac{\langle \underline{x}, \underline{y} \rangle}{\sqrt{\langle \underline{x}, \underline{x} \rangle \langle \underline{y}, \underline{y} \rangle}} \tag{54}$$

linear regression

$\underline{y}$ = true value
$\underline{\hat{y}}$ = estimator
$\underline{\epsilon}$ = residual

$$\underline{\hat{y}} = \beta_0 + \sum_{j=1}^{n} \beta_j \underline{x}_j \tag{55}$$

$$\underline{y} = \underline{\hat{y}} + \underline{\epsilon} \tag{56}$$

Assume

$$\langle \underline{x}_j, \underline{\epsilon} \rangle = 0 \tag{57}$$

for all j.

For $k = 1, \ldots, n$,

$$\langle \underline{x}_k, \underline{y} \rangle = \sum_{j=1}^{n} \beta_j \langle \underline{x}_k, \underline{x}_j \rangle \ . \tag{58}$$

Let $\underline{x}^n$ and β^n be column vectors. Then

$$\langle \underline{x}^n, \underline{y} \rangle = \langle \underline{x}^n, (\underline{x}^n)^T \rangle \beta^n \ , \tag{59}$$

$$\beta^n = \langle \underline{x}^n, (\underline{x}^n)^T \rangle^{-1} \langle \underline{x}^n, \underline{y} \rangle \ . \tag{60}$$

Sigmoid function

For $x \in \mathbb{R}$,

$$\text{sig}(x) = \frac{1}{1 + e^{-x}} \tag{61}$$

$\mathcal{N}(!a)$ will denote a normalization constant that does not depend on a. For example, $P(x) = \mathcal{N}(!x)e^{-x}$ where $\int_0^\infty dx\ P(x) = 1$.

A **one hot** vector of zeros and ones is a vector with all entries zero with the exception of a single entry which is one. A **one cold** vector has all entries equal to one with the exception of a single entry which is zero. For example, if $x^n = (x_0, x_1, \ldots, x_{n-1})$ and $x_i = \delta(i, 0)$ then x^n is one hot.

Short Summary of Boolean Algebra.

See Ref.[35] for more info about this topic.

Suppose $x, y, z \in \{0, 1\}$. Define

$$x \text{ or } y = x \vee y = x + y - xy \ , \tag{62}$$

$$x \text{ and } y = x \wedge y = xy \ , \tag{63}$$

and

$$\text{not } x = \overline{x} = 1 - x \ , \tag{64}$$

where we are using normal addition and multiplication on the right hand sides.[1]

Actually, since $x \wedge y = xy$, we can omit writing the symbol $\wedge$. The symbol $\wedge$ is useful to exhibit the symmetry of the identities, and to remark about the analogous identities for sets, where $\wedge$ becomes intersection $\cap$ and $\vee$ becomes union $\cup$. However, for practical calculations, $\wedge$ is an unnecessary nuisance.

Since $x \in \{0, 1\}$,

$$P(\overline{x}) = 1 - P(x) \ . \tag{65}$$

[1] Note the difference between $\vee$ and modulus 2 addition $\oplus$. For $\oplus$ (aka XOR): $x \oplus y = x + y - 2xy$.

Associativity	$x \vee (y \vee z) = (x \vee y) \vee z$ $x \wedge (y \wedge z) = (x \wedge y) \wedge z$
Commutativity	$x \vee y = y \vee x$ $x \wedge y = y \wedge x$
Distributivity	$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$ $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$
Identity	$x \vee 0 = x$ $x \wedge 1 = x$
Annihilator	$x \wedge 0 = 0$ $x \vee 1 = 1$
Idempotence	$x \vee x = x$ $x \wedge x = x$
Absorption	$x \wedge (x \vee y) = x$ $x \vee (x \wedge y) = x$
Complementation	$x \wedge \overline{x} = 0$ $x \vee \overline{x} = 1$
Double negation	$\overline{(\overline{x})} = x$
De Morgan Laws	$\overline{x} \wedge \overline{y} = \overline{(x \vee y)}$ $\overline{x} \vee \overline{y} = \overline{(x \wedge y)}$

Table 1: Boolean Algebra Identities

Clearly, from analyzing the simple event space $(x, y) \in \{0, 1\}^2$,

$$P(x \vee y) = P(x) + P(y) - P(x \wedge y) \ . \tag{66}$$

Definition of various entropies used in Shannon Information Theory

- **(plain) Entropy of $\underline{x}$**

$$H(\underline{x}) = -\sum_{x} P(x) \ln P(x) \tag{67}$$

This quantity measures the spread of $P_{\underline{x}}$.

- **Conditional Entropy of $\underline{y}$ given $\underline{x}$**

$$\begin{aligned} H(\underline{y}|\underline{x}) &= -\sum_{x,y} P(x, y) \ln P(y|x) & (68) \\ &= H(\underline{y}, \underline{x}) - H(\underline{x}) & (69) \end{aligned}$$

This quantity measures the conditional spread of $\underline{y}$ given $\underline{x}$.

- **Mutual Information (MI) of $\underline{x}$ and $\underline{y}$**

$$\begin{aligned} H(\underline{y} : \underline{x}) &= \sum_{x,y} P(x,y) \ln \frac{P(x,y)}{P(x)P(y)} && (70) \\ &= H(\underline{x}) + H(\underline{y}) - H(\underline{y}, \underline{x}) && (71) \end{aligned}$$

  This quantity measures the correlation between $\underline{x}$ and $\underline{y}$.

- **Conditional Mutual Information (CMI)[2] of $\underline{x}$ and $\underline{y}$ given $\underline{\lambda}$**

$$\begin{aligned} H(\underline{y} : \underline{x} | \underline{\lambda}) &= \sum_{x,y,\lambda} P(x,y,\lambda) \ln \frac{P(x,y|\lambda)}{P(x|\lambda)P(y|\lambda)} && (72) \\ &= H(\underline{x}|\underline{\lambda}) + H(\underline{y}|\underline{\lambda}) - H(\underline{y}, \underline{x}|\underline{\lambda}) && (73) \end{aligned}$$

  This quantity measures the conditional correlation of $\underline{x}$ and $\underline{y}$ given $\underline{\lambda}$.

- **Kullback-Liebler Divergence from $P_{\underline{x}}$ to $P_{\underline{y}}$.**

  Assume random variables $\underline{x}$ and $\underline{y}$ have the same set of states $S_{\underline{x}} = S_{\underline{y}}$. Then

$$D_{KL}(P_{\underline{x}} \parallel P_{\underline{y}}) = \sum_{x} P_{\underline{x}}(x) \ln \frac{P_{\underline{x}}(x)}{P_{\underline{y}}(x)} \tag{74}$$

  This measures a non-symmetric distance between the probability distributions $P_{\underline{x}}$ and $P_{\underline{y}}$. $D_{KL}(P_{\underline{x}} \parallel P_{\underline{y}})$ is non-negative and equals zero iff $P_{\underline{x}} = P_{\underline{y}}$.

[2] CMI can be read as "see me".

0.4 Navigating the ocean of Judea Pearl's Books

Many of the greatest ideas in the bnet field were invented by Judea Pearl and his collaborators. Thus, this book is heavily indebted to those great scientists. Those ideas have had no clearer and more generous expositor than Judea Pearl himself.

Pearl has written 4 books that I have used in writing Bayesuvius. His 1988 book Ref.[18] dates back to his pre-causal period. That book I used to learn about topics such as d-separation, belief propagation, Markov-blankets, and noisy-ORs. 3 other books that he wrote later, in his causal period, are:

1. In 2000 (1st ed.), and 2013 (2nd ed.), Pearl published what is so far his most technical and exhaustive book on the subject of causality, Ref.[19].

2. In 2016, he released together with Glymour and Jewell, a less advanced "primer" on causality, Ref.[21].

3. In 2018, he released together with Mackenzie his lovely "The Book of Why", Ref.[22].

Those 3 books I used to learn about causality topics such as do-calculus, backdoor and front-door adjustments, linear deterministic bnets with exogenous noise, and counterfactuals.

Chapter 2

Backdoor Adjustment

The backdoor (BD) adjustment theorem is proven in Chapter 11 from the rules of do-calculus. The goal of this chapter is to give examples of the use of that theorem. We will restate the theorem in this chapter, sans proof. There is no need to understand the theorem's proof in order to use it. However, you will need to skim Chapter 11 in order to familiarize yourself with the notation used to state the theorem. This chapter also assumes that you are comfortable with the rules for checking for d-separation. Those rules are covered in Chapter 12.

Suppose that we have access to data that allows us to estimate a probability distribution $P(x., y., z.)$. Hence, the variables $\underline{x}., \underline{y}., \underline{z}.$ are all observed (i.e, not hidden). Then we say that the backdoor $\underline{z}.$ satisfies the **backdoor adjustment criterion** relative to $(\underline{x}., \underline{y}.)$ if

1. All backdoor paths from $\underline{x}.$ to $\underline{y}.$ are blocked by $\underline{z}.$.
2. $\underline{z}. \cap de(\underline{x}.) = \emptyset$.

Claim 2 Backdoor Adjustment Theorem

If $\underline{z}.$ satisfies the backdoor criterion relative to $(\underline{x}., \underline{y}.)$, then

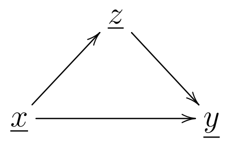
$$P(y.|\rho\underline{x}. = x.) = \sum_{z.} P(y.|x., z.)P(z.) \tag{2.1}$$

$$= \sum_{z.} \left\{ \begin{tikzcd} \underline{z}. = z. \arrow[dr] & \\ \underline{x}. = x. \arrow[r] & \underline{y}. \end{tikzcd} \right\} \tag{2.2}$$

proof: See Chapter 11
QED

Examples:

1.

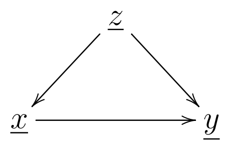


(2.3)

BD criterion satisfied if $\underline{x}. = \underline{x}, \underline{y}. = \underline{y}, \underline{z}. = \emptyset$. No adjustment necessary.

$$P(y|\rho\underline{x} = x) = P(y|x) \tag{2.4}$$

2.

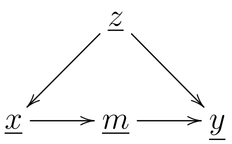


(2.5)

BD criterion satisfied if $\underline{x}. = \underline{x}, \underline{y}. = \underline{y}, \underline{z}. = \underline{z}$.

Note that here the backdoor formula adjusts the parents of $\underline{x}$..

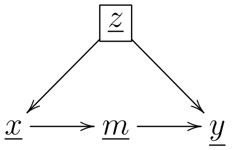
3.



(2.6)

BD criterion satisfied if $\underline{x}. = \underline{x}, \underline{y}. = \underline{y}, \underline{z}. = \underline{z}$.

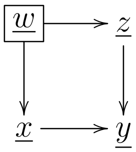
4.



(2.7)

BD criterion is impossible to satisfy if $\underline{x}. = \underline{x}, \underline{y}. = \underline{y}$. However, the front-door criterion can be satisfied. See Chapter 15.

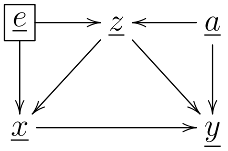
5.



(2.8)

BD criterion satisfied if $\underline{x}. = \underline{x}, \underline{y}. = \underline{y}, \underline{z}. = \underline{z}$. Note that here the backdoor formula cannot adjust the single parent $\underline{w}$ of $\underline{x}$ because it is hidden, but we are able to block the backdoor path by conditioning on $\underline{z}$ instead.

6.

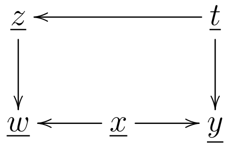


(2.9)

Conditioning on $\underline{z}$ blocks backdoor path $\underline{x} - \underline{z} - \underline{y}$, but opens path $\underline{x} - \underline{e} - \underline{z} - \underline{a} - \underline{y}$ because $\underline{z}$ is a collider for that path. That path is blocked if we also condition on $\underline{a}$, which is possible because $\underline{a}$ is observed. In conclusion, the BD criterion is satisfied if $\underline{x}. = \underline{x}$, $\underline{y}. = \underline{y}$ and $\underline{z}. = (\underline{z}, \underline{a})$.

Conditioning on the parents of $\underline{x}.$ is often enough to block all backdoor paths. However, sometimes some of the parents are unobserved and one must condition on other nodes that are not parents of $\underline{x}.$ in order to satisfy the BD criterion.

7.



(2.10)

No need to control anything because only possible backdoor path is blocked by collider $\underline{w}$. Hence,

$$P(y|\rho\underline{x} = x) = P(y|x) \ . \tag{2.11}$$

However, if for some reason we want to control $\underline{t}$, we can do so. We can't control $\underline{w}$ though, because $\underline{w} \in de(\underline{x})$. Thus, the BD criterion is satisfied if $\underline{x}. = \underline{x}$, $\underline{y}. = \underline{y}$ and $\underline{z}. = \underline{t}$. Therefore,

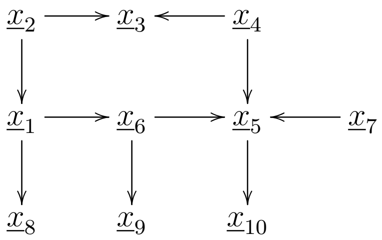
$$P(y|\rho\underline{x} = x) = \sum_t P(y|x,t)P(t) \ . \tag{2.12}$$

8. Discuss reasons why multiple possible sets $\underline{z}.$ that satisfy the BD criterion can be advantageous.

   - Can evaluate $P(y.|\rho\underline{x}. = x.)$ multiple ways and compare the results. This is a test that the causal bnet is correct.
   - It might be easier or less expensive to get data for some $\underline{z}.$ more than for others.

9. (Taken from online course notes Ref.[6])

Consider the bnet



(2.13)

If $\underline{x}. = \underline{x}_1$ and $\underline{y}. = \underline{x}_5$, find all possible adjustment multinodes $\underline{z}.$ that satisfy the BD criterion. Ans:

- $\emptyset$
- $\underline{x}_2$
- $\underline{x}_4$
- $\underline{x}_2, \underline{x}_4$
- $\underline{x}_2, \underline{x}_3$
- $\underline{x}_3, \underline{x}_4$
- $\underline{x}_2, \underline{x}_3, \underline{x}_4$

Add $\underline{x}_7$ to each of the previous 7 possible $\underline{z}.$. This gives a total of 14 possible adjustment multinodes $\underline{z}.$.

Chapter 7

Chow-Liu Trees and Tree Augmented Naive Bayes (TAN)

This chapter is mostly based on chapter 8 of Pearl's 1988 book Ref.[18]. See also Ref.[37] and references therein.

This chapter uses various Shannon Information Theory entropies. Our notation for these entropies is described in Chapter 0.3 on Notational Conventions.

Chow-Liu trees refers to an algorithm for finding a bnet tree that fits an a priori given probability distribution as closely as possible.

Consider a bnet with n nodes $\underline{x}^n = (\underline{x}_0, \underline{x}_1, \ldots, \underline{x}_{n-1})$ such that $\underline{x}_i \in S_{\underline{x}_i}$ for all i. Let its total probability distribution be $P_{\underline{x}^n}$. For simplicity, we will abbreviate $P_{\underline{x}^n}$ by P. Hence

$$P(x^n) = P_{\underline{x}^n}(x^n) \ . \tag{7.1}$$

Suppose we want to fit $P_{\underline{x}^n}$ by a tree bnet with nodes $\underline{t}^n = (\underline{t}_0, \underline{t}_1, \ldots, \underline{t}_{n-1})$ such that $\underline{t}_i \in S_{\underline{t}_i} = S_{\underline{x}_i}$ for all i. For simplicity, we will abbreviate $P_{\underline{t}^n}$ by P_T. Hence

$$P_T(x^n) = P_{\underline{t}^n}(x^n) \ . \tag{7.2}$$

Throughout this chapter, let $V = \{0, 1, \ldots, n-1\}$, the set of vertices. Suppose μ is a function $\mu : V \to V$ such that $\mu(i) < i$. Let $T_\mu = \{\underline{t}_{\mu(i)} \to \underline{t}_i : i \in V - \{0\}\}$. Then T_μ is a tree that spans (i.e., it includes all nodes) $\underline{t}^n$. Its root node is $\underline{t}_0$, because $\underline{t}_0$ has no parents. All other nodes $\underline{t}_i$ have exactly one parent, namely $\underline{t}_{\mu(i)}$. Let P_T, the total probability distribution for the tree, be parameterized by the function μ as follows:

$$P_T(x^n) = \prod_{i=0}^{n-1} P_T(x_i|x_{\mu(i)}) \ , \tag{7.3}$$

where, for the root node 0, $P_T(x_0|x_{\mu(0)}) = P_T(x_0)$.

Claim 3 *$D_{KL}(P \parallel P_T)$ is minimized over all probability distributions P_T that are expressible as Eq.(7.3) iff*

$$P_T(x_i|x_{\mu(i)}) = P(x_i|x_{\mu(i)}) \tag{7.4}$$

for all i, and

$$\sum_i H(\underline{x}_i : \underline{x}_{\mu(i)}) \tag{7.5}$$

is maximized over all μ.

proof:

$$\begin{aligned}
D_{KL}(P \parallel P_T) &= \sum_{x^n} P(x^n) \ln \frac{P(x^n)}{P_T(x^n)} &(7.6)\\
&= -\sum_{x^n}\sum_i P(x^n) \ln P_T(x_i|x_{\mu(i)}) + \sum_i P(x^n) \ln P(x^n) &(7.7)\\
&= -\sum_i \sum_{x_i, x_{\mu(i)}} P(x_i, x_{\mu(i)}) \ln P_T(x_i|x_{\mu(i)}) - H(\underline{x}^n) &(7.8)\\
&= -\sum_i \sum_{x_{\mu(i)}} P(x_{\mu(i)}) \left[\sum_{x_i} P(x_i|x_{\mu(i)}) \ln P_T(x_i|x_{\mu(i)})\right] - H(\underline{x}^n) \ . &(7.9)
\end{aligned}$$

Now note that

$$\sum_{x_i} P(x_i|x_{\mu(i)}) \ln \frac{P(x_i|x_{\mu(i)})}{P_T(x_i|x_{\mu(i)})} \geq 0 \tag{7.10}$$

and this inequality becomes an equality iff

$$P(x_i|x_{\mu(i)}) = P_T(x_i|x_{\mu(i)}) \ . \tag{7.11}$$

Therefore

$$D_{KL}(P \parallel P_T) \geq -\sum_i \underbrace{\sum_{x_{\mu(i)}} P(x_{\mu(i)}) \left[\sum_{x_i} P(x_i|x_{\mu(i)}) \ln P(x_i|x_{\mu(i)})\right]}_{=H(\underline{x}_i|\underline{x}_{\mu(i)})=H(\underline{x}_i:\underline{x}_{\mu(i)})-H(\underline{x}_i)} - H(\underline{x}^n) \ , \tag{7.12}$$

and this inequality becomes an equality iff Eq.(7.11) is satisfied.

Note from the last equation that

$$\underset{\mu}{\text{argmin}}\, D_{KL}(P \parallel P_T) = \underset{\mu}{\text{argmax}} \sum_i H(\underline{x}_i : \underline{x}_{\mu(i)}) \ . \tag{7.13}$$

QED

Claim 4

$$\underset{\mu}{\text{argmin}}\, H(\underline{x}^n) = \underset{\mu}{\text{argmax}} \sum_i H(\underline{x}_i : \underline{x}_{\mu(i)}) \tag{7.14}$$

proof:

$$\begin{aligned}
H(\underline{x}^n) &= -\sum_{x^n} P(x^n) \sum_i \ln P(x_i|x_{\mu(i)}) && (7.15)\\
&= -\sum_i \sum_{x_i, x_{\mu(i)}} P(x_i, x_{\mu(i)}) \ln P(x_i|x_{\mu(i)}) && (7.16)\\
&= -\sum_i \sum_{x_i, x_{\mu(i)}} P(x_i, x_{\mu(i)}) \left[\ln \frac{P(x_i|x_{\mu(i)})}{P(x_i)} + \ln P(x_i)\right] && (7.17)\\
&= -\sum_i \left[H(\underline{x}_i : \underline{x}_{\mu(i)}) - H(\underline{x}_i)\right] && (7.18)\\
&= \sum_i H(\underline{x}_i) - \sum_i H(\underline{x}_i : \underline{x}_{\mu(i)}) && (7.19)
\end{aligned}$$

QED

The meaning of Claims 3 and 4 is as follows. If $D_{KL}(P \parallel P_T)$ is minimized over all P_T, then

1. P_T inherits its TPM's from P, and
2. P_T gets its structure, which is being parameterized by the function μ, by maximizing the score given by

$$\text{score} = \sum_i H(\underline{x}_i : \underline{x}_{\mu(i)}) \ . \qquad (7.20)$$

(mutual information $H(\underline{a} : \underline{b})$ measures correlation between $\underline{a}$ and $\underline{b}$). Maximizing the score is the same as minimizing the entropy $H(\underline{x}^n)$ over all the structures μ. (i.e., finding least complex structure).

So far, we have studied the properties of those probability distributions P_T for a tree bnet that best approximates an a priori given probability distribution P, but we haven't yet described how to build a Chow-Liu tree based on empirical data. Next we give Chow-Liu's algorithm for doing so.

1. **Find MST using Kruskal's algorithm[1]. (see Fig.7.1)**
Calculate weights $w_{i,j} = H(\underline{x}_i : \underline{x}_j)$ for all $i, j \in V$ and store them in a dictionary D that maps edges to weights.
Order D by weight size.

[1] Kruskal's algorithm is one several famous algorithms (Prim's algo is another one) for finding an MST (maximum or minimum spanning tree). An MST algorithm takes an undirected graph with weights along its edges as input. It then finds a tree subgraph (i.e., subset of the edges of the graph with no loops) that (1) spans the graph (i.e., includes every vertex of the graph) and (2) maximizes (or minimizes) the sum of weights among all possible tree subgraphs. For more information, see Ref[53] and references therein, or any other of numerous explanations of MST in the Internet.

```
Let $T$ be a list of the edges in the tree. Initialize $T$ to empty.
Repeat this until $T$ has $n-1$ elements:
    Remove largest weight $w$ from $D$ and corresponding edge $e$.
    Add $e$ to $T$ if $\{e\} \cup T$ has no loops. Otherwise discard $e$ and $w$.
```

2. **Give directions to edges in T. (see Fig.7.2)**

```
Let $DT$ be a list of directed edges. Initialize $DT$ to empty.
Choose any node as root node.
Point arrows along edges in $T$, away from root node.
Add new arrows to $DT$.
Repeat this until $DT$ has $n-1$ elements:
    Point arrows along edges in $T$, away from leaf nodes of current $DT$.
    Add new arrows to $DT$.
```

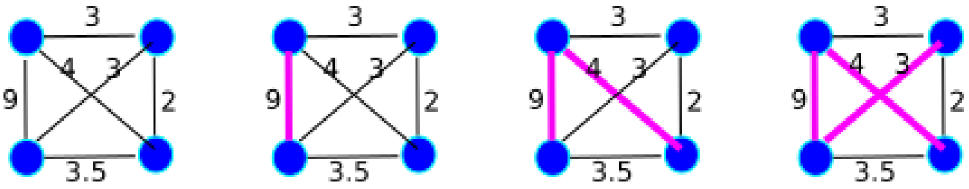


Figure 7.1: Example of finding MST (maximum spanning tree)

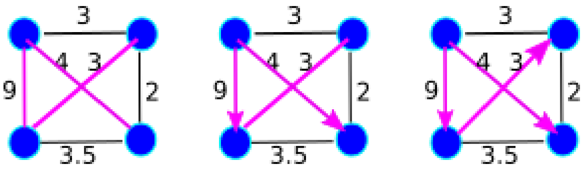


Figure 7.2: Example of giving directions to edges of spanning tree.

Nodes in a Chow-Liu tree can be rated in terms of their relative importance. Here are 2 possible metrics for measuring the importance of a node $\underline{a}$:

$$N_{nb}(\underline{a}) = \text{ number of neighbors of } \underline{a} \tag{7.21}$$

$$\text{traffic}(\underline{a}) = \sum_{\underline{n} \in nb(\underline{a})} H(\underline{a} : \underline{n}) \tag{7.22}$$

For example, to get a tree with low depth, one can choose as the root node the node which has largest N_{nb}, and if there are several with the same largest N_{nb}, choose out of those the one with the largest traffic.

Tree Augmented Naive Bayes (TAN)

Recall from Chapter 30 that a Naive Bayes bnet consists of a class node $\underline{c}$ with n children nodes $\underline{x}^n$, called the feature nodes. A Tree Augmented Naive Bayes (TAN) bnet is a Naive Bayes bnet with a tree grafted onto it like a chimera. More precisely, one starts with a Naive Bayes bnet and adds arrows between the feature nodes. The arrows are added in such a way that the TAN bnet sans node $\underline{c}$ constitutes a tree. It's not the most well motivated bnet in human history, but at least it adds a bit of correlation between the feature nodes of the Naive Bayes bnet. Those nodes are independent at fixed $\underline{c}$ in the Naive Bayes bnet, but are no longer so in the TAN bnet. See Figs.7.3 and 7.4 for an example of a TAN bnet.

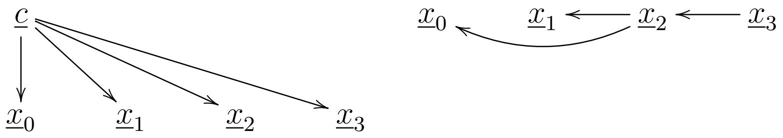


Figure 7.3: bnet for Naive Bayes with 4 feature nodes and another bnet for a tree made of the same feature nodes.

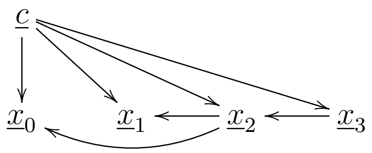


Figure 7.4: TAN bnet constructed by merging Naive Bayes bnet and tree bnet of Fig.7.3.

The total probability distribution P_{TAN} for a TAN bnet can be parameterized as follows.

$$P_{TAN}(x^n, c) = P_{TAN}(c) \prod_{i=0}^{n-1} P_{TAN}(x_i | x_{\mu(i)}, c) \; . \tag{7.23}$$

As with Chow Liu trees, we can attempt to find a TAN bnet whose total probability $P_{TAN} = P_{\underline{t}^n, \underline{c}}$ best approximates an a priori given probability distribution $P = P_{\underline{x}^n, \underline{c}}$.

Note that

Claim 5

$$\operatorname*{argmin}_{\mu} H(\underline{x}^n, \underline{c}) = \operatorname*{argmax}_{\mu} \sum_i H(\underline{x}_i : \underline{x}_{\mu(i)} | \underline{c}) \tag{7.24}$$

proof:

$$
\begin{aligned}
H(\underline{x}^n, \underline{c}) &= -\sum_{x^n,c} P(x^n, c)\left[\ln P(c) + \sum_i \ln P(x_i|x_{\mu(i)}, c)\right] && (7.25)\\
&= -\sum_{x^n,c} P(x^n, c)\left[\ln P(c) + \sum_i \ln\left(\frac{P(x_i, x_{\mu(i)}|c)}{P(x_i|c)P(x_{\mu(x_i)}|c)} P(x_i|c)\right)\right] && (7.26)\\
&= \sum_i H(\underline{x}_i, \underline{c}) - \sum_i H(\underline{x}_i : \underline{x}_{\mu(i)}|\underline{c}) && (7.27)
\end{aligned}
$$

QED

Following the same line of reasoning that we followed for Chow-Liu trees, we conclude that: If $D_{KL}(P \parallel P_{TAN})$ is minimized over all P_{TAN}, then

1. P_{TAN} inherits its TPM's from P, and
2. P_{TAN} gets its structure, which is being parameterized by the function μ, by maximizing the score defined by

$$
\text{score} = \sum_i H(\underline{x}_i : \underline{x}_{\mu(i)}|\underline{c}) \tag{7.28}
$$

One can build a TAN bnet from empirical data as follows:

Calculate a Chow-Liu Tree for each $c \in S_{\underline{c}}$. For each of those trees, create a TAN bnet, and calculate its score given by Eq.(7.28). Keep the TAN bnet with the largest score.

Chapter 8

Counterfactual Reasoning

This chapter is mostly based on Ref.[20], a 2019 review of causality by Pearl.

This chapter assumes that the reader has read Chapter 11 on do-calculus and Chapter 23 on LDEN (linear deterministic systems with external noise).

The 3 Rungs of Causal AI

According to Judea Pearl, there are 3 rungs in the ladder of causal AI. These are (as I see them):

1. **Observing Dumbly:** Collecting data and fitting curves to it, without any plan designed to investigate Nature's causal connections.

2. **Doing causal experiments:** Doing experiments consciously designed to elucidate Nature's causal connections. Even cats do this!, but current AI doesn't.

3. **Imagining counterfactual situations, Analogizing:** Imagining gedanken experiments to further understand Nature's causal connections, and to decide what future courses of action are more likely to succeed, even if there is zero direct data for those courses of action. Making predictions based on zero direct data is a very Bayesian concern, well out of the purview of frequentists. Nevertheless, humans do such "analogizing" all the time to great advantage. It becomes possible if there is some indirect but similar data that can be transported (transplanted, applied) to the situation of interest.

Chapter 27 on message passing is about rung 1. Chapter 11 on do-calculus is about rung 2. This chapter is dedicated to rung 3.

Two kinds of intervention operators

In Chapter 11, we introduced a **do operator** $\rho_{\underline{x}=x}$ (this is our notation for what Pearl symbolizes by $do(\underline{x}) = x$). The study of counterfactuals requires that we introduce a new kind of intervention operator that we will call an **imagine operator** and denote by $\kappa_{\underline{x}\to\underline{a}}(x)$.

The 2 types of intervention operators are defined graphically in Fig.8.1.

- The do operator $\rho_{\underline{x}=5}$ (called ρ because it turns $\underline{x}$ into a root node) amputates the incoming arrows of node $\underline{x}$ and sets the TPM of the new root node $\underline{x}$ to a delta function $\delta(x, 5)$ (or some state of $\underline{x}$ other than 5). Sometimes it is convenient, rather than calling the new node $\underline{x}$ like the old one, to call it by the new name $\rho\underline{x}$.

- The imagine operator $\kappa_{\underline{x}\to\underline{b}}(5)$ (called κ because it creates konstant nodes) operates on arrows unlike the ρ operator which operates on nodes. $\kappa_{\underline{x}\to\underline{b}}(5)$ deletes arrow $\underline{x} \to \underline{b}$ and creates a new root node $\underline{x}'$ and a new arrow $\underline{x}' \to \underline{b}$. The TPM of the new node $\underline{x}'$ is a delta function $\delta(x', 5)$ (or some state of $\underline{x}$ other than 5). Sometimes it is convenient, rather than calling the new node $\underline{x}'$, to call it by the more explicit name $\kappa_{\underline{b}}\underline{x}$.

Now that we have both a do and an imagine operator, we realize, as Pearl did long ago, that we can create a **do-imagine-calculus** whose objective is to express probabilities such as $P(\underline{y}|\rho\underline{r} = r, \kappa_{\underline{b}}\underline{s} = s, t)$ in terms of observable probabilities that do not contain any do or imagine operators in them. As with do-calculus, this reduction is not always possible, and we say a probability is **identifiable** if it can be reduced in that manner. Such a do-imagine-calculus has already been developed by Pearl and collaborators, but we won't discuss it in this chapter (perhaps we will discuss it in a future one).

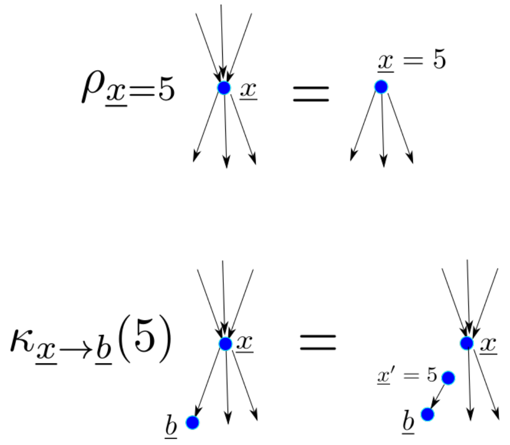


Figure 8.1: Action of "do" operator $\rho_{\underline{x}=5}$ on node $\underline{x}$ and of "imagine" operator $\kappa_{\underline{x}\to\underline{b}}(5)$ on arrow $\underline{x} \to \underline{b}$.

Do operator $\rho_{\underline{a}=a}$ for DEN diagrams

By the end of this chapter, the two kinds of intervention operators will be applied to DEN diagrams. Let us begin that journey by showing in this section how to apply the already familiar do operator to DEN diagrams.

Recall that the structural equations for a linear DEN, as given by Eq.(23.21) of Chapter 23, are:

$$\underline{x} = A\underline{x} + \underline{u}\ . \tag{8.1}$$

Therefore,

$$\underline{x} = (1 - A)^{-1}\underline{u} \tag{8.2}$$

which can be represented for both linear and non-linear DEN diagrams by:

$$\underline{x}_i = x_i(\underline{u}.) \tag{8.3}$$

If now we apply the operator $\rho_{\underline{a}=a}$ to the diagram described by the structural equations Eqs.8.1, we get the following new structural equations:

$$\underline{x}^*_i = \left\{ \begin{array}{ll} \sum_{j<i} A_{i|j}\underline{x}^*_j + \underline{u}_i & \text{if } \underline{x}_i \neq \underline{a} \\ a & \text{if } \underline{x}_i = \underline{a} \end{array} \right. , \tag{8.4}$$

where we are calling $\underline{x}^*_i$ the nodes of the DEN diagram post intervention.

Eqs.(8.4) can be expressed in matrix notation as follows. Define $\pi_{\underline{a}}$ to be the $nx \times nx$ matrix with all entries equal to zero except for the (i_0, i_0) entry, which is 1. And define $e_{\underline{a}}$ to be the column vector with all entries zero except for the i_0'th one, which is 1. Here i_0 is defined so that $\underline{x}_{i_0} = \underline{a}$. In other words, $\pi_{\underline{a}}$ and $e_{\underline{a}}$ are defined by

$$(\pi_{\underline{a}})_{i,j} = \mathbb{1}(i = j, \underline{a} = \underline{x}_i) \tag{8.5}$$

and

$$(e_{\underline{a}})_i = \mathbb{1}(\underline{a} = \underline{x}_i)\ , \tag{8.6}$$

for $i, j \in \{0, 1, \ldots, nx - 1\}$. Next define

$$\pi_{!\underline{a}} = 1 - \pi_{\underline{a}}\ , \tag{8.7}$$

$$A^* = \pi_{!\underline{a}}A\ , \tag{8.8}$$

and

$$\underline{u}_{!\underline{a}} = \pi_{!\underline{a}}\underline{u}\ . \tag{8.9}$$

The effect of pre-multiplying the matrix A and the column vector $\underline{u}$ by $\pi_{!\underline{a}}$ is to leave all rows intact except for the i_0 row, which is set to zero. Here i_0 is defined by $\underline{a} = \underline{x}_{i_0}$. Finally, using all of the variables just defined, we can express the structural equations of the linear DEN diagram, post intervention, as

$$\underline{x}^* = A^*\underline{x}^* + \underline{u}_{!\underline{a}} + ae_{\underline{a}}\ . \tag{8.10}$$

Thus,

$$\underline{x}^* = (1 - A^*)^{-1}(\underline{u}_{!\underline{a}} + ae_{\underline{a}}) \ . \tag{8.11}$$

which can be represented for both linear and non-linear DEN diagrams by:

$$\underline{x}_i^* = x_i^*(\underline{u}_{!\underline{a}}, a) \ . \tag{8.12}$$

For any bnet,

$$P(\underline{y} = y|\underline{x} = x) = P_G(\underline{y} = y|\underline{x} = x) \tag{8.13}$$

$$P(\underline{y} = y|\rho\underline{x} = x) = P_{\rho_{\underline{x}=x}G}(\underline{y} = y) \tag{8.14}$$

Claim 6 *For a non-linear DEN diagram,*

$$P(y|\rho\underline{x} = x) = E\left[\delta[y, y(\underline{u}_{!\underline{x}}, x)]\right] \ . \tag{8.15}$$

proof:

$$\begin{aligned}
P(\underline{y} = y|\rho\underline{x} = x) &= P_{\rho_{\underline{x}=x}G}(\underline{y} = y) && (8.16)\\
&= \sum_{u_{!\underline{x}}} P(u_{!\underline{x}})P_{\rho_{\underline{x}=x}G}(\underline{y} = y|u_{!\underline{x}}) && (8.17)\\
&= \sum_{u_{!\underline{x}}} P(u_{!\underline{x}})\delta[y, y(u_{!\underline{x}}, x)] && (8.18)\\
&= E_{\underline{u}_{!\underline{x}}}[\delta[y, y(u_{!\underline{x}}, x)]] && (8.19)\\
&= E[\delta[y, y(\underline{u}_{!\underline{x}}, x)]] && (8.20)
\end{aligned}$$

QED

Claim 7 *For a nonlinear DEN diagram,*

$$E[\underline{y}|\rho\underline{x} = x] = E[y(\underline{u}_{!\underline{x}}, x)] \ . \tag{8.21}$$

proof:

$$\begin{aligned}
E[\underline{y}|\rho\underline{x} = x] &= \sum_{y} yP(\underline{y} = y|\rho\underline{x} = x) && (8.22)\\
&= \sum_{y} yE[\delta[y, y(u_{!\underline{x}}, x)]] && (8.23)\\
&= E[y(\underline{u}_{!\underline{x}}, x)] && (8.24)
\end{aligned}$$

QED

For any bnet

$$P(y|\rho\underline{x}=x,z) \;=\; \frac{P(y,z|\rho\underline{x}=x)}{P(z|\rho\underline{x}=x)} = P_{\rho_{\underline{x}=x}G}(y|x,z) \tag{8.25}$$

For a nonlinear DEN diagram,

$$P(y,z|\rho\underline{x}=x) = \sum_{u_{!\underline{x}}} P(u_{!\underline{x}})\delta[y,y(u_{!\underline{x}},x)]\delta[z,z(u_{!\underline{x}},x)] \tag{8.26}$$

$$P(z|\rho\underline{x}=x) = \sum_{u_{!\underline{x}}} P(u_{!\underline{x}})\delta[z,z(u_{!\underline{x}},x)]\;. \tag{8.27}$$

Mediation Analysis

In the previous section, we applied the do operator to DEN diagrams. Mediation analysis is a nice example which applies both do and imagine operators to DEN diagrams.

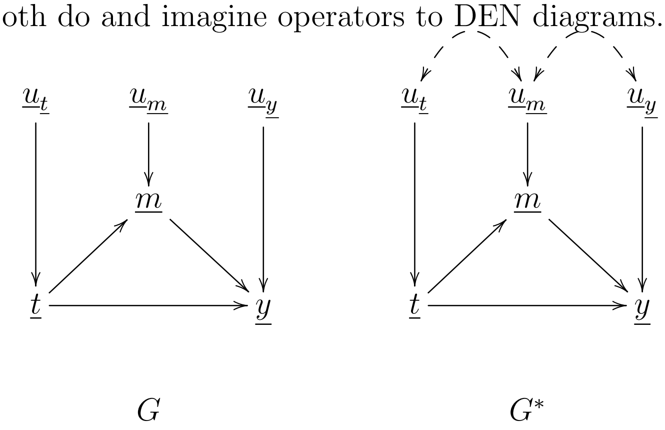

$\underline{u}_{\underline{t}}$
$\underline{u}_{\underline{m}}$
$\underline{u}_{\underline{y}}$
$\underline{m}$
$\underline{t}$
$\underline{y}$
G
$\underline{u}_{\underline{t}}$
$\underline{u}_{\underline{m}}$
$\underline{u}_{\underline{y}}$
$\underline{m}$
$\underline{t}$
$\underline{y}$
G^*

Figure 8.2: Graphs G and G^* are used to discuss mediation. In graph G, the exogenous variables are independent, whereas in graph G^* they are not.

The term "mediation analysis" refers to the analysis of the DEN diagram G in Fig.8.2. In that diagram, node $\underline{t}$ influences node $\underline{y}$ both directly and via the mediator node $\underline{m}$. The structural equations for that diagram are of the form:

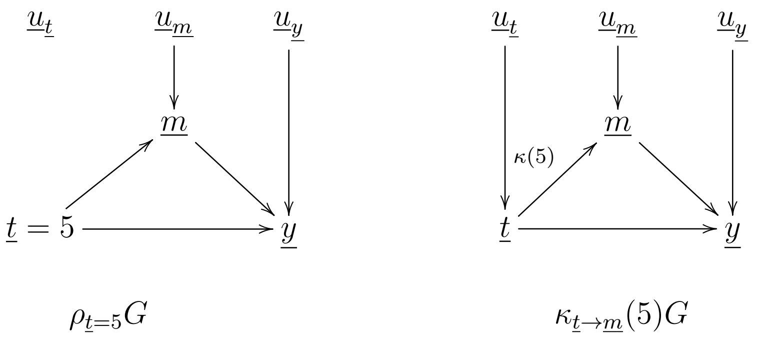
$$\underline{t} \;=\; \underline{u}_{\underline{t}} \tag{8.28a}$$

$$\underline{m} \;=\; f_{\underline{m}}(\underline{t},\underline{u}_{\underline{m}}) \tag{8.28b}$$

$$\underline{y} \;=\; f_{\underline{y}}(\underline{t},\underline{m},\underline{u}_{\underline{y}})\;. \tag{8.28c}$$

Thus,

$$\underline{y} = f_{\underline{y}}(u_{\underline{t}}, f_{\underline{m}}(\underline{u}_{\underline{t}},\underline{u}_{\underline{m}}),\underline{u}_{\underline{y}})\;. \tag{8.29}$$


$\underline{u}_{\underline{t}}$
$\underline{u}_{\underline{m}}$
$\underline{u}_{\underline{y}}$
$\underline{m}$
$\underline{t} = 5$
$\underline{y}$
$\underline{u}_{\underline{t}}$
$\underline{u}_{\underline{m}}$
$\underline{u}_{\underline{y}}$
$\kappa(5)$
$\underline{m}$
$\underline{t}$
$\underline{y}$

$\rho_{\underline{t}=5}G$ $\kappa_{\underline{t}\rightarrow\underline{m}}(5)G$

Figure 8.3: Graph G of Fig.8.2 after applying do operator $\rho_{\underline{t}=5}$ and imagine operator $\kappa_{\underline{t}\rightarrow\underline{m}}(5)$.

If we apply $\rho_{\underline{t}=5}G$ to Eqs.(8.28), we get

$$\begin{aligned}\underline{t} &= 5 && (8.30a)\\ \underline{m} &= f_{\underline{m}}(\underline{t}, \underline{u}_{\underline{m}}) && (8.30b)\\ \underline{y} &= f_{\underline{y}}(\underline{t}, \underline{m}, \underline{u}_{\underline{y}}) \; . && (8.30c)\end{aligned}$$

Eqs.8.30 are represented graphically in Fig.8.3. We will often denote the random variable $\underline{y}$ in Eqs.(8.30) by the more explicit symbol $\underline{y}_{\rho_{\underline{t}=5}G}$. Pearl often refers to this $\underline{y}$ by the less explicit symbol Y_5 or $Y_5(u)$ where $Y = \underline{y}$ and $u = (u_{\underline{m}}, u_{\underline{y}}) = u_{!\underline{t}}$.

If we apply $\kappa_{\underline{t}\rightarrow\underline{m}}(5)G$ to Eqs.(8.28), we get

$$\begin{aligned}\underline{t} &= \underline{u}_{\underline{t}} && (8.31a)\\ \underline{m} &= f_{\underline{m}}(5, \underline{u}_{\underline{m}}) && (8.31b)\\ \underline{y} &= f_{\underline{y}}(\underline{t}, \underline{m}, \underline{u}_{\underline{y}}) \; . && (8.31c)\end{aligned}$$

Eqs.8.31 are represented graphically in Fig.8.3. We will often denote the random variable $\underline{y}$ in Eqs.(8.31) by the more explicit symbol $\underline{y}_{\kappa_{\underline{t}\rightarrow\underline{m}}(5)G}$. Pearl often refers to this $\underline{y}$ by the less explicit symbol Y_5 or $Y_5(u)$ where $Y = \underline{y}$ and $u = (u_{\underline{t}}, u_{\underline{m}}, u_{\underline{y}})$.

Define the Total Effect (TE), and the Controlled Direct Effect (CDE) by

$$\begin{aligned}TE &= E[\underline{y}_{\rho_{\underline{t}=1}G} - \underline{y}_{\rho_{\underline{t}=0}G}] && (8.32)\\ CDE(m) &= E[\underline{y}_{\rho_{\underline{t}=1}\rho_{\underline{m}=m}G} - \underline{y}_{\rho_{\underline{t}=0}\rho_{\underline{m}=m}G}] && (8.33)\end{aligned}$$

The two DEN diagrams $\rho_{\underline{t}=t}G$ and $\rho_{\underline{t}=t}\rho_{\underline{m}=m}G$ used in the definitions of TE and CDE are given in Fig.8.4.

Let

$$E_a^b = E[\underline{y}_{\kappa_{\underline{t}\rightarrow\underline{y}}(a)\kappa_{\underline{t}\rightarrow\underline{m}}(b)G}] \qquad (8.34)$$

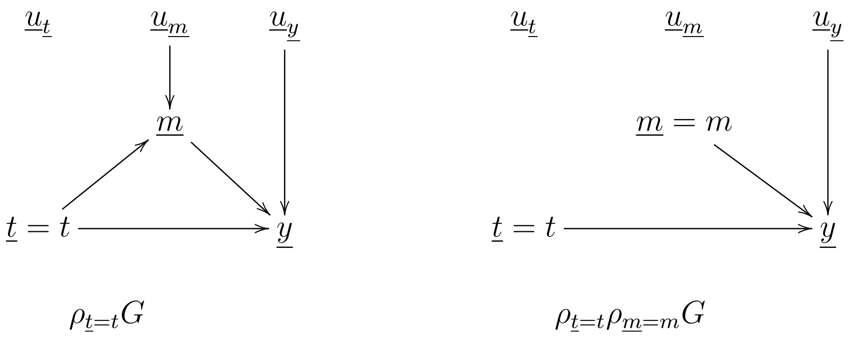

$\underline{u}_{\underline{t}}$
$\underline{u}_{\underline{m}}$
$\underline{u}_{\underline{y}}$
$\underline{m}$
$\underline{t} = t$
$\underline{y}$
$\rho_{\underline{t}=t}G$
$\underline{u}_{\underline{t}}$
$\underline{u}_{\underline{m}}$
$\underline{u}_{\underline{y}}$
$\underline{m} = m$
$\underline{t} = t$
$\underline{y}$
$\rho_{\underline{t}=t}\rho_{\underline{m}=m}G$

Figure 8.4: Graph G of Fig.8.2 after applying the do operators $\rho_{\underline{t}=t}$ and $\rho_{\underline{t}=t}\rho_{\underline{m}=m}$.

$\kappa_{\underline{t}\to\underline{y}}(a)\kappa_{\underline{t}\to\underline{m}}(b)G =$

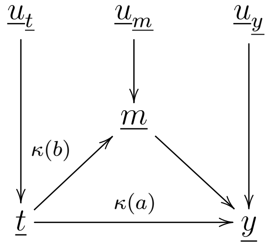

$\underline{u}_{\underline{t}}$
$\underline{u}_{\underline{m}}$
$\underline{u}_{\underline{y}}$
$\underline{m}$
$\kappa(b)$
$\kappa(a)$
$\underline{t}$
$\underline{y}$

Figure 8.5: Graph G of Fig.8.2 after applying the imagine operator κ to arrows $\underline{t} \to \underline{m}$ and $\underline{t} \to \underline{y}$.

Fig.8.5 shows the diagram $\kappa_{\underline{t}\to\underline{y}}(a)\kappa_{\underline{t}\to\underline{m}}(b)G$ used in the definition of E_a^b.

Now define the Natural Direct Effect (NDE), and the Natural Indirect Effect (NIE) by

$$\begin{aligned} NDE &= E_1^0 - E_0^0 & (8.35)\\ NIE(t) &= E_t^1 - E_t^0\ . & (8.36)\end{aligned}$$

Note that

$$\begin{aligned} NDE + NIE(1) &= (E_1^0 - E_0^0) + (E_1^1 - E_1^0) & (8.37)\\ &= E_1^1 - E_0^0 & (8.38)\\ &= TE\ . & (8.39)\end{aligned}$$

Chapter 11

Do-Calculus

The do-calculus and associated ideas were invented by Judea Pearl and collaborators. This chapter is based on Judea Pearl's books. (See 0.4).

When doing do-calculus, it is convenient to separate the nodes of a bnet into 2 types: **visible (observed)**, and **non-visible (not observed, latent, hidden)**, depending on whether data describing the state of that node is available (visible) or not (non-visible). In this chapter, hidden nodes will be indicated in a bnet diagram by either: (1) enclosing their random variable in a box (as if it were inside a black box) or (2) making the arrows coming out of them dashed. Accordingly, the 3 diagrams in Fig.11.1 all mean the same thing.

A **confounder node for $\underline{x}$ and $\underline{y}$** (such as node $\underline{c}$ in Fig.11.1) is a hidden node with arrows pointing from it to both $\underline{x}$ and $\underline{y}$. In other words, it's an unobserved common cause of $\underline{x}$ and $\underline{y}$.

In this book, we will refer to a path all of whose nodes are observed as an **opath**.

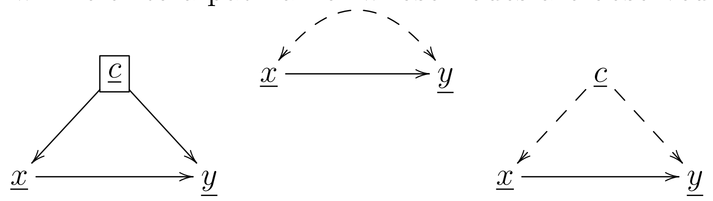


Figure 11.1: These 3 diagrams are equivalent. They mean that node $\underline{c}$ is hidden. Node $\underline{c}$ is implicit in the middle diagram.

Define an operator $\rho_{\underline{x}}$ that acts on a node $\underline{x}$ of a bnet to delete all the arrows entering $\underline{x}$, thus coverting $\underline{x}$ into a new node $\rho\underline{x}$ that is a root node. Define an analogous operator $\lambda\underline{x}$ that acts on a node $\underline{x}$ of a bnet to delete all the arrows leaving $\underline{x}$, thus converting $\underline{x}$ into a new node $\lambda\underline{x}$ that is a leaf node. $\rho_{\underline{x}}$ and $\lambda_{\underline{x}}$ are depicted in Fig.11.2.

If you don't know yet what we mean by a a multi-node $\underline{a}.$, see Chapter 0.2

Given a bnet G, we define as follows the operators $\rho_{\underline{a}.}$ and $\lambda_{\underline{a}.}$ for a multi-node $\underline{a}.$.

$$\rho_{\underline{a}.}G = \left[\prod_j \rho_{\underline{a}_j}\right] G \;, \quad \lambda_{\underline{a}.}G = \left[\prod_j \lambda_{\underline{a}_j}\right] G \;. \tag{11.1}$$

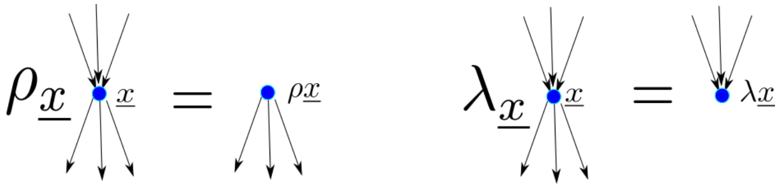


Figure 11.2: The operator $\rho_{\underline{x}}$ converts node $\underline{x}$ into a root node $\rho\underline{x}$. The operator $\lambda_{\underline{x}}$ converts node $\underline{x}$ into a leaf node $\lambda\underline{x}$.

Consider a bnet whose totality of nodes is labeled $\underline{X}.$. Recall that

$$P(X.) = \prod_j P(X_j | (X_k)_{k:\underline{X}_k \in pa(\underline{X}_j)}) \; . \tag{11.2}$$

Define an operator ρ that acts as follows[1]: Let $X. - a. = (X_k)_{k:\underline{X}_k \notin \underline{a}.}$.

$$\begin{aligned}
P(X. - a. | \rho\underline{a}. = a.) &= \mathcal{N}(!(X. - a.)) \frac{P(X.)}{\prod_{j:\underline{X}_j \in \underline{a}.} P(X_j | (X_k)_{k:\underline{X}_k \in pa(\underline{X}_j)})} &(11.3)\\
&= \mathcal{N}(!(X. - a.)) \prod_{j:\underline{X}_j \notin \underline{a}.} P(X_j | (X_k)_{k:\underline{X}_k \in pa(\underline{X}_j)}) &(11.4)\\
&\neq P(X. - a. | \underline{a}. = a.) \; . &(11.5)
\end{aligned}$$

Also,

$$P(X. - a., \rho\underline{a}. = a.') = P(X. - a. | \rho\underline{a}. = a.) \delta(a'., a.) \; . \tag{11.6}$$

In words, we replace the TPM for multinode $\underline{a}.$ by a deterministic prior distribution.

For instance, for the bnet

$$\underline{x} \longrightarrow \underline{y} \tag{11.7}$$

with

$$P(x, y) = P(y|x)P(x) \; , \tag{11.8}$$

one has

$$P(y | \rho\underline{x} = x) = P(y|x) \tag{11.9}$$

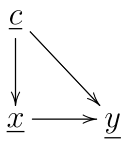
and

$$P(x | \rho\underline{y} = y) = P(x) \; . \tag{11.10}$$

[1] As usual, $\mathcal{N}(!x)$ denotes a constant that is independent of x.

This means that $\underline{x}$ causes $\underline{y}$ and $\underline{y}$ does not cause $\underline{x}$.

For the bnet



(11.11)

with

$$P(x, y, c) = P(y|x, c)P(x|c)P(c) \;, \tag{11.12}$$

one has

$$P(y, c|\rho\underline{x} = x) = P(y|x, c)P(c) \;. \tag{11.13}$$

Hence,

$$P(y|\rho\underline{x} = x) = \sum_c P(y|x, c)P(c) \;. \tag{11.14}$$

This is called **adjusting the parents of $\underline{x}$**.

For $\underline{b}. \subset \underline{X}. - \underline{a}.$, define

$$P(b.|\rho\underline{a}. = a.) = \sum_{X.-a.-b.} P(X. - a.|\rho\underline{a}. = a.) \;, \tag{11.15}$$

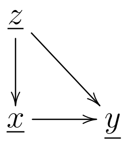
and for $\underline{s}. \subset \underline{X}. - \underline{a}. - \underline{b}.$, define

$$P(b.|\rho\underline{a}. = a., s.) = \frac{P(b., s.|\rho\underline{a}. = a.)}{P(s.|\rho\underline{a}. = a.)} \;. \tag{11.16}$$

$P(b.|\rho\underline{a}. = a., s.)$ is usually denoted instead by $P(b.|do(\underline{a}. = a.), s.)$. I prefer to use ρ instead of $do()$ to remind me that it generates root nodes. I'll still call ρ a **do operator**.

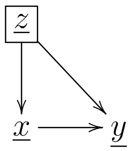
In $P(y|\rho\underline{x} = x)$, node $\underline{x}$ is turned into a root node. This guarantees that there is no confounding node connecting $\underline{x}$ and $\underline{y}$. Such confounding nodes are unwelcomed when calculating causal effects between the 2 variables $\underline{x}$ and $\underline{y}$ because they introduce non-causal correlations between the two. This is also what happens in a **Randomized Clinical Trial (RCT)**. In a RCT with treatment $\underline{x}$, the value of $\underline{x}$ for each patient is determined by a coin toss, effectively turning $\underline{x}$ into a root node. Hence, the do operator mimics a RCT.

$P(b.|\rho\underline{a}. = a., s.)$ is said to be **identifiable** if it can be expressed in terms of probability distributions that only depend on observed variables and that have no do operators in them. For example, $P(y|\rho\underline{x} = x)$ is identifiable for the bnet



(11.17)

but it is non-identifiable for the bnet



(11.18)

For $\underline{x}, \underline{y} \in \{0,1\}$, the **causal effect difference** , or **average causal effect (ACE)** is defined as

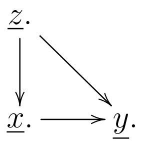
$$ACE = P(y=1|\rho\underline{x}=1) - P(y=1|\rho\underline{x}=0) \tag{11.19}$$

and the **Risk Difference (RD)** as

$$RD = P(y=1|\underline{x}=1) - P(y=1|\underline{x}=0)\ . \tag{11.20}$$

Parent Adjustment

Suppose that $\underline{x}., \underline{y}., \underline{z}.$ are disjoint multinodes and their union equals the totality of all nodes of a bnet. Suppose we have data available that allows us to estimate $P(x., y., z.)$. Hence, all nodes of the bnet are observable. Furthermore, suppose $\underline{z}. = pa(\underline{x}.)$. In other words, we are considering the bnet



. (11.21)

Then

$$P(y., z.|\rho\underline{x}. = x.) = P(y.|x., z.)P(z.) \tag{11.22}$$

so

$$P(y.|\rho\underline{x}. = x.) = \sum_{z.} P(y.|x., z.)P(z.)\ . \tag{11.23}$$

This is called **adjusting the parents** of $\underline{x}.$.

We say that we are **adjusting or controlling a variable** $\underline{a}$ if we condition a probability on $\underline{a}$ and then we average that probability over $\underline{a}$. More generally, we can adjust a whole multinode $\underline{a}.$ together.

Later on, we will introduce a generalization of this parent adjustment called the backdoor adjustment. In a backdoor adjustment, the adjusted multinode is not necessarily the parents of $\underline{x}.$, and $P(x., y., z.)$ need not represent the whole bnet.

3 Rules of do-calculus

Throughout this section, suppose $\underline{a}., \underline{b}., \underline{r}., \underline{s}.$ are disjoint multinodes in a bnet G.

Recall from Chapter 12 on d-separation, that $(\underline{b}. \perp_G \underline{a}.|\underline{r}., \underline{s}.)$ means that we have established from the d-separation rules that that all paths in G from $\underline{a}.$ to $\underline{b}.$ are blocked if we condition on $\underline{r}. \cup \underline{s}..$ Recall also that:

- **Rule 0:** Insertion or deletion of observations, without do operators. $(\underline{a}. = a. \leftrightarrow 1\ )$

  If $(\underline{b}. \perp_G \underline{a}.|\underline{r}., \underline{s}.)$, then $P(b.|a., r., s.) = P(b.|r., s.)$

The 3 rules of do-calculus can be presented in the same format.

- **Rule 1:** Insertion or deletion of observations $(\underline{a}. = a. \leftrightarrow 1\ )$

  If $(\underline{b}. \perp \underline{a}.|\underline{r}., \underline{s}.)$ in $\rho_{\underline{r}.}G$, then $P(b.|a., \rho\underline{r}. = r., s.) = P(b.|\rho\underline{r}. = r., s.)$.

- **Rule 2:** Action or observation exchange $(\rho\underline{a}. = a. \leftrightarrow \underline{a}. = a.)$

  If $(\underline{b}. \perp \underline{a}.|\underline{r}., \underline{s}.)$ in $\lambda_{\underline{a}.}\rho_{\underline{r}.}G$, then $P(b.|\rho\underline{a}. = a., \rho\underline{r}. = r., s.) = P(b.|a., \rho\underline{r}. = r., s.)$.

- **Rule 3:** Insertion and deletion of actions $(\rho\underline{a}. = a. \leftrightarrow 1)$

  If $(\underline{b}. \perp \underline{a}.|\underline{r}., \underline{s}.)$ in $\rho_{\underline{a}.-an(\underline{s}.)}\rho_{\underline{r}.}G$, then $P(b.|\rho\underline{a}. = a., \rho\underline{r}. = r., s.) = P(b.|\rho\underline{r}. = r., s.)$.

These rules have been proven to be sufficient for removing all do operators from an expression for which it is possible to do so.

Next we discuss two theorems that can be proven using do-calculus: the backdoor and the front-door adjustment theorems.

The backdoor theorem adjusts one multinode and the front-door theorem adjusts two.

Backdoor Adjustment

See Chapter 2 for examples of the use of the backdoor adjustment theorem. In this section, we shall mainly be concerned with proving this theorem using do-calculus.

For any two disjoint multinodes $\underline{x}.$ and $\underline{y}.$, we define a **backdoor path** from $\underline{x}.$ to $\underline{y}.$ as a path from $\underline{x}.$ and $\underline{y}.$ that starts with an arrow pointing into $\underline{x}.$,

Suppose that we have access to data that allows us to estimate a probability distribution $P(x., y., z.)$. Hence, the variables $\underline{x}., \underline{y}., \underline{z}.$ are all observed (i.e, not hidden). Then we say that the backdoor $\underline{z}.$ satisfies the **backdoor adjustment criterion** relative to $(\underline{x}., \underline{y}.)$ if

1. All backdoor paths from $\underline{x}.$ to $\underline{y}.$ are blocked by $\underline{z}..$
2. $\underline{z}. \cap de(\underline{x}.) = \emptyset$.

Motivation for BD criterion: Part 1 rules out paths from $\underline{x}$ to $\underline{y}$ containing a fork node (confounder) which, if not blocked by $\underline{z}.$, would introduce a non-causal correlation (confounder bias). Part 2 rules out a directed path from $\underline{x}$ to $\underline{y}$ that has a mediator node blocked by $\underline{z}.$ or a collider node unblocked by $\underline{z}.$.

Claim 8 *Backdoor Adjustment Theorem*

If $\underline{z}.$ satisfies the backdoor criterion relative to $(\underline{x}., \underline{y}.)$, then

$$P(y.|\rho\underline{x}. = x.) = \sum_{z.} P(y.|x., z.)P(z.) \tag{11.24}$$

$$= \sum_{z.} \left\{ \begin{tikzcd} \underline{z}. = z. \arrow[rd] & \\ \underline{x}. = x. \arrow[r] & \underline{y}. \end{tikzcd} \right\} \tag{11.25}$$

proof:

For simplicity, let us omit the dots from the multinodes. If z satisfies the backdoor criterion relative to $(\underline{x}, \underline{y})$, then $\underline{x}, \underline{y}, \underline{z}$ must have the following structure.

$$\begin{tikzcd} \underline{z} \arrow[d] \arrow[rd] & \\ \underline{x} \arrow[r] & \underline{y} \end{tikzcd} \tag{11.26}$$

$P(y|\rho\underline{x} = x) =$

$= \sum_m P(y|\rho\underline{x} = x, z)P(z|\rho\underline{x} = x)$

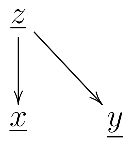
by Probability Axioms

$= \sum_P (y|x, z)P(z|\rho\underline{x} = x)$

$P(y|\rho\underline{x} = x, z) \to P(y|x, z)$

by Rule 2: If $(\underline{b}. \perp \underline{a}.|\underline{r}., \underline{s}.)$ in $\lambda_{\underline{a}.}\rho_{\underline{r}.}G$, then $P(b.|\rho\underline{a}. = a., \rho\underline{r}. = r., s.) = P(b.|a., \rho\underline{r}. = r., s.)$.

$\underline{y} \perp \underline{x}|\underline{z}$ in $\lambda_{\underline{x}}G$

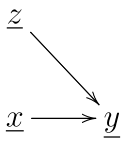


$= \sum_z P(y|x, z)P(z)$

$P(z|\rho\underline{x} = x) \to P(z)$

by Rule 3: If $(\underline{b}. \perp \underline{a}.|\underline{r}., \underline{s}.)$ in $\rho_{\underline{a}.-an(\underline{s}.)}\rho_{\underline{r}.}G$, then $P(b.|\rho\underline{a}. = a., \rho\underline{r}. = r., s.) = P(b.|\rho\underline{r}. = r., s.)$.

$\underline{z} \perp \underline{x}$ in $\rho_{\underline{x}}G$



(11.27)

QED

Note that the backdoor adjustment formula can be written as

$$P(y.|\rho\underline{x}. = x.) = \sum_{z.} P(y.|x., z.)P(z.) \tag{11.28}$$

$$= \sum_{z.} \frac{P(y., x., z.)}{P(x.|z.)} \tag{11.29}$$

This assumes $P(x.|z.) \neq 0$ for all $x., z.$. This assumption is referred to as **positivity**, and is violated if $P(x.|z.) = \delta(x., x.(z.))$. $P(x.|z.)$ is called the **propensity score** of $x.$ given $z.$. This equation does **inverse probability weighting**. One can approximate $P(x.|z.)$ in this equation to get an approximation to $P(y|\rho\underline{x} = x)$.

Front Door Adjustment

See Chapter 15 for examples of the use of the front-door adjustment theorem. In this section, we shall mainly be concerned with proving this theorem using do-calculus.

Suppose that we have access to data that allows us to estimate a probability distribution $P(x., m., y.)$. Hence, the variables $\underline{x}., \underline{m}., \underline{y}.$ are all observed (i.e, not hidden). Then we say that the front-door $\underline{m}.$ satisfies the **front-door adjustment criterion** relative to $(\underline{x}., \underline{y}.)$ if

1. All directed paths from $\underline{x}.$ to $\underline{y}.$ are intercepted by (i.e., have a node in) $\underline{m}.$.

2. All backdoor paths from $\underline{x}.$ to $\underline{m}.$ are blocked.

3. All backdoor paths from on $\underline{m}.$ to $\underline{y}.$ are blocked by $\underline{x}..$

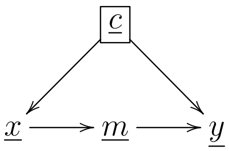
Claim 9 *Front-Door Adjustment Theorem*
If $\underline{m}.$ satisfies the front-door criterion relative to $(\underline{x}., \underline{y}.)$, and $P(x., m.) > 0$, then

$$P(y.|\rho\underline{x}. = x.) = \sum_{m.} \underbrace{\left[\sum_{x.'} P(y.|x'., m.)P(x'.)\right]}_{P(y.|\rho\underline{m}.=m.)} \underbrace{P(m.|x.)}_{P(m.|\rho\underline{x}.=x.)} \tag{11.30}$$

$$= \sum_{m.,x.'} \left\{ \begin{tikzcd} & \underline{x}. = x.' \arrow[dr] & \\ \underline{x}. = x. \arrow[r] & \underline{m}. = m. \arrow[r] & \underline{y}. \end{tikzcd} \right\} \tag{11.31}$$

proof: (See also Ref.[16] for a proof of the Front-Door Adjustment Theorem without using do-calculus.)

For simplicity, let us omit the dots from the multinodes. If $\underline{m}$ satisfies the front-door criterion relative to $(\underline{x}, \underline{y})$, then $\underline{x}, \underline{m}, \underline{y}$ must have the following structure, where node $\underline{c}$ is hidden.



(11.32)

Continues in next page.

$P(y|\rho\underline{x} = x) =$

$= \sum_m P(y|\rho\underline{x} = x, m)P(m|\rho\underline{x} = x)$

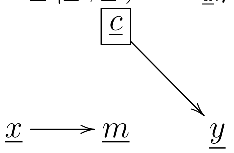
by Probability Axioms

$= \sum_m P(y|\rho\underline{x} = x, \rho\underline{m} = m)P(m|\rho\underline{x} = x)$

$P(y|\rho\underline{x} = x, m) \to P(y|\rho\underline{x} = x, \rho m = m)$

by Rule 2: If $(\underline{b}. \perp \underline{a}.|\underline{r}., \underline{s}.)$ in $\lambda_{\underline{a}.}\rho_{\underline{r}.}G$, then $P(b.|\rho\underline{a}. = a., \rho\underline{r}. = r., s.) = P(b.|a., \rho\underline{r}. = r., s.)$.

$\underline{y} \perp \underline{m}|\underline{x}$ in $\lambda_{\underline{m}}\rho_{\underline{x}}G$

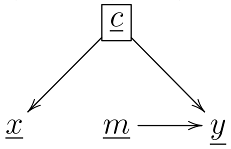


$= \sum_m P(y|\rho\underline{x} = x, \rho\underline{m} = m)P(m|x)$

$P(m|\rho\underline{x} = x) \to P(m|x)$

by Rule 2: If $(\underline{b}. \perp \underline{a}.|\underline{r}., \underline{s}.)$ in $\lambda_{\underline{a}.}\rho_{\underline{r}.}G$, then $P(b.|\rho\underline{a}. = a., \rho\underline{r}. = r., s.) = P(b.|a., \rho\underline{r}. = r., s.)$.

$\underline{m} \perp \underline{x}$ in $\lambda_{\underline{x}}G$

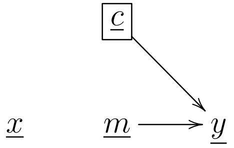


$= \sum_m P(y|\rho\underline{m} = m)P(m|x)$

$P(y|\rho\underline{x} = x, \rho\underline{m} = m) \to P(y|\rho\underline{m} = m)$

by Rule 3: If $(\underline{b}. \perp \underline{a}.|\underline{r}., \underline{s}.)$ in $\rho_{\underline{a}.-an(\underline{s}.)}\rho_{\underline{r}.}G$, then $P(b.|\rho\underline{a}. = a., \rho\underline{r}. = r., s.) = P(b.|\rho\underline{r}. = r., s.)$.

$\underline{y} \perp \underline{x}|\underline{m}$ in $\rho_{\underline{x}}\rho_{\underline{m}}G$



$= \sum_{x'} \sum_m P(y|\rho\underline{m} = m, x')P(x'|\rho\underline{m} = m)P(m|x)$

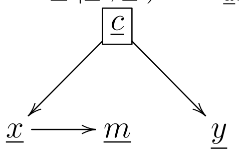
by Probability Axioms

$= \sum_{x'} \sum_m P(y|m, x')P(x'|\rho\underline{m} = m)P(m|x)$

$P(y|\rho\underline{m} = m, x') \to P(y|m, x')$

by Rule 2: If $(\underline{b}. \perp \underline{a}.|\underline{r}., \underline{s}.)$ in $\lambda_{\underline{a}.}\rho_{\underline{r}.}G$, then $P(b.|\rho\underline{a}. = a., \rho\underline{r}. = r., s.) = P(b.|a., \rho\underline{r}. = r., s.)$.

$\underline{y} \perp \underline{m}|\underline{x}$ in $\lambda_{\underline{m}}G$

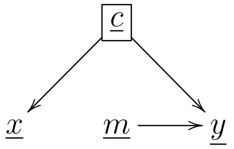


$= \sum_{x'} \sum_m P(y|m, x')P(x')P(m|x)$

$P(x'|\rho\underline{m} = m) \to P(x')$

by Rule 3: If $(\underline{b}. \perp \underline{a}.|\underline{r}., \underline{s}.)$ in $\rho_{\underline{a}.-an(\underline{s}.)}\rho_{\underline{r}.}G$, then $P(b.|\rho\underline{a}. = a., \rho\underline{r}. = r., s.) = P(b.|\rho\underline{r}. = r., s.)$.

$\underline{x} \perp \underline{m}$ in $\rho_{\underline{m}}G$



(11.33)

QED

Chapter 12

D-Separation

Before reading this chapter, I recommend that you read Chapter 0.2 on the definition of bnets.

A path γ that isn't a loop can have 3 types of intermediate nodes $\underline{x}$ (an intermediate node of γ is a node in γ that isn't one of the two end nodes). Suppose $\underline{a}$ and $\underline{b}$ are the two neighbors of $\underline{x}$. Then the 3 possible cases are:

1. **mediator node:** $(\underline{a} \leftarrow \underline{x} \leftarrow \underline{b})$ or $(\underline{a} \rightarrow \underline{x} \rightarrow \underline{b})$

2. **fork node:** $(\underline{a} \leftarrow \underline{x} \rightarrow \underline{b})$

3. **collider node:** $(\underline{a} \rightarrow \underline{x} \leftarrow \underline{b})$

We say that a non-loop path γ from $\underline{a}$ to $\underline{b}$ (i.e., with end nodes $\underline{a}, \underline{b}$) is **blocked** by a multinode $\underline{Z}.$ if one or more of the following statements is true:

1. There is a node $\underline{x} \in \underline{Z}.$ which is a mediator or a fork of γ.

2. γ contains a collider node $\underline{c}$ and $(\underline{c} \cup de(\underline{c})) \cap \underline{Z}. = \emptyset$ (i.e., neither $\underline{c}$ nor any of the descendants of $\underline{c}$ is contained in $\underline{Z}.$)

This definition of a blocked path is easy to remember if one thinks of the following analogy with pipes carrying a fluid. Think of path γ as if it were a pipe carrying a fluid. Think of the nodes of γ as junctions in the pipe. If $\underline{Z}.$ intersects γ at either a mediator or a fork junction, that blocks the pipe flow. A collider junction $\underline{c}$ is like a blackhole or a huge leak. Its presence blocks passage of the fluid as long as neither $\underline{c}$ nor any of the descendants of $\underline{c}$ are in $\underline{Z}.$. If, on the other hand, $\underline{c} \in \underline{Z}.$, or $\underline{c}' \in \underline{Z}.$ where $\underline{c}' \in de(\underline{c})$, then that acts as a complete (in the case of $\underline{c} \in \underline{Z}.$) or a partial (in the case of $\underline{c}' \in \underline{Z}.$) bridge across the blackhole.

See Fig.12.1 for some examples of paths that are blocked or not blocked by a multinode $\underline{Z}.$.

Given 3 disjoint multinodes $\underline{A}., \underline{B}., \underline{Z}.$ of a graph G, we write " $\underline{A}. \perp_G \underline{B}.|\underline{Z}.$" or say " $\underline{A}.$ **and $\underline{B}.$ are d-separated by $\underline{Z}.$ in** G" iff there exists no path γ from $\underline{a} \in \underline{A}.$, to $\underline{b} \in \underline{B}.$ which is not blocked by $\underline{Z}.$.

The minimal Markov blanket (see Chapter 24) of a node $\underline{a}$ is the smallest multinode $\underline{Z}.$ such that $\underline{a} \perp_G \underline{b}|\underline{Z}.$ for all $\underline{b} \notin \underline{a} \cup \underline{Z}.$.

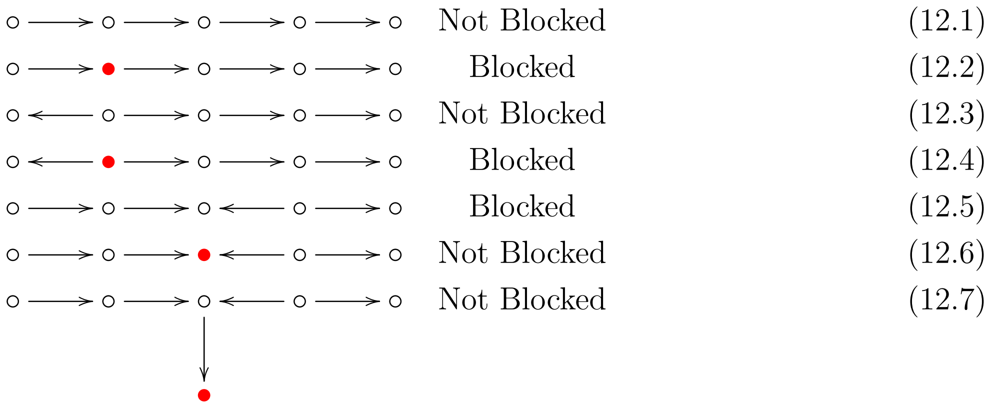


Figure 12.1: Examples of paths that are blocked or not blocked by a multinode $\underline{Z}.$. Nodes belonging to $\underline{Z}.$ are colored red.

We are finally ready to state the d-separation theorem, without proof.

A probability distribution P **is compatible with a DAG** G if P and G have the same random variables, and they can be combined to form a bnet without contradictions; i.e., one can calculate all the TPMs from P and multiply them together to obtain P again.

Claim 10 *(d-separation Theorem)*

Suppose $\underline{A}., \underline{B}., \underline{Z}.$ are disjoint multinodes of a DAG G.

If $\underline{A}. \perp_G \underline{B}.|\underline{Z}.$, then $P(B.|A., Z.) = P(B.|Z.)$ for all $B., A., Z.$, for all P compatible with G.

The full converse of the theorem can also be proven, but we won't be using it in this book.

Often, the right hand side of this theorem is stated as "$\underline{A}. \perp_P \underline{B}.|\underline{Z}.$ for all P". Then the theorem is stated: "If $\underline{A}. \perp_G \underline{B}.|\underline{Z}.$, then $\underline{A}. \perp_P \underline{B}.|\underline{Z}.$ for all P."

Note that the following are equivalent:

- $P(B.|A., Z.) = P(B.|Z.)$ for all $B., A., Z.$.
- $\underline{A}. \perp_P \underline{B}.|\underline{Z}.$
- $H(\underline{A}. : \underline{B}.|\underline{Z}.) = 0$ (see Chapter 0.3 for definition of conditional mutual information (CMI))

Extra stuff: mostly only for pure mathematicians

Below, we will use the notation $nde(\underline{a})$ to denote all nondescendants, including $\underline{a}$ itself, of a node $\underline{a}$ in a DAG G; i.e., all nodes of G that are not in $de(\underline{a}) \cup \underline{a}$, where $de(\underline{a})$ is defined in Chapter 0.2.

Given a DAG G, define the following sets of d-separations:[1]

$$DS(G) = \{(\underline{A}. \perp_G \underline{B}. \mid \underline{Z}.) : \underline{A}., \underline{B}., \underline{Z}. \text{ are multinodes of } G\} . \tag{12.8}$$

[1] Note that $(\underline{A}. \perp_G nde(\underline{A}.) \mid pa(\underline{A}.))$ and $(\underline{A}. \perp_G nde(\underline{A}.) - pa(\underline{A}.) \mid pa(\underline{A}.))$ are equivalent because $H(\underline{a} : \underline{b}, \underline{c}|\underline{c}) = H(\underline{a} : \underline{b}|\underline{c})$.

$$DS_{min}(G) = \{(\underline{A}. \perp_G nde(\underline{A}.) \,|\, pa(\underline{A}.)) : \ \underline{A}. \text{ is a multinode of } G\} \ . \tag{12.9}$$

See Chapter 34 for an example where set $DS_{min}(G)$ is calculated for a particular DAG G.

Claim 11 *For all DAGs G, $DS(G) = DS_{min}(G)$.*

Given a probability distribution P, define the following set of conditional independencies:

$$CI(P) = \{(\underline{A}. \perp_P \underline{B}. \,|\, \underline{Z}.) : \ \underline{A}., \underline{B}., \underline{Z}. \text{ are multinodes of } P\} \ , \tag{12.10}$$

For a DAG G and a probability distribution P compatible with G, define a map ϕ by

$$\begin{aligned} \phi : DS_{min}(G) \ &\to \ CI(P) && (12.11) \\ \phi : \underline{A}. \perp_G nde(\underline{A}.) \,|\, pa(\underline{A}.) \ &\mapsto \ \underline{A}. \perp_P nde(\underline{A}.) \,|\, pa(\underline{A}.) && (12.12) \end{aligned}$$

In general, this map is 1-1 but not onto.

Claim 12 *For a bnet with a DAG G and a total probability distribution P, the map ϕ is a bijection.*

$DS(G)$ does not fully specify a DAG. DAGs with the same $DS(G)$ are said to be **d-separation equivalent**. See Chapter 34 for more info about d-separation equivalence.

Chapter 15

Front-door Adjustment

The front-door (FD) adjustment theorem is proven in Chapter 11 from the rules of do-calculus. The goal of this chapter is to give examples of the use of that theorem. We will restate the theorem in this chapter, sans proof. There is no need to understand the theorem's proof in order to use it. However, you will need to skim Chapter 11 in order to familiarize yourself with the notation used to state the theorem. This chapter also assumes that you are comfortable with the rules for checking for d-separation. Those rules are covered in Chapter 12.

Suppose that we have access to data that allows us to estimate a probability distribution $P(x., m., y.)$. Hence, the variables $\underline{x}., \underline{m}., \underline{y}.$ are all observed (i.e, not hidden). Then we say that the front-door $\underline{m}.$ satisfies the **front-door adjustment criterion** relative to $(\underline{x}., \underline{y}.)$ if

1. All directed paths from $\underline{x}.$ to $\underline{y}.$ are intercepted by (i.e., have a node in) $\underline{m}.$.

2. All backdoor paths from $\underline{x}.$ to $\underline{m}.$ are blocked.

3. All backdoor paths from on $\underline{m}.$ to $\underline{y}.$ are blocked by $\underline{x}.$.

Claim 13 *Front-Door Adjustment Theorem*
If $\underline{m}.$ satisfies the front-door criterion relative to $(\underline{x}., \underline{y}.)$, and $P(x., m.) > 0$, then

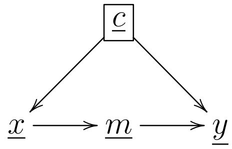
$$P(y.|\rho\underline{x}. = x.) = \sum_{m.} \underbrace{\left[\sum_{x.'} P(y.|x'., m.)P(x'.)\right]}_{P(y.|\rho\underline{m}.=m.)} \underbrace{P(m.|x.)}_{P(m.|\rho\underline{x}.=x.)} \tag{15.1}$$

$$= \sum_{m.,x.'} \left\{ \begin{tikzcd} & \underline{x}. = x.' \arrow[dr] & \\ \underline{x}. = x. \arrow[r] & \underline{m}. = m. \arrow[r] & \underline{y}. \end{tikzcd} \right\} \tag{15.2}$$

proof: See Chapter 11
QED

Examples

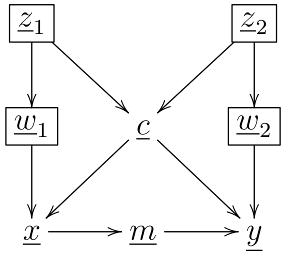
1.



(15.3)

If $\underline{x}. = \underline{x}, \underline{m}. = \underline{m}$ and $\underline{y}. = \underline{y}$, then the FD criterion is satisfied. Can't satisfy backdoor criterion because $\underline{z}.$ must be observed so can't block backdoor path $\underline{x} - \underline{c} - \underline{y}$.

2.



(15.4)

If $\underline{x}. = \underline{x}, \underline{m}. = \underline{m}$ and $\underline{y}. = \underline{y}$, then the FD criterion is satisfied. Can't satisfy backdoor criterion because to block backdoor path $\underline{x} - \underline{c} - \underline{y}$, need to condition on $\underline{c}$ (i.e., need $\underline{c} \in \underline{z}.$) but if this is true, then long path $\underline{x} - \underline{w}_1 - \underline{z}_1 - \underline{c} - \underline{z}_2 - \underline{w}_2 - \underline{y}$ becomes unblocked.

Chapter 23

Linear Deterministic Bnets with External Noise

In this chapter, we will consider bnets which were referred to, prior to the invention of bnets, as: Sewall Wright's **Path Analysis (PA)** and **linear Structural Equations Models (SEM)**. Judea Pearl in his books calls them **linear Structural Causal Models (SCM)**, because they are very convenient for doing causal analysis. We will refer to them as linear Deterministic with External Noise (LDEN) diagrams. This chapter is devoted to LDEN diagrams, except that we will say a few words about non-linear DEN diagrams at the end.

A **DEN diagram** is a special kind of bnet. To build a DEN diagram, start with a deterministic bnet G. The deterministic nodes of G are called the **endogenous (internal) variables**. Now make a bigger bnet $\overline{G}$ called a DEN diagram by adding to each node $\underline{a}$ of G a non-deterministic root node $\underline{u}_{\underline{a}}$ pointing into $\underline{a}$ only. The nodes $\underline{u}_{\underline{a}}$ are called the **exogenous (external) variables**. The exogenous variables make their children noisy. They are assumed to be unobserved and their TPMs are prior probability distributions. Since they are root nodes, they are mutually independent. When we draw a DEN diagram, we will never draw the exogenous nodes, leaving them implicit.

A **linear DEN diagram (LDEN)** is a DEN diagram whose deterministic nodes have a TPM that is a linear function of the states of the parent nodes.

Example of LDEN diagram:

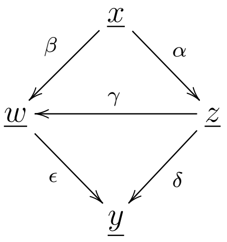


Figure 23.1: Example of a LDEN diagram wherein $\underline{x}$ splits into two nodes $\underline{z}$ and $\underline{w}$, then merges into node $\underline{y}$. There is also an arrow $\underline{z} \rightarrow \underline{w}$. Exogenous nodes are not shown. The Greek letters represent real numbers.

The TPMs, printed in blue, for the nodes of the LDEN diagram Fig.23.1, are as follows.

$$P(y|w,z,u_{\underline{y}}) = \mathbb{1}(y = \epsilon w + \delta z + u_{\underline{y}}) \tag{23.1}$$

$$P(w|x,z,u_{\underline{w}}) = \mathbb{1}(w = \beta x + \gamma z + u_{\underline{w}}) \tag{23.2}$$

$$P(z|x,u_{\underline{z}}) = \mathbb{1}(z = \alpha x + u_{\underline{z}}) \tag{23.3}$$

$$P(x|u_{\underline{x}}) = \mathbb{1}(x = u_{\underline{x}}) \tag{23.4}$$

Hence,

$$\begin{aligned} y &= \epsilon w + \delta z + u_{\underline{y}} \qquad (23.5) \\ &= \epsilon(\beta x + \gamma z + u_{\underline{w}}) + \delta z + u_{\underline{y}} \qquad (23.6) \\ &= (\epsilon\gamma + \delta)z + \epsilon\beta x + \epsilon u_{\underline{w}} + u_{\underline{y}} \qquad (23.7) \\ &= (\epsilon\gamma + \delta)z + \epsilon\beta u_{\underline{x}} + \epsilon u_{\underline{w}} + u_{\underline{y}}\ . \qquad (23.8) \end{aligned}$$

Therefore

$$\left(\frac{\partial y}{\partial z}\right)_{u.-u_{\underline{z}}} = \epsilon\gamma + \delta\ , \tag{23.9}$$

where the partial derivative holds fixed all exogenous variables except $u_{\underline{z}}$. Note that this partial derivative is a sum of terms, and that each of those terms represents a different directed path from $\underline{z}$ to $\underline{y}(\underline{z})$. This is a general property of LDEN diagrams.

Fully Connected LDEN diagrams

The bnets that will be considered in this section will all be fully connected. Fully connected bnets are defined in Chapter 0.2. This section uses the notation $\langle \underline{x}, \underline{y} \rangle$ for the covariance of any two random variables $\underline{x}, \underline{y}$. This $\langle \underline{x}, \underline{y} \rangle$ notation is defined in the Notational Conventions Chapter 0.3.

Consider a LDEN diagram with deterministic nodes $\underline{x}. = (\underline{x}_k)_{k=0,1,\ldots nx-1}$ and corresponding exogenous nodes $\underline{u}. = (\underline{u}_k)_{k=0,1,\ldots nx-1}$. Assume $\langle \underline{u}_i, \underline{u}_j \rangle = 0$ if $i \neq j$. The strength of each connection $\underline{x}_i \to \underline{x}_j$ of the LDEN diagram is measured by a **structural coefficient** $\alpha_{j|i} \in \mathbb{R}$. Some of the $\alpha_{j|i}$ may be zero, in which case the corresponding arrow $i \to j$ would not be drawn.

Fully connected LDEN diagram with $nx = 2$

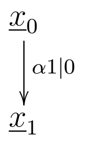


Figure 23.2: Fully connected LDEN diagram with two $\underline{x}_j$ nodes (exogenous nodes $\underline{u}_j$ not shown).

Consider the LDEN diagram of Fig.23.2. This diagram represents the following **structural equations**:

$$\underline{x}_0 = \underline{u}_0 \tag{23.10a}$$
$$\underline{x}_1 = \alpha_{1|0}\underline{x}_0 + \underline{u}_1 \; . \tag{23.10b}$$

Eqs.23.10 constitute a system of 2 linear equations in 2 unknowns (the $\underline{x}$'s) so we can solve for the $\underline{x}$'s in terms of the α's and $\underline{u}$'s.

Note also that

$$\langle \underline{x}_1, \underline{x}_0 \rangle = \alpha_{1|0} \langle \underline{x}_0, \underline{x}_0 \rangle \; . \tag{23.11}$$

Thus, $\alpha_{1|0}$ can be estimated from the covariances $\langle \underline{x}_i, \underline{x}_j \rangle$.

Fully connected LDEN diagram with $nx = 3$

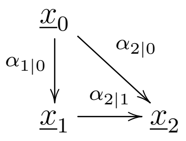


Figure 23.3: Fully connected LDEN diagram with three $\underline{x}_j$ nodes (exogenous nodes $\underline{u}_j$ not shown).

Consider the LDEN diagram of Fig.23.3. This diagram represents the following **structural equations**:

$$\underline{x}_0 = \underline{u}_0 \tag{23.12a}$$
$$\underline{x}_1 = \alpha_{1|0}\underline{x}_0 + \underline{u}_1 \tag{23.12b}$$
$$\underline{x}_2 = \alpha_{2|0}\underline{x}_0 + \alpha_{2|1}\underline{x}_1 + \underline{u}_2 \; . \tag{23.12c}$$

Eqs.23.12 constitute a system of 3 linear equations in 3 unknowns (the $\underline{x}$'s) so we can solve for the $\underline{x}$'s in terms of the α's and $\underline{u}$'s.

Note also that

$$\langle \underline{x}_1, \underline{x}_0 \rangle = \alpha_{1|0} \langle \underline{x}_0, \underline{x}_0 \rangle \tag{23.13a}$$
$$\langle \underline{x}_2, \underline{x}_0 \rangle = \alpha_{2|0} \langle \underline{x}_0, \underline{x}_0 \rangle + \alpha_{2|1} \langle \underline{x}_1, \underline{x}_0 \rangle \tag{23.13b}$$
$$\langle \underline{x}_2, \underline{x}_1 \rangle = \alpha_{2|0} \langle \underline{x}_0, \underline{x}_1 \rangle + \alpha_{2|1} \langle \underline{x}_1, \underline{x}_1 \rangle \tag{23.13c}$$

Eqs.23.13 constitute a system of 3 linear equations in 3 unknowns (the α's) so we can solve solve for the α's in terms of covariances $\langle \underline{x}_i, \underline{x}_j \rangle$. This gives an estimate for the α's.

Fully connected LDEN diagram with arbitrary nx.

Let $\underline{x}. = (x_i)_{i=0,1,\ldots,nx-1}$ and $\underline{x}_{<i} = (x_k)_{k=0,1,\ldots,i-1}$. Consider a fully connected LDEN diagram with deterministic nodes labeled $\underline{x}_i$. The $\underline{x}_i$ labels are assumed to be in **topological order** (i.e., the parents of node $\underline{x}_i$ are $\underline{x}_{<i}$). Let the TPMs, printed in blue, for the nodes $\underline{x}.$ of the LDEN diagram, be

$$P(x_i|x_{<i}, u_i) = \mathbb{1}(x_i = \sum_{k<i} \alpha_{i|k} x_k + u_i) \;, \tag{23.14}$$

for some parameters $\alpha_{i|k} \in \mathbb{R}$. The exogenous nodes $\underline{u}.$ are assumed to be independent so

$$P(u.) = \prod_i P(u_i) \tag{23.15}$$

and

$$\langle \underline{u}_i, \underline{u}_j \rangle = 0 \text{ if } i \neq j \;. \tag{23.16}$$

Note that

$$\begin{aligned} P(x.) &= \sum_{u.} P(u.) \prod_i P(x_i|x_{<i}, u_i) && (23.17) \\ &= E_{\underline{u}.}[\prod_i P(x_i|x_{<i}, u_i)] \;. && (23.18) \end{aligned}$$

In terms of random variables, this system is described by the following **structural equations**:

$$\underline{x}_i = \sum_{k<i} \alpha_{i|k} \underline{x}_k + \underline{u}_i \;. \tag{23.19}$$

The structural equations can be written in matrix form as follows. Define a strictly lower triangular matrix A with the connection strengths $\alpha_{i|k} \in \mathbb{R}$ as entries. For example, for $nx = 4$,

$$A = \left[\begin{array}{cccc} 0 & 0 & 0 & 0 \\ \alpha_{1|0} & 0 & 0 & 0 \\ \alpha_{2|0} & \alpha_{2|1} & 0 & 0 \\ \alpha_{3|0} & \alpha_{3|1} & \alpha_{3|2} & 0 \end{array}\right] \;. \tag{23.20}$$

If we now represent the multinodes $\underline{x}.$ and $\underline{u}.$ as column vectors $\underline{x}$ and $\underline{u}$, we get

$$\underline{x} = A\underline{x} + \underline{u} \;. \tag{23.21}$$

Note that

$$\underline{x} = (1 - A)^{-1} \underline{u} \;. \tag{23.22}$$

Therefore,

$$\underline{x}_i = f_i(\underline{u}_{\leq i}) \; . \tag{23.23}$$

Therefore, if $i > j$,

$$\langle \underline{u}_i, \underline{x}_j \rangle = \langle \underline{u}_i, f_j(\underline{u}_{\leq j}) \rangle = 0 \; . \tag{23.24}$$

Thus, if $i > j$,

$$\begin{aligned} \langle \underline{x}_i, \underline{x}_j \rangle &= \sum_{k<i} \alpha_{i|k} \langle \underline{x}_k, \underline{x}_j \rangle + \langle \underline{u}_i, \underline{x}_j \rangle & (23.25) \\ &= \sum_{k<i} \alpha_{i|k} \langle \underline{x}_k, \underline{x}_j \rangle \; . & (23.26) \end{aligned}$$

In matrix notation, Eq.(23.26) becomes

$$\langle \underline{x}, \underline{x}^T \rangle_L = A[\langle \underline{x}, \underline{x}^T \rangle_L + \langle \underline{x}, \underline{x}^T \rangle_D] \tag{23.27}$$

where we are using $\langle \underline{x}, \underline{x}^T \rangle_{i,j} = \langle \underline{x}_i, \underline{x}_j \rangle$ and denoting the strictly lower triangular part and diagonal part of a matrix M by M_L and M_D. Thus,

$$A = \langle \underline{x}, \underline{x}^T \rangle_L [\langle \underline{x}, \underline{x}^T \rangle_L + \langle \underline{x}, \underline{x}^T \rangle_D]^{-1} \; . \tag{23.28}$$

This gives an estimate for the α's in terms of the covariances $\langle \underline{x}_i, \underline{x}_j \rangle$.

Non-linear DEN diagrams

This chapter is dedicated to linear DEN diagrams. This implicitly assumes that the deterministic nodes $\underline{x}_.$ of the DEN diagram have an interval of real values as their possible states. A trivial but very useful generalization of linear DEN diagrams is to replace Eq.(23.14) for the TPMs of the deterministic nodes of the diagram by

$$P(x_i | x_{<i}, u_i) = \mathbb{1}(x_i = f_i(x_{<i}, u_i)) \; , \tag{23.29}$$

with structural equations

$$\underline{x}_i = f_i(\underline{x}_{<i}, \underline{u}_i) \; , \tag{23.30}$$

for $i = 0, 1, \ldots, nx - 1$. Here the f_i are possibly non-linear functions that depend the states $x_{<i}$ and u_i of nodes $\underline{x}_{<i}$ and $\underline{u}_i$. If a node $\underline{x}_i$ has no arrows entering it (i.e., is a root node), then

$$P(x_i | x_{<i}, u_i) = P(x_i) = \delta(x_i, a) \tag{23.31}$$

and

$$\underline{x}_i = a \tag{23.32}$$

for some $a \in S_{\underline{x}_i}$.

With this generalization, we can make any $f_i()$ represent a continuous probability distribution such as a Gaussian, or a discrete-valued Boolean function such as an OR gate.

Eqs.(23.29) and (23.30) are the TPMs and structural equations for a fully connected, non-linear DEN diagram. For a non-fully connected diagram,

- replace the multinode $x_{<i}$ by a subset of itself, in Eqs.(23.29) and (23.30) , and
- delete the corresponding arrows from the graph.

Bibliography

[1] Dan Bendel. Metropolis-Hastings: A comprehensive overview and proof. https://similarweb.engineering/mcmc/.

[2] Alexandra M Carvalho. Scoring functions for learning Bayesian networks. http://www.lx.it.pt/~asmc/pub/talks/09-TA/ta_pres.pdf.

[3] Charles Fox, Neil Girdhar, and Kevin Gurney. A causal Bayesian network view of reinforcement learning. https://www.aaai.org/Papers/FLAIRS/2008/FLAIRS08-030.pdf".

[4] Bruno Gonçalves. Model testing and causal search. blog post https://medium.com/data-for-science/causal-inference-part-vii-model-testing-and-causal-search-536b796f

[5] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, David Warde-Farley Bing Xu, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. https://arxiv.org/abs/1406.2661.

[6] Christina Heinze-Deml. Causality, spring semester 2019 at ETH Zurich. https://stat.ethz.ch/lectures/ss19/causality.php#course_materials.

[7] Cecil Huang and Adnan Darwiche. Inference in belief networks: A procedural guide. *International journal of approximate reasoning*, 15(3):225–263, 1996. http://www.ar-tiste.com/Huang-Darwiche1996.pdf.

[8] Steffen L Lauritzen and David J Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society: Series B (Methodological)*, 50(2):157–194, 1988. http://www.eecis.udel.edu/~shatkay/Course/papers/Lauritzen1988.pdf.

[9] Sergey Levine. Course CS 285 at UC Berkeley, Deep reinforcement learning. http://rail.eecs.berkeley.edu/deeprlcourse/.

[10] Dimitris Margaritis. Learning Bayesian network model structure from data (thesis, 2003, Carnegie Mellon Univ). https://apps.dtic.mil/sti/citations/ADA461103.

[11] Adam A Margolin, Ilya Nemenman, Katia Basso, Chris Wiggins, Gustavo Stolovitzky, Riccardo Dalla Favera, and Andrea Califano. Aracne: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. In *BMC bioinformatics*, volume 7, page S7. Springer, 2006. https://link.springer.com/article/10.1186/1471-2105-7-S1-S7.

[12] Robert J. McEliece, David J. C. MacKay, and Jung-Fu Cheng. Turbo decoding as an instance of Pearls belief propagation algorithm. `http://authors.library.caltech.edu/6938/1/MCEieeejstc98.pdf`.

[13] Richard E Neapolitan. *Learning Bayesian networks*. Pearson Prentice Hall, 2004.

[14] Andrew Ng. Lecture at deeplearning.ai on recurrent neural networks. `http://www.ar-tiste.com/ng-lec-rnn.pdf`.

[15] Gregory Nuel. Tutorial on exact belief propagation in Bayesian networks: from messages to algorithms. `https://arxiv.org/abs/1201.4724`.

[16] Judea Pearl. Mediating instrumental variables. `https://ftp.cs.ucla.edu/pub/stat_ser/r210.pdf`.

[17] Judea Pearl. Reverend Bayes on inference engines: A distributed hierarchical approach. `https://www.aaai.org/Papers/AAAI/1982/AAAI82-032.pdf`, 1982.

[18] Judea Pearl. *Probabilistic Inference in Intelligent Systems*. Morgan Kaufmann, 1988.

[19] Judea Pearl. *Causality: Models, Reasoning, and Inference, Second Edition*. Cambridge University Press, 2013.

[20] Judea Pearl. Causal and counterfactual inference. *The Handbook of Rationality*, pages 1–41, 2019. `https://ftp.cs.ucla.edu/pub/stat_ser/r485.pdf`.

[21] Judea Pearl, Madelyn Glymour, and Nicholas P Jewell. *Causal inference in statistics: A primer*. John Wiley & Sons, 2016.

[22] Judea Pearl and Dana Mackenzie. *The book of why: the new science of cause and effect*. Basic Books, 2018.

[23] ReliaSoft. System analysis reference. `http://reliawiki.org/index.php/System_Analysis_Reference`.

[24] Marco Scutari. bnlearn. `https://www.bnlearn.com/`.

[25] Marco Scutari, Catharina Elisabeth Graafland, and José Manuel Gutiérrez. Who learns better Bayesian network structures: Accuracy and speed of structure learning algorithms. *International Journal of Approximate Reasoning*, 115:235–253, 2019. `https://arxiv.org/abs/1805.11908`.

[26] Nitish Srivastava, G E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. `http://www.cs.toronto.edu/~rsalakhu/papers/srivastava14a.pdf`.

[27] Masayoshi Takahashi. Statistical inference in missing data by MCMC and non-MCMC multiple imputation algorithms: Assessing the effects of between-imputation iterations. *Data Science Journal*, 16, 2017. https://datascience.codata.org/articles/10.5334/dsj-2017-037/.

[28] theinvestorsbook.com. Pert analysis. https://theinvestorsbook.com/pert-analysis.html.

[29] Robert R. Tucci. Bell's inequalities for Bayesian statisticians. blog post in blog Quantum Bayesian Networks, https://qbnets.wordpress.com/2008/09/19/bells-inequaties-for-bayesian-statistician/.

[30] Robert R. Tucci. Quantum Fog. https://github.com/artiste-qb-net/quantum-fog.

[31] W. E. Vesely, F. F. Goldberg, N. H. Roberts, and D. F. Haasl. Fault tree handbook nureg-0492. https://www.nrc.gov/reading-rm/doc-collections/nuregs/staff/sr0492/.

[32] Wikipedia. Belief propagation. https://en.wikipedia.org/wiki/Belief_propagation.

[33] Wikipedia. Beta function. https://en.wikipedia.org/wiki/Beta_function.

[34] Wikipedia. Binary decision diagram. https://en.wikipedia.org/wiki/Binary_decision_diagram.

[35] Wikipedia. Boolean algebra. https://en.wikipedia.org/wiki/Boolean_algebra.

[36] Wikipedia. Categorical distribution. https://en.wikipedia.org/wiki/Categorical_distribution.

[37] Wikipedia. Chow-Liu tree. https://en.wikipedia.org/wiki/Chow%E2%80%93Liu_tree.

[38] Wikipedia. Data processing inequality. https://en.wikipedia.org/wiki/Data_processing_inequality.

[39] Wikipedia. Dirichlet distribution. https://en.wikipedia.org/wiki/Dirichlet_distribution.

[40] Wikipedia. Expectation maximization. https://en.wikipedia.org/wiki/Expectation%E2%80%93maximization_algorithm.

[41] Wikipedia. Gamma function. https://en.wikipedia.org/wiki/Gamma_function.

[42] Wikipedia. Gated recurrent unit. https://en.wikipedia.org/wiki/Gated_recurrent_unit.

[43] Wikipedia. Gibbs sampling. https://en.wikipedia.org/wiki/Gibbs_sampling.

[44] Wikipedia. Hidden Markov model. https://en.wikipedia.org/wiki/Hidden_Markov_model.

[45] Wikipedia. Importance sampling. https://en.wikipedia.org/wiki/Importance_sampling.

[46] Wikipedia. Inverse transform sampling. https://en.wikipedia.org/wiki/Inverse_transform_sampling.

[47] Wikipedia. Junction tree algorithm. https://en.wikipedia.org/wiki/Junction_tree_algorithm.

[48] Wikipedia. k-means clustering. https://en.wikipedia.org/wiki/K-means_clustering.

[49] Wikipedia. Kalman filter. https://en.wikipedia.org/wiki/Kalman_filter.

[50] Wikipedia. Long short term memory. https://en.wikipedia.org/wiki/Long_short-term_memory.

[51] Wikipedia. Markov blanket. https://en.wikipedia.org/wiki/Markov_blanket.

[52] Wikipedia. Metropolis-Hastings method. https://en.wikipedia.org/wiki/Metropolis%E2%80%93Hastings_algorithm.

[53] Wikipedia. Minimum spanning tree. https://en.wikipedia.org/wiki/Minimum_spanning_tree.

[54] Wikipedia. Monte Carlo methods. https://en.wikipedia.org/wiki/Category:Monte_Carlo_methods.

[55] Wikipedia. Multinomial distribution. https://en.wikipedia.org/wiki/Multinomial_distribution.

[56] Wikipedia. Multinomial theorem. https://en.wikipedia.org/wiki/Multinomial_theorem.

[57] Wikipedia. Multivariate normal distribution. https://en.wikipedia.org/wiki/Multivariate_normal_distribution.

[58] Wikipedia. Non-negative matrix factorization. https://en.wikipedia.org/wiki/Non-negative_matrix_factorization.

[59] Wikipedia. Program evaluation and review technique. https://en.wikipedia.org/wiki/Program_evaluation_and_review_technique.

[60] Wikipedia. Rejection sampling. https://en.wikipedia.org/wiki/Rejection_sampling.

[61] Wikipedia. Simpson's paradox. https://en.wikipedia.org/wiki/Simpson's_paradox.

[62] Wikipedia. Spring system. https://en.wikipedia.org/wiki/Spring_system.

[63] Wikipedia. Variational Bayesian methods. https://en.wikipedia.org/wiki/Variational_Bayesian_methods.

[64] Hao Wu and Zhaohui Steve Qin. course notes, BIOS731: Advanced statistical computing, 2016 Emory Univ. http://web1.sph.emory.edu/users/hwu30/teaching/statcomp/statcomp.html.