Learning Bayesian Network tables

David Barber

University College London

Learning the bias of a coin

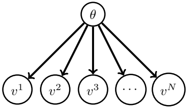
$$v^n = \begin{cases} 1 & \text{if on toss } n \text{ the coin comes up heads} \\ 0 & \text{if on toss } n \text{ the coin comes up tails} \end{cases}$$

Our aim is to estimate the probability θ that the coin will be a head, $p(v^n = 1|\theta) = \theta$ – called the 'bias' of the coin.

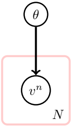
Building a model

The variables are $v^1, \ldots, v^N$ and θ and we require a model of the probabilistic interaction of the variables, $p(v^1, \ldots, v^N, \theta)$. Assuming there is no dependence between the observed tosses, except through θ, we have the belief network

$$p(v^1, \ldots, v^N, \theta) = p(\theta) \prod_{n=1}^{N} p(v^n|\theta)$$



(a)



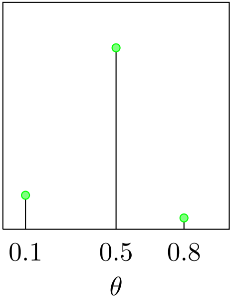
(b)

Figure: **(a)**: Belief network for coin tossing model. **(b)**: Plate notation equivalent of (a). A plate replicates the quantities inside the plate a number of times as specified in the plate.

The prior

We still need to fully specify the prior $p(\theta)$. To avoid complexities resulting from continuous variables, we'll consider a discrete θ with only three possible states, $\theta \in \{0.1, 0.5, 0.8\}$. Specifically, we assume

$$p(\theta = 0.1) = 0.15,\ p(\theta = 0.5) = 0.8,\ p(\theta = 0.8) = 0.05$$



The posterior

$$
\begin{aligned}
p(\theta|v^1,\dots,v^N) &\propto p(\theta)\prod_{n=1}^{N} p(v^n|\theta) \\
&= p(\theta)\prod_{n=1}^{N} \theta^{\mathbb{I}[v^n=1]}\left(1-\theta\right)^{\mathbb{I}[v^n=0]} \\
&\propto p(\theta)\theta^{\sum_{n=1}^{N}\mathbb{I}[v^n=1]}\left(1-\theta\right)^{\sum_{n=1}^{N}\mathbb{I}[v^n=0]}
\end{aligned}
$$

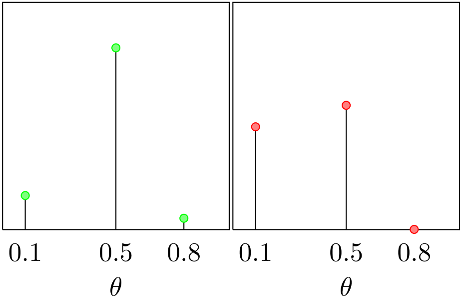
Hence

$$
p(\theta|v^1,\dots,v^N) \propto p(\theta)\theta^{N_H}\left(1-\theta\right)^{N_T}
$$

$N_H = \sum_{n=1}^{N}\mathbb{I}\left[v^n = 1\right]$ is the number of occurrences of heads.
$N_T = \sum_{n=1}^{N}\mathbb{I}\left[v^n = 0\right]$ is the number of tails.

Coin posterior

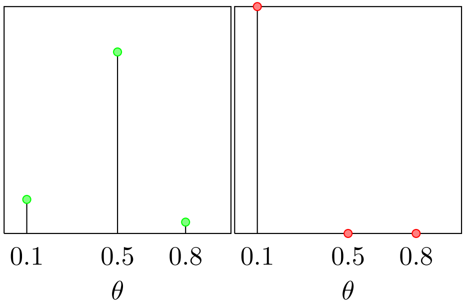
For an experiment with $N_H = 2$, $N_T = 8$, the posterior distribution is



If we were asked to choose a single *a posteriori* most likely value for θ, it would be $\theta = 0.5$, although our confidence in this is low since the posterior belief that $\theta = 0.1$ is also appreciable. This result is intuitive since, even though we observed more Tails than Heads, our prior belief was that it was more likely the coin is fair.

The coin posterior

Repeating the above with $N_H = 20$, $N_T = 80$, the posterior changes to



so that the posterior belief in $\theta = 0.1$ dominates. There are so many more tails than heads that this is unlikely to occur from a fair coin. Even though we *a priori* thought that the coin was fair, *a posteriori* we have enough evidence to change our minds.

The posterior effect

Note that in both examples, $N_T/N_H = 4$, although in the latter we are much more confident that $\theta = 0.1$

Continuous Parameters

We first examine the case of a 'flat' prior $p(\theta) = k$ for some constant k. For continuous variables, normalisation requires

$$\int_0^1 p(\theta) d\theta = k = 1$$

Repeating the previous calculations with this flat continuous prior, we have

$$p(\theta|\mathcal{V}) = \frac{1}{c}\theta^{N_H} (1-\theta)^{N_T}$$

where c is a constant to be determined by normalisation,

$$c = \int_0^1 \theta^{N_H} (1-\theta)^{N_T} d\theta \equiv B(N_H + 1, N_T + 1)$$

where $B(\alpha, \beta)$ is the Beta function.

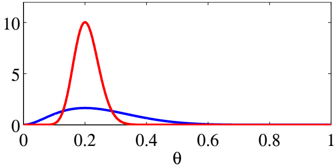


Figure: Posterior $p(\theta|\mathcal{V})$ assuming a flat prior on θ. (blue) $N_H = 2$, $N_T = 8$ and (red) $N_H = 20$, $N_T = 80$. The Maximum A Posteriori setting is $\theta = 0.2$ in both cases, this being the value of θ for which the posterior attains its highest value.

Using a conjugate prior

For the coin tossing case, it is clear that if the prior is of the form of a Beta distribution, then the posterior will be of the same parametric form:

$$p(\theta) = \frac{1}{B(\alpha,\beta)} \theta^{\alpha-1} (1-\theta)^{\beta-1}$$

the posterior is

$$p(\theta|\mathcal{V}) \propto \theta^{\alpha-1} (1-\theta)^{\beta-1} \theta^{N_H} (1-\theta)^{N_T}$$

so that

$$\begin{aligned} p(\theta|\mathcal{V}) &= \frac{1}{B(\alpha+N_H, \beta+N_T)} \theta^{\alpha+N_H-1} (1-\theta)^{\beta+N_T-1} \\ &\equiv B(\theta|\alpha+N_H, \beta+N_T) \end{aligned}$$

The prior and posterior are of the same form (both Beta distributions) but simply with different parameters. Hence the Beta distribution is ‘conjugate’ to the Binomial distribution.

Maximum Likelihood Training of Belief Networks

Consider the following model of the relationship between exposure to asbestos (a), being a smoker (s) and the incidence of lung cancer (c)

$$p(a, s, c) = p(c|a, s)p(a)p(s)$$

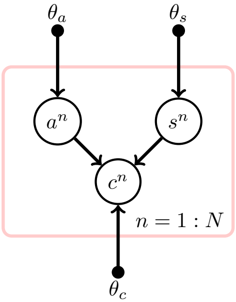
Each variable is binary, $\mathrm{dom}(a) = \{0, 1\}$, $\mathrm{dom}(s) = \{0, 1\}$, $\mathrm{dom}(c) = \{0, 1\}$. Furthermore, we assume that we have a list of patient records, where each row represents a patient's data.

a	s	c
1	1	1
1	0	0
0	1	1
0	1	0
1	1	1
0	0	0
1	0	1

A database containing information about the Asbestos exposure (1 signifies exposure), being a Smoker (1 signifies the individual is a smoker), and lung Cancer (1 signifies the individual has lung Cancer). Each row contains the information for an individual, so that there are 7 individuals in the database.

Learning the table

a	s	c
1	1	1
1	0	0
0	1	1
0	1	0
1	1	1
0	0	0
1	0	1



To learn the table entries $p(c|a, s)$ we can do so by counting the number of c is in state 1 for each of the 4 parental states of a and s:

$$
\begin{array}{ll}
p(c = 1|a = 0, s = 0) = 0, & p(c = 1|a = 0, s = 1) = 0.5 \\
p(c = 1|a = 1, s = 0) = 0.5 & p(c = 1|a = 1, s = 1) = 1
\end{array}
$$

Similarly, based on counting, $p(a = 1) = 4/7$, and $p(s = 1) = 4/7$. These three CPTs then complete the full distribution specification.

Maximum Likelihood and the KL divergence

$$\mathrm{KL}(q(x)|p(x|\theta)) = \left\langle \log \frac{q(x)}{p(x)} \right\rangle_{q(x)} \geq 0$$

Let q be the empirical distribution:

$$q(x) = \frac{1}{N} \sum_{n=1}^{N} \mathbb{I}\left[x = x^n\right]$$

Then

$$\begin{aligned}
\mathrm{KL}(q|p(x|\theta)) &= \langle \log q \rangle_q - \langle \log p(x|\theta) \rangle_q \\
&= -\frac{1}{N} \sum_{n=1}^{N} \log p(x^n|\theta)) + \text{const.}
\end{aligned}$$

Hence setting parameters of p that maximise the likelihood is equivalent to setting parameters of p that minimise the KL divergence between p and the empirical distribution.

Maximum Likelihood BN training and counting

A BN takes the form:

$$p(x) = \prod_{i=1}^{K} p(x_i|\mathrm{pa}\,(x_i))$$

For the BN $p(x)$, and empirical distribution $q(x)$ we have

$$\begin{aligned}
\mathrm{KL}(q|p) &= -\left\langle \sum_{i=1}^{K} \log p\,(x_i|\mathrm{pa}\,(x_i)) \right\rangle_{q(x)} + \text{const.} \\
&= -\sum_{i=1}^{K} \langle \log p\,(x_i|\mathrm{pa}\,(x_i)) \rangle_{q(x_i,\mathrm{pa}(x_i))} + \text{const.} \\
&= \sum_{i=1}^{K} \left[\langle \log q(x_i|\mathrm{pa}\,(x_i)) \rangle_{q(x_i,\mathrm{pa}(x_i))} - \langle \log p\,(x_i|\mathrm{pa}\,(x_i)) \rangle_{q(x_i,\mathrm{pa}(x_i))} \right] + \\
&= \sum_{i=1}^{K} \langle \mathrm{KL}(q(x_i|\mathrm{pa}\,(x_i))|p(x_i|\mathrm{pa}\,(x_i))) \rangle_{q(\mathrm{pa}(x_i))} + \text{const.}
\end{aligned}$$

Maximum Likelihood BN training and counting

$$\mathrm{KL}(q|p) = \sum_{i=1}^{K} \left\langle \mathrm{KL}(q(x_i|\mathrm{pa}\,(x_i))|p(x_i|\mathrm{pa}\,(x_i))) \right\rangle_{q(\mathrm{pa}(x_i))} + \text{const.}$$

The minimal Kullback-Leibler setting, and that which corresponds to Maximum Likelihood, is therefore

$$p(x_i|\mathrm{pa}\,(x_i)) = q(x_i|\mathrm{pa}\,(x_i))$$

In terms of the original data, this is

$$p(x_i = \mathsf{s}|\mathrm{pa}\,(x_i) = \mathsf{t}) \propto \sum_{n=1}^{N} \mathbb{I}\left[x_i^n = \mathsf{s}\right] \prod_{x_j \in \mathrm{pa}(x_i)} \mathbb{I}\left[x_j^n = \mathsf{t}^j\right]$$

The table entry $p(x_i|\mathrm{pa}\,(x_i))$ can be set by counting the number of times the state $\{x_i = \mathsf{s}, \mathrm{pa}\,(x_i) = \mathsf{t}\}$ occurs in the dataset (where t is a vector of parental states). The table is then given by the relative number of counts of being in state s compared to the other states s′, for fixed joint parental state t.

Naive Bayes Classifier

A joint model of observations $\mathbf{x}$ and the corresponding class label c using a Belief network of the form

$$p(\mathbf{x}, c) = p(c) \prod_{i=1}^{D} p(x_i|c)$$

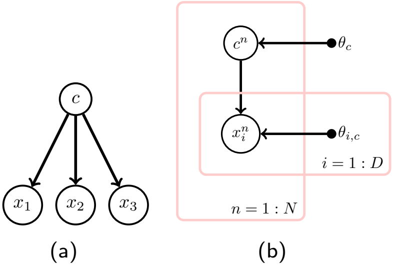


Figure: Naive Bayes classifier. **(a)**: The central assumption is that given the class c, the attributes x_i are independent. **(b)**: Assuming the data is i.i.d., Maximum Likelihood learns the optimal parameters of the distribution $p(c)$ and the class-dependent attribute distributions $p(x_i|c)$.

Coupled with a suitable choice for each conditional distribution $p(x_i|c)$, we can then use Bayes' rule to form a classifier for a novel attribute vector $\mathbf{x}^*$:

$$p(c|\mathbf{x}^*) = \frac{p(\mathbf{x}^*|c)p(c)}{p(\mathbf{x}^*)} = \frac{p(\mathbf{x}^*|c)p(c)}{\sum_c p(\mathbf{x}^*|c)p(c)}$$

Naive Bayes example

Consider the following vector of attributes:

$$\text{(likes shortbread, likes lager, drinks whiskey, eats porridge, watched England play football)}$$

Together with each vector $\mathbf{x}$, there is a label nat describing the nationality of the person, $\mathrm{dom}(nat) = \{\text{scottish}, \text{english}\}$.
We can use Bayes' rule to calculate the probability that $\mathbf{x}$ is Scottish or English:

$$\begin{aligned} p(\text{scottish}|\mathbf{x}) &= \frac{p(\mathbf{x}|\text{scottish})p(\text{scottish})}{p(\mathbf{x})} \\ &= \frac{p(\mathbf{x}|\text{scottish})p(\text{scottish})}{p(\mathbf{x}|\text{scottish})p(\text{scottish}) + p(\mathbf{x}|\text{english})p(\text{english})} \end{aligned}$$

For $p(\mathbf{x}|nat)$ under the Naive Bayes assumption:

$$p(\mathbf{x}|nat) = p(x_1|nat)p(x_2|nat)p(x_3|nat)p(x_4|nat)p(x_5|nat)$$

0	1	1	1	0	0
0	0	1	1	1	0
1	1	0	0	0	0
1	1	0	0	0	1
1	0	1	0	1	0

(a) English

1	1	1	1	1	1	1
0	1	1	1	1	0	0
0	0	1	0	0	1	1
1	0	1	1	1	1	0
1	1	0	0	1	0	0

(b) Scottish

Using Maximum Likelihood we have: $p(\text{scottish}) = 7/13$ and $p(\text{english}) = 6/13$.

$$\begin{aligned}
p(x_1 = 1|\text{english}) &= 1/2 & \quad p(x_1 = 1|\text{scottish}) &= 1 \\
p(x_2 = 1|\text{english}) &= 1/2 & \quad p(x_2 = 1|\text{scottish}) &= 4/7 \\
p(x_3 = 1|\text{english}) &= 1/3 & \quad p(x_3 = 1|\text{scottish}) &= 3/7 \\
p(x_4 = 1|\text{english}) &= 1/2 & \quad p(x_4 = 1|\text{scottish}) &= 5/7 \\
p(x_5 = 1|\text{english}) &= 1/2 & \quad p(x_5 = 1|\text{scottish}) &= 3/7
\end{aligned}$$

For $\mathbf{x} = (1, 0, 1, 1, 0)^{\mathsf{T}}$, we get

$$p(\text{scottish}|\mathbf{x}) = \frac{1 \times \frac{3}{7} \times \frac{3}{7} \times \frac{5}{7} \times \frac{4}{7} \times \frac{7}{13}}{1 \times \frac{3}{7} \times \frac{3}{7} \times \frac{5}{7} \times \frac{4}{7} \times \frac{7}{13} + \frac{1}{2} \times \frac{1}{2} \times \frac{1}{3} \times \frac{1}{2} \times \frac{1}{2} \times \frac{6}{13}} = 0.8076$$

Since this is greater than 0.5, we would classify this person as being Scottish.

Bayesian Belief Net training

We continue with the Asbestos, Smoking, Cancer scenario,

$$p(a, c, s) = p(c|a, s)p(a)p(s)$$

and a set of visible observations, $\mathcal{V} = \{(a^n, s^n, c^n), n = 1, \ldots, N\}$. With all variables binary we have parameters such as

$$p(a = 1|\theta_a) = \theta_a, \quad p(c = 1|a = 0, s = 1, \theta_c) = \theta_c^{0,1}$$

The parameters are

$$\theta_a, \theta_s, \underbrace{\theta_c^{0,0}, \theta_c^{0,1}, \theta_c^{1,0}, \theta_c^{1,1}}_{\theta_c} \tag{1}$$

In Bayesian learning of BNs, we need to specify a prior on the joint table entries. Since in general dealing with multi-dimensional continuous distributions is computationally problematic, it is useful to specify only uni-variate distributions in the prior. As we show below, this has a pleasing consequence that for i.i.d. data the posterior also factorises into uni-variate distributions.

Global parameter independence

A convenient assumption is that the prior factorises over parameters. For our Asbestos, Smoking, Cancer example, we assume

$$p(\theta_a, \theta_s, \theta_c) = p(\theta_a)p(\theta_s)p(\theta_c)$$

Assuming the data is i.i.d., we then have the joint model

$$p(\theta_a, \theta_s, \theta_c, \mathcal{V}) = p(\theta_a)p(\theta_s)p(\theta_c) \prod_n p(a^n|\theta_a)p(s^n|\theta_s)p(c^n|s^n, a^n, \theta_c)$$

Learning then corresponds to inference of

$$p(\theta_a, \theta_s, \theta_c|\mathcal{V}) = \frac{p(\mathcal{V}|\theta_a, \theta_s, \theta_c)p(\theta_a, \theta_s, \theta_c)}{p(\mathcal{V})} = \frac{p(\mathcal{V}|\theta_a, \theta_s, \theta_c)p(\theta_a)p(\theta_s)p(\theta_c)}{p(\mathcal{V})}$$

The posterior also factorises, since

$$\begin{aligned} p(\theta_a, \theta_s, \theta_c|\mathcal{V}) &\propto p(\theta_a, \theta_s, \theta_c, \mathcal{V}) \\ &= \left\{p(\theta_a)\prod_n p(a^n|\theta_a)\right\}\left\{p(\theta_s)\prod_n p(s^n|\theta_s)\right\}\left\{p(\theta_c)\prod_n p(c^n|s^n, a^n, \theta_c)\right\} \\ &\propto p(\theta_a|\mathcal{V}_a)p(\theta_s|\mathcal{V}_s)p(\theta_c|\mathcal{V}_c) \end{aligned}$$

Local parameter independence

If we further assume that the prior for the table factorises over all states a, c:

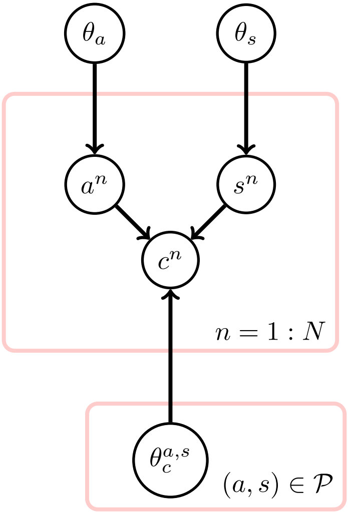
$$p(\theta_c) = p(\theta_c^{0,0})p(\theta_c^{1,0})p(\theta_c^{0,1})p(\theta_c^{1,1})$$

then the posterior

$$\begin{aligned} p(\theta_c|\mathcal{V}_c) &\propto p(\mathcal{V}_c|\theta_c)p(\theta_c^{0,0})p(\theta_c^{1,0})p(\theta_c^{0,1})p(\theta_c^{1,1}) \\ &= \underbrace{\left[\theta_c^{0,0}\right]^{\sharp(a=0,s=0)} p(\theta_c^{0,0})}_{\propto p(\theta_c^{0,0}|\mathcal{V}_c)} \underbrace{\left[\theta_c^{0,1}\right]^{\sharp(a=0,s=1)} p(\theta_c^{0,1})}_{\propto p(\theta_c^{0,1}|\mathcal{V}_c)} \\ &\times \underbrace{\left[\theta_c^{1,0}\right]^{\sharp(a=1,s=0)} p(\theta_c^{1,0})}_{\propto p(\theta_c^{1,0}|\mathcal{V}_c)} \underbrace{\left[\theta_c^{1,1}\right]^{\sharp(a=1,s=1)} p(\theta_c^{1,1})}_{\propto p(\theta_c^{1,1}|\mathcal{V}_c)} \end{aligned}$$

so that the posterior also factorises over the parental states of the local conditional table.

Global and Local independence



Using a Beta prior

$$p(\theta_a) = B\left(\theta_a|\alpha_a, \beta_a\right) = \frac{1}{B(\alpha_a, \beta_a)}\theta_a^{\alpha_a - 1}\left(1-\theta_a\right)^{\beta_a - 1}$$

for which the posterior is also a Beta distribution:

$$p(\theta_a|\mathcal{V}_a) = B\left(\theta_a|\alpha_a + \sharp\left(a=1\right), \beta_a + \sharp\left(a=0\right)\right)$$

The marginal table is given by

$$p(a=1|\mathcal{V}_a) = \int_{\theta_a} p(\theta_a|\mathcal{V}_a)\theta_a = \frac{\alpha_a + \sharp\left(a=1\right)}{\alpha_a + \sharp\left(a=1\right) + \beta_a + \sharp\left(a=0\right)}$$

---

hyperparameters

The prior parameters α_a, β_a are called hyperparameters. If one had no preference, one would set $\alpha_a = \beta_b = 1$.

Bayes vs ML

$$p(a=1|\mathcal{V}_a) = \int_{\theta_a} p(\theta_a|\mathcal{V}_a)\theta_a = \frac{\alpha_a + \sharp(a=1)}{\alpha_a + \sharp(a=1) + \beta_a + \sharp(a=0)}$$

Corresponds in this case to adding 'pseudo counts' to the data.

No data limit

The marginal probability table corresponds to the prior ratios:

$$p(a=1) = \frac{\alpha_a}{\alpha_a + \beta_a}$$

For a flat prior $\alpha = \beta = 1$, $p(a=1) = 0.5$.

Infinite data limit

The marginal probability tables are dominated by the data counts:

$$p(a=1|\mathcal{V}) \to \frac{\sharp(a=1)}{\sharp(a=1) + \sharp(a=0)}$$

which corresponds to the Maximum Likelihood solution.

Summary

- Maximum Likelihood in general corresponds to the intuitive use of ‘counting’ to set tables
- When there are no counts of a particular configuration, the learned probabilities are zero. This can have severe effects in classifiers such as Naive Bayes.
- The Bayesian approach places priors on the tables.
- Convenient to assume global parameter independence since then the posterior factorises over the tables (assuming i.i.d.)
- Convenient also to assume local parameter independence of each conditional since then the posterior table factorises over its parental states.
- A very simple classifier is Naive Bayes. A Bayesian treatment is equivalent to using ‘pseudo counts’ and avoids overfitting.
- Naive Bayes is extremely popoular. (Spam filtering, credit scoring, ...)