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

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,

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

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

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

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

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

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

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

Transition probability matrix 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 node by sums over new states, and Dirac delta functions by Kronecker delta functions. More precisely, consider a function $f : S \to \mathbb{R}$. Let $S_{\underline{x}} \subset S$ and $S \to S_{\underline{x}}$ upon discretization (binning). Then

$$\int_S dx\ P_{\underline{x}}(x) f(x) \to \frac{1}{n_{\underline{x}}} \sum_{x \in S_{\underline{x}}} f(x)\ . \tag{6}$$

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

$$\int_S dx\ \delta(x - y) f(x) = f(y) \to \sum_{x \in S_{\underline{x}}} \delta(x, y) f(x) = f(y)\ . \tag{7}$$

Indicator function:

$$\hat{1}(\mathcal{S}) = \begin{cases} 1 \text{ if } \mathcal{S} \text{ is true} \\ 0 \text{ if } \mathcal{S} \text{ is false} \end{cases} \tag{8}$$

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

$$\vec{x} = (x[0], x[1], x[2] \ldots, x[nsam(\vec{x}) - 1]) = x[:] \tag{9}$$

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

$$x[i] \sim P_{\underline{x}} \ \text{(i.e. } P_{\underline{x[i]}} = P_{\underline{x}}) \tag{10}$$

$$P(\underline{x} = x) = \frac{1}{nsam(\vec{x})} \sum_i \hat{1}(x[i] = x) \tag{11}$$

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.

$$P(\vec{x}) = \prod_i P(x[i]) \tag{12}$$

$$\sum_{\vec{x}} = \prod_i \sum_{x[i]} \tag{13}$$

$$\partial_{\vec{x}} = [\partial_{x[0]}, \partial_{x[1]}, \partial_{x[2]}, \ldots, \partial_{x[nsam(\vec{x})-1]}] \tag{14}$$

$$\begin{aligned} P(\vec{x}) &\approx [\prod_x P(x)^{P(x)}]^{nsam(\vec{x})} && (15)\\ &= e^{nsam(\vec{x}) \sum_x P(x) \log P(x)} && (16)\\ &= e^{-nsam(\vec{x}) H(P_{\underline{x}})} && (17) \end{aligned}$$

$$f^{[1,\partial_x,\partial_y]}(x, y) = [f, \partial_x f, \partial_y f] \tag{18}$$

$$f^+ = f^{[1,\partial_x,\partial_y]} \tag{19}$$

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

- Entropy:

$$H(p) = -\sum_x p(x) \log p(x) \geq 0 \tag{20}$$

- Kullback-Liebler divergence:

$$D_{KL}(p \parallel q) = \sum_x p(x) \log \frac{p(x)}{q(x)} \geq 0 \tag{21}$$

- Cross entropy:

$$\begin{aligned} CE(p \to q) &= -\sum_x p(x) \log q(x) && (22)\\ &= H(p) + D_{KL}(p \parallel q) && (23) \end{aligned}$$

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

$$\mathcal{N}(\mu, \sigma)(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \tag{24}$$

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

$$\mathcal{U}(a, b)(x) = \frac{1}{b-a} \tag{25}$$

Chapter 1

Generative Adversarial Network (GAN)

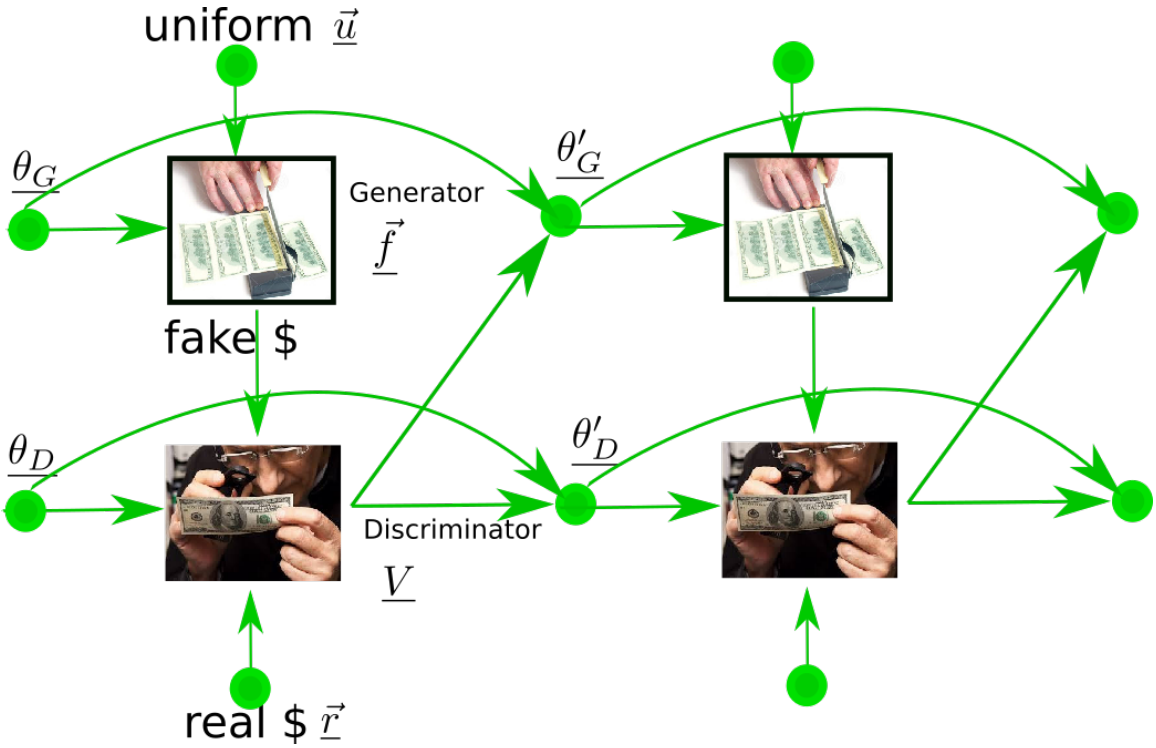

Figure 1.1: Generative Adversarial Network (GAN)

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 veredict $V \in [0, 1]$. G depends on parameter θ_G and D on parameter θ_D. Veredict 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.

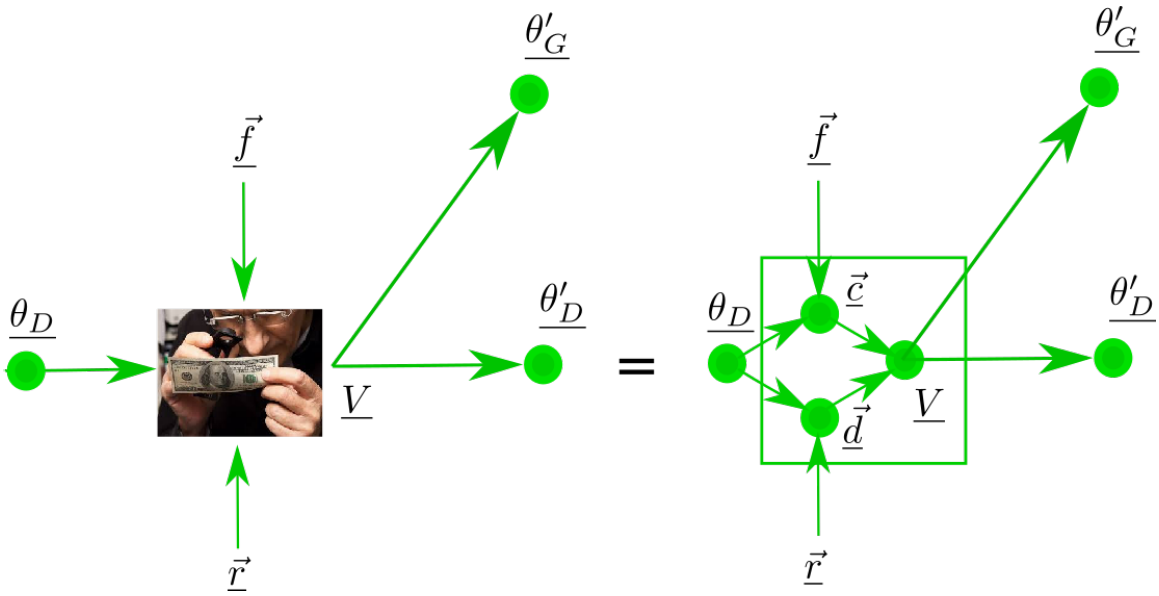

Figure 1.2: Discriminator node $\underline{V}$ in Fig.1.1 can be split into 3 nodes $\vec{\underline{c}}$, $\vec{\underline{d}}$ and $\underline{V}$.

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

$$0 \leq D(x, \theta_D) \leq 1 \ . \tag{1.1}$$

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

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

In general, $G(\cdot, \theta_G)$ need not be real valued.
Assume that for every $u \in S_{\underline{u}}$, $G(u, \theta_G) = f \in S_{\underline{f}} \subset \mathcal{D}$. Define

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

Note that

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

Define:

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

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

$$\delta V(\theta_G, \theta_D) = 0 \ . \tag{1.6}$$

This implies

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

and

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

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

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

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

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

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

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

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

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

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

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.

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

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

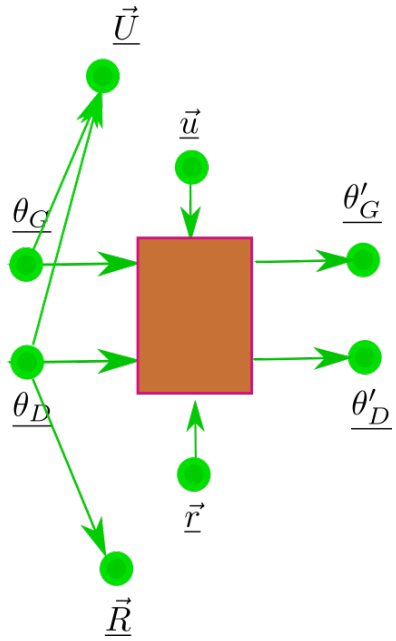

Figure 1.3: GAN, Constraining Bayesian Network

Constraining B net given in Fig.1.3. It adds 2 new nodes, namely $\underline{\vec{U}}$ and $\underline{\vec{R}}$, 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.

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

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

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

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

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

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

$\mathcal{L}$ = likelihood

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

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

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

Chapter 2

Linear and Logistic Regression

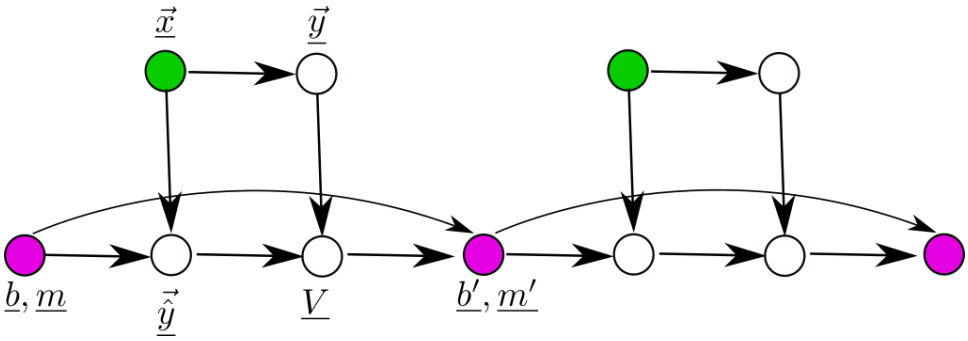

Figure 2.1: Linear Regression

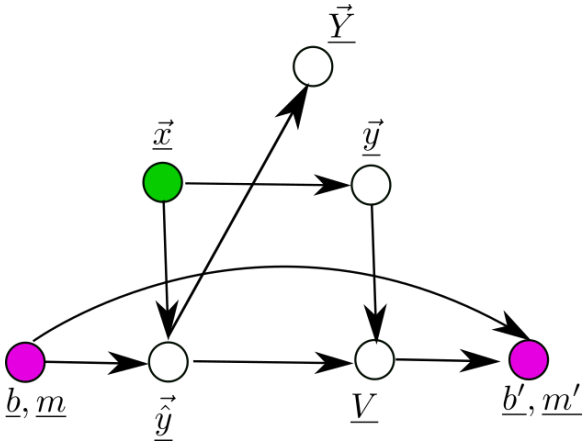

Figure 2.2: B net of Fig.2.1 with new $\vec{Y}$ node.

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.

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

- **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.

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

Define

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

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.

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

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

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

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

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

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".

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

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

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:

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

Slope m is replaced by weights

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

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

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

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

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.

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

and

$$\begin{aligned} V(b,m) &= \frac{1}{nsam(\vec{x})} \sum_i CE(p[i] \to \hat{p}[i]) && (2.16)\\ &= \frac{-1}{nsam(\vec{x})} \sum_i \{y[i] \log \hat{y}[i] + (1 - y[i]) \log(1 - \hat{y}[i])\} && (2.17)\\ &= \frac{-1}{nsam(\vec{x})} \sum_i \log \left\{\hat{y}[i]^{y[i]} (1 - \hat{y}[i])^{(1-y[i])}\right\} && (2.18)\\ &= \frac{-1}{nsam(\vec{x})} \sum_i \log P(\underline{Y} = y[i] | \hat{y} = \hat{y}[i]) && (2.19)\\ &= -\sum_{x,y} P(x,y) \log P(\underline{Y} = y | \hat{y} = \hat{y}(x,b,m)) && (2.20) \end{aligned}$$

Above, we used

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

for $Y \in S_{\underline{Y}} = \{0,1\}$. (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

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

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

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

Bibliography

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