

Bayesuvius,
a small visual dictionary of Bayesian Networks

Robert R. Tucci
www.ar-tiste.xyz

June 25, 2020



Figure 1: View of Mount Vesuvius from Pompeii

Contents

| | | |
|-----|--------------------------------------|----|
| 0.1 | Foreword | 3 |
| 0.2 | Notational Conventions | 4 |
| 1 | Generative Adversarial Network (GAN) | 6 |
| 2 | Linear and Logistic Regression | 11 |
| | Bibliography | 15 |

0.1 Foreword

Welcome to Bayesuvius! a proto-book uploaded to github.

A different Bayesian network is discussed in each chapter. Each chapter title is the name of a B net. Chapter titles are in alphabetical order.

This is a volcano in its early stages. First version uploaded to a github repo called Bayesuvius on June 24, 2020. First version only covers 2 B nets (Linear Regression and GAN). I will add more chapters periodically. Remember, this is a moonlighting effort so I can't do it all at once.

For any questions about notation, please go to Notational Conventions section.

Requests and advice are welcomed.

Thanks for reading this.

Robert R. Tucci

www.ar-tiste.xyz

0.2 Notational Conventions

bnet=B net=Bayesian Network

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 .

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

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

$$P_{\underline{x}, \underline{y}}(x, y) = P(\underline{x} = x, \underline{y} = y) = P(x, y) \quad (2)$$

$$P_{\underline{x}|\underline{y}}(x|y) = P(\underline{x} = x | \underline{y} = y) = P(x|y) = \frac{P(x, y)}{P(y)} \quad (3)$$

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

$$\delta(x, y) = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{if } x \neq y \end{cases} \quad (4)$$

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

$$\int_{-\infty}^{+\infty} dx \delta(x - y) f(x) = f(y) \quad (5)$$

Indicator function:

$$\hat{1}(\mathcal{S}) = \begin{cases} 1 & \text{if } \mathcal{S} \text{ is true} \\ 0 & \text{if } \mathcal{S} \text{ is false} \end{cases} \quad (6)$$

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

$$\vec{x} = (x[0], x[1], x[2] \dots, x[nsam(\vec{x}) - 1]) = x[:] \quad (7)$$

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

$$x[i] \sim P_{\underline{x}} \text{ (i.e. } P_{\underline{x}[i]} = P_{\underline{x}}) \quad (8)$$

$$P(\underline{x} = x) = \frac{1}{nsam(\vec{x})} \sum_i \hat{1}(x[i] = x) \quad (9)$$

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.

$$P(\vec{x}) = \prod_i P(x[i]) \quad (10)$$

$$\sum_{\vec{x}} = \prod_i \sum_{x[i]} \quad (11)$$

$$\partial_{\vec{x}} = [\partial_{x[0]}, \partial_{x[1]}, \partial_{x[2]}, \dots, \partial_{x[nsam(\vec{x})-1]}] \quad (12)$$

$$P(\vec{x}) \approx \left[\prod_x P(x)^{P(x)} \right]^{nsam(\vec{x})} \quad (13)$$

$$= e^{nsam(\vec{x}) \sum_x P(x) \log P(x)} \quad (14)$$

$$= e^{-nsam(\vec{x}) H(P_{\underline{x}})} \quad (15)$$

$$f^{[1, \partial_x, \partial_y]}(x, y) = [f, \partial_x f, \partial_y f] \quad (16)$$

$$f^+ = f^{[1, \partial_x, \partial_y]} \quad (17)$$

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

- Entropy:

$$H(p) = - \sum_x p(x) \log p(x) \geq 0 \quad (18)$$

- Kullback-Liebler divergence:

$$D_{KL}(p \parallel q) = \sum_x p(x) \log \frac{p(x)}{q(x)} \geq 0 \quad (19)$$

- Cross entropy:

$$CE(p \rightarrow q) = - \sum_x p(x) \log q(x) \quad (20)$$

$$= H(p) + D_{KL}(p \parallel q) \quad (21)$$

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

$$\mathcal{N}(\mu, \sigma)(x) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{x-\mu}{\sigma} \right)^2} \quad (22)$$

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

$$\mathcal{U}(a, b)(x) = \frac{1}{b-a} \quad (23)$$

Chapter 1

Generative Adversarial Network (GAN)

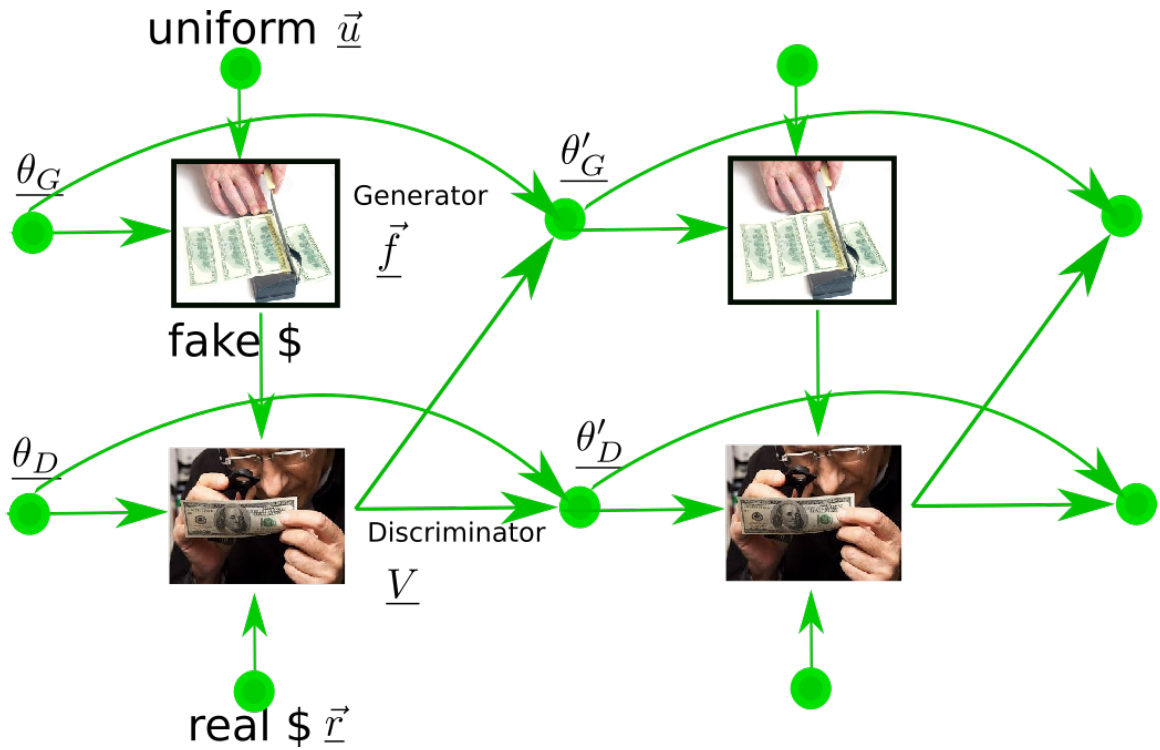


Figure 1.1: Generative Adversarial Network (GAN)

Original GAN, Ref.[1](2014).

Generator G (counterfeiter) generates samples \vec{f} of fake money and submits them to Discriminator D (Treasury agent). D also gets samples \vec{r} of real money. D submits verdict $V \in [0, 1]$. G depends on parameter θ_G and D on parameter θ_D . Verdict V and initial θ_G, θ_D are used to get new parameters θ'_G, θ'_D . Process is repeated (Dynamical Bayesian Network) until saddle point in $V(\theta_G, \theta_D)$ is reached. D makes G better and vice versa. Zero-sum game between D and G .



Figure 1.2: Discriminator node \underline{V} in Fig.1.1 can be split into 3 nodes \underline{c} , \underline{d} and \underline{V} .

Let \mathcal{D} be the domain of $D(\cdot, \theta_D)$. Assume that for any $x \in \mathcal{D}$,

$$0 \leq D(x, \theta_D) \leq 1 . \quad (1.1)$$

For any $S \subset \mathcal{D}$, define

$$\sum_{x \in S} D(x, \theta_D) = \lambda(S, \theta_D) . \quad (1.2)$$

In general, $G(\cdot, \theta_G)$ need not be real valued.

Assume that for every $u \in S_u$, $G(u, \theta_G) = f \in S_f \subset \mathcal{D}$. Define

$$\overline{D}(f, \theta_D) = 1 - D(f, \theta_D) . \quad (1.3)$$

Note that

$$0 \leq \overline{D}(f, \theta_D) \leq 1 . \quad (1.4)$$

Define:

$$V(\theta_G, \theta_D) = \sum_r P(r) \log D(r, \theta_D) + \sum_u P(u) \log \overline{D}(G(u, \theta_G), \theta_D) . \quad (1.5)$$

We want the first variation of $V(\theta_G, \theta_D)$ to vanish.

$$\delta V(\theta_G, \theta_D) = 0 . \quad (1.6)$$

This implies

$$\partial_{\theta_G} V(\theta_G, \theta_D) = \partial_{\theta_D} V(\theta_G, \theta_D) = 0 \quad (1.7)$$

and

$$V_{opt} = \min_{\theta_G} \max_{\theta_D} V(\theta_G, \theta_D) . \quad (1.8)$$

Node transition probability matrices for Figs. 1.1 and 1.2 are given next in blue:

$$P(\theta_G) = \text{given} \quad (1.9)$$

$$P(\theta_D) = \text{given} \quad (1.10)$$

$$P(\vec{u}) = \prod_i P(u[i]) \quad (\text{usually uniform distribution}) \quad (1.11)$$

$$P(\vec{r}) = \prod_i P(r[i]) \quad (1.12)$$

$$P(f[i]|\vec{u}, \theta_G) = \prod_i \delta[f[i], G(u[i], \theta_G)] \quad (1.13)$$

$$P(c[i]|\vec{f}, \theta_D) = \delta(c[i], \overline{D}(f[i], \theta_D)) \quad (1.14)$$

$$P(d[j]|\vec{r}, \theta_D) = \delta(d[j], D(r[j], \theta_D)) \quad (1.15)$$

$$P(V|\vec{d}, \vec{c}) = \delta(V, \frac{1}{N} \log \prod_{i,j} (c[i]d[j])) \quad (1.16)$$

where $N = nsam(\vec{r})nsam(\vec{u})$.

Let $\eta_G, \eta_D > 0$. Maximize V wrt θ_D , and minimize it wrt θ_G .

$$P(\theta'_G|V, \theta_G) = \delta(\theta'_G, \theta_G - \eta_G \partial_{\theta_G} V) \quad (1.17)$$

$$P(\theta'_D|V, \theta_D) = \delta(\theta'_D, \theta_D + \eta_D \partial_{\theta_D} V) \quad (1.18)$$



Figure 1.3: GAN, Constraining Bayesian Network

Constraining B net given in Fig.1.3. It adds 2 new nodes, namely \vec{U} and \vec{R} , to the bnet of Fig.1.1. The purpose of these 2 barren (childrenless) nodes is to constrain certain functions to be probability distributions.

Node transition probabilities for the 2 new nodes given next in blue.

$$P(U[i]|\theta_G) = \frac{\overline{D}(G(U[i], \theta_G), \theta_D))}{\overline{\lambda}(\theta_G, \theta_D)} \quad (1.19)$$

where $S_{\underline{U}[i]} = S_{\underline{u}}$ and $\overline{\lambda}(\theta_G, \theta_D) = \sum_u \overline{D}(G(u, \theta_G), \theta_D)$.

$$P(R[i]|\theta_G, \theta_D) = \frac{D(R[i], \theta_D)}{\lambda(\theta_D)} \quad (1.20)$$

where $S_{\underline{R}[i]} = S_{\underline{r}}$ and $\lambda(\theta_D) = \sum_r D(r, \theta_D)$.

$$P(V|\vec{u}, \vec{r}) = \delta(V, \frac{1}{N} \log \prod_{i,j} (P(\underline{R}[i] = r[i]|\theta_G, \theta_D) P(\underline{U}[i] = u[j]|\theta_G))) \quad (1.21)$$

where $N = nsam(\vec{r})nsam(\vec{u})$.

\mathcal{L} = likelihood

$$\mathcal{L} = P(\vec{r}, \vec{u}|\theta_G, \theta_D) \quad (1.22)$$

$$= \prod_{i,j} \left[\frac{D(r[i], \theta_D)}{\lambda(\theta_D)} \frac{\overline{D}(G(u[j], \theta_G), \theta_D))}{\overline{\lambda}(\theta_G, \theta_D)} \right] \quad (1.23)$$

$$\log \mathcal{L} = N[V(\theta_G, \theta_D) - \log \lambda(\theta_D) - \log \bar{\lambda}(\theta_G, \theta_D)] \quad (1.24)$$

Chapter 2

Linear and Logistic Regression

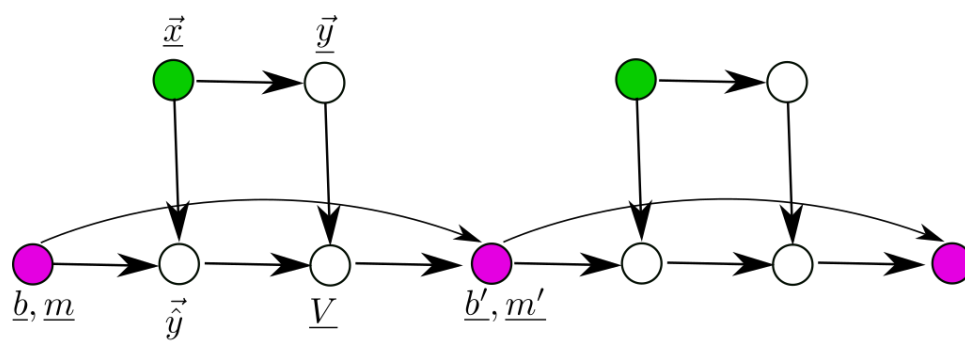


Figure 2.1: Linear Regression

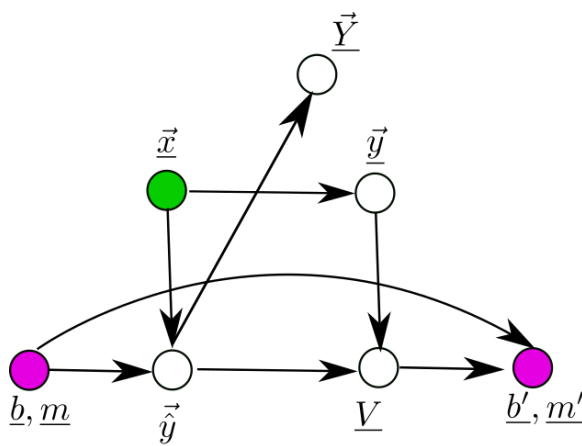


Figure 2.2: B net of Fig. 2.1 with new \vec{Y} node.

Estimators \hat{y} for linear and logistic regression.

- **Linear Regression:** $y \in \mathbb{R}$. Note $\hat{y} \in \mathbb{R}$. $(x, \hat{y}(x))$ is a straight line with y-intercept b and slope m .

$$\hat{y}(x; b, m) = b + mx \quad (2.1)$$

- **Logistic Regression:** $y \in \{0, 1\}$. Note $\hat{y} \in [0, 1]$. $(x, \hat{y}(x))$ is a sigmoid. Often in literature, b, m are replaced by β_0, β_1 .

$$\hat{y}(x; b, m) = \frac{1}{1 + e^{-(b+mx)}} \quad (2.2)$$

Define

$$V(b, m) = \sum_{x,y} P(x, y) |y - \hat{y}(x; b, m)|^2 . \quad (2.3)$$

We want to minimize $V(b, m)$ (called a cost or loss function) wrt b and m .

Node transition probabilities of B net of Fig.2.1 given next in blue.

$$P(b, m) = \text{given} \quad (2.4)$$

$$P(\vec{x}) = \prod_i P(x[i]) \quad (2.5)$$

$$P(\vec{y}|\vec{x}) = \prod_i P(y[i]|x[i]) \quad (2.6)$$

$$P(\vec{\hat{y}}|\vec{x}, b, m) = \prod_i \delta(\hat{y}[i], \hat{y}(x[i], b, m)) \quad (2.7)$$

$$P(V|\vec{\hat{y}}, \vec{y}) = \delta(V, \frac{1}{nsam(\vec{x})} \log \prod_i |\hat{y}[i] - y[i]|^2) \quad (2.8)$$

Let $\eta_b, \eta_m > 0$. For $x = b, m$, if $x' - x = \Delta x = -\eta \frac{\partial V}{\partial x}$, then $\Delta V \approx \frac{-1}{\eta} (\Delta x)^2 \leq 0$ for $\eta > 0$. This is called “gradient descent”.

$$P(b'|V, b) = \delta(b', b - \eta_b \partial_b V) \quad (2.9)$$

$$P(m'|V, m) = \delta(m', m - \eta_m \partial_m V) \quad (2.10)$$

Generalization to x with multiple components(features)

Suppose that for each sample i , instead of $x[i]$ being a scalar, it has n components called features:

$$x[i] = (x_0[i], x_1[i], x_2[i], \dots x_{n-1}[i]) . \quad (2.11)$$

Slope m is replaced by weights

$$w = (w_0, w_1, w_3, \dots, w_{n-1}), \quad (2.12)$$

and the product of 2 scalars $mx[i]$ is replaced by the inner vector product $w^T x[i]$.

Alternative $V(b, m)$ for logistic regression

For logistic regression, since $y[i] \in \{0, 1\}$ and $\hat{y}[i] \in [0, 1]$ are both in the interval $[0, 1]$, they can be interpreted as probabilities. Define probability distributions $p[i](x)$ and $\hat{p}[i](x)$ for $x \in \{0, 1\}$ by

$$p[i](1) = y[i], \quad p[i](0) = 1 - y[i] \quad (2.13)$$

$$\hat{p}[i](1) = \hat{y}[i], \quad \hat{p}[i](0) = 1 - \hat{y}[i] \quad (2.14)$$

Then for logistic regression, the following 2 cost functions $V(b, m)$ can be used as alternatives to the cost function Eq.(2.3) previously given.

$$V(b, m) = \frac{1}{nsam(\vec{x})} \sum_i D_{KL}(p[i] \parallel \hat{p}[i]) \quad (2.15)$$

and

$$V(b, m) = \frac{1}{nsam(\vec{x})} \sum_i CE(p[i] \rightarrow \hat{p}[i]) \quad (2.16)$$

$$= \frac{-1}{nsam(\vec{x})} \sum_i \{y[i] \log \hat{y}[i] + (1 - y[i]) \log(1 - \hat{y}[i])\} \quad (2.17)$$

$$= \frac{-1}{nsam(\vec{x})} \sum_i \log \{ \hat{y}[i]^{y[i]} (1 - \hat{y}[i])^{(1-y[i])} \} \quad (2.18)$$

$$= \frac{-1}{nsam(\vec{x})} \sum_i \log P(\underline{Y} = y[i] | \hat{y} = \hat{y}[i]) \quad (2.19)$$

$$= - \sum_{x,y} P(x, y) \log P(\underline{Y} = y | \hat{y} = \hat{y}(x, b, m)) \quad (2.20)$$

Above, we used

$$P(\underline{Y} = Y | \hat{y}) = \hat{y}^Y [1 - \hat{y}]^{1-Y} \quad (2.21)$$

for $Y \in S_Y = \{0, 1\}$. (Bernoulli distribution).

There is no node corresponding to \underline{Y} in the B net of Fig.2.1. Fig.2.2 shows a new B net that has a new node called $\vec{\underline{Y}}$ compared to the B net of Fig.2.1. One defines the transition probabilities for all nodes of Fig.2.2 except $\vec{\underline{Y}}$ and \underline{V} the same as for Fig.2.1. For $\vec{\underline{Y}}$ and \underline{V} , one defines

$$P(Y[i] | \vec{\hat{y}}) = P(\underline{Y} = Y[i] | \hat{y}[i]) \quad (2.22)$$

$$P(V|\vec{Y}, \vec{y}) = \delta(V, \frac{-1}{nsam(\vec{x})} \log \mathcal{L}) , \quad (2.23)$$

where $\mathcal{L} = \prod_i P(Y = y[i]|\hat{y}[i])$ =likelihood.

Bibliography

- [1] Ian J. Goodfellow et al. *Generative Adversarial Networks*. <https://arxiv.org/abs/1406.2661>.