Lecture 3: Evidence and Message Passing

In the last lecture we looked at calculating probabilities in a simple decision tree. In this lecture we will generalise the process into a message passing algorithm. For this purpose, it is useful to introduce the notion of evidence. Consider the "eyes" node in the tree we introduced in the last lecture:

$$P'(E) = \alpha P(E)P(S|E)P(D|E)$$

The constant α was calculated to normalise the probabilities so that they sum to 1. Without the normalisation step we refer to the un-normalised probabilities as evidence.

$$\epsilon(E) = P(E)P(S|E)P(D|E)$$

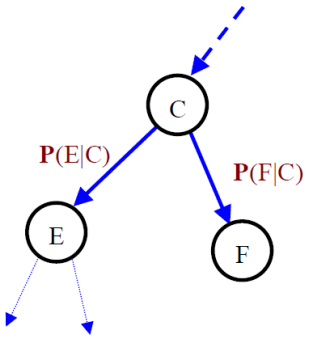
We can further subdivide the evidence into the part that comes from the parents and the parts that come from the children. For the case of the simple inference in a tree we can write:

Evidence for E from the parents: $\pi(E) = P(E)$
Evidence for E from the children: $\lambda(E) = P(S|E)P(D|E)$

We use λ to mean the likelihood evidence and π to represent the prior evidence. Each has a value for each state of E, and we can write it as a vector using the notation $\boldsymbol{\lambda(E)} = [\lambda(e_1), \lambda(e_2), \lambda(e_3)]$. Evidence is useful since, in many cases, it is not necessary to keep normalising our results to form probabilities, and simplification of the equations is possible. In the case of the node E we observe that the λ evidence can be divided into the part that comes from node S - $P(S|E)$ - and the part that comes from node D - $P(D|E)$. These individual parts from each child are called λ messages.

Conditioning

Calculating the λ evidence from children is simple if the child node is instantiated, however for nodes higher up a tree the picture is more complex. Consider the case of calculating the likelihood (λ) evidence at node C. If we could instantiate E then we would simply select the appropriate entry from the link matrices. For example if E was in state e_2 and F was in state f_3 we would write:



$$\lambda(c_1) = P(e_2|c_1)P(f_3|c_1)$$

Suppose however we don't know the exact state of E. As we saw in lecture 2 we will have likelihood evidence for each state of E, and we need to average the link matrix entries according to this likelihood evidence thus:

$$\lambda(c_1) = [\lambda(e_1)P(e_1|c_1) + \lambda(e_2)P(e_2|c_1) + \lambda(e_3)P(e_3|c_1)]P(f_3|c_1)$$

This process is called conditioning. In general to find the λevidence for a node we use the conditioning equation:

$$\lambda(c_i) = \prod_{(children)} \sum_j \lambda(h_j)P(h_j|c_i)$$

Where $\lambda(c_i)$ is the evidence for state c_i of C and h_j is a state of a child node H. For nodes which have been instantiated to known values, only one state has $\lambda(h_k) = 1$ and for all others $\lambda(h_l) = 0$.

Instantiation and Evidence

If we take a measurement, for example, for the separation measure S, which for example is 0.35 we instantiate the corresponding state (s_5). We can express the same idea by stating that $\lambda(s_5) = 1$ (any positive number would do since λ evidence is not normalised) and for all other states s_i, $\lambda(s_i) = 0$. This is equivalent to setting the evidence as in the following table.

State	Range	$\lambda(s_i)$
s_1	$[below - 1.5]$	0
s_2	$[-1.5 \cdots - 0.75]$	0
s_3	$[-0.75 \cdots - 0.25]$	0
s_4	$[-0.25 \cdots 0.25]$	0
s_5	$[0.25 \cdots 0.75]$	1
s_6	$[0.75 \cdots 1.5]$	0
s_7	$[above 1.5]$	0

Types of evidence

Sometimes, when we make a measurement it is possible to express uncertainty about it by distributing the evidence values. For example, instead of setting $\lambda(s_5) = 1$ we could set evidence values using a Gaussian distribution centered on the actual measurement:

$$\boldsymbol{\lambda(s_i)} = (0, 0, 0.08, 0.3, 0.5, 0.1, 0.02)$$

This is often referred to as *virtual evidence*. If we use virtual evidence at the children nodes we need to use the full conditioning equation as well.

Sometimes, it is not possible to make a measurement, or data may be missing for a particular variable. In this case we wish to express the condition that there is *no evidence* and set all the evidence values to be the same. For convenience we choose the value 1.

$$\boldsymbol{\lambda(s_i)} = (1, 1, 1, 1, 1, 1, 1)$$

If we expand the conditioning equation for our two variables S and D, and we put no evidence in S we can see that the term involving S simply evaluates to 1 and therefore does not affect the evidence for E.

$$\lambda(e_i) = (\textstyle\sum_j \lambda(s_j)P(s_j|e_i))(\sum_k \lambda(d_k)P(d_k|e_i))$$
$$\lambda(e_i) = (\textstyle\sum_j P(s_j|e_i))(\sum_k \lambda(d_k)P(d_k|e_i))$$
$$\lambda(e_i) = \textstyle\sum_k \lambda(d_k)P(d_k|e_i)$$

Up until now we have mostly been considering the case where the root node is the hypothesis we wish to evaluate. However, this need not be the case For example we may be solely interested in whether the image we have just processed shows eyes, rather than a cat.

λ Messages

We can consider the process of calculating the likelihood (λ) evidence of a node combining messages from each individual child. If a child node, say node S,has some λ evidence of its own, we can consider that a message to be passed to its parent (E). The message is passed by means of the conditional probability matrix. For a single parent case this is achieved using a matrix equation:

$$\boldsymbol{\lambda_S(E) = \lambda(S)P(S|E)}$$

Prior (π) evidence and messages

We noted last lecture that for nodes other than root nodes we have no explicit prior, and in its place we calculate the evidence it received from its parent. If, to be specific, our problem is to determine the probability distribution over the node E then we can distinguish two cases (one being a simplification of the other). The first is where we know whether there is a cat in the picture, in other words we instantiate C to one of its states. It is important to note that if C is instantiated we know its value, and the λ evidence from any of its children becomes irrelevant, and is no longer used. In particular the value of F will not appear anywhere in the calculation of the probability distribution over C or E. This is intuitively plausible, since the F node does not relate at all to the presence of eyes in the picture. Furthermore, with knowledge of the state of C the link matrix from E to C simplifies to what is essentially a prior probability for node E. If $\boldsymbol{P(C)} = (1, 0)$ then

$$\boldsymbol{P}(\boldsymbol{E}) = \begin{bmatrix} P(e_1|c_1) & P(e_1|c_2) \\ P(e_2|c_1) & P(e_2|c_2) \\ P(e_3|c_1) & P(e_3|c_2) \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} P(e_1|c_1) \\ P(e_2|c_1) \\ P(e_3|c_1) \end{bmatrix}$$

Our decision tree simplifies to the naive Bayes network shown in figure 1, and the calculation of the probabilities is a simple matter of applying Bayes' theorem.

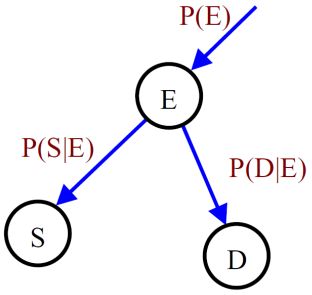


Figure 1: Network after C is instantiated

in the more general case, we might not know for certain that there was a cat in the picture, and are only able to instantiate instantiate S D and F. Clearly the geometric evidence from below still stands, but instead of having a prior probability of a pair of eyes we need to determine the evidence from the cat node that there is a pair of eyes. Instead of C being in a known state, we may simply have evidence for it being in a particular state. This evidence on C has an influence on the state of E, in the same way that the prior probability had in the first case. The evidence that is passed from a parent to a child is made up of the evidence for the parent node collected from all sources except the child. In the last lecture we treated it as a probability and denoted it $P_{-E}(C)$. To generalise the process we will treat it as evidence called the π message that the parent sends to a child, using the notation $\pi_E(C)$ for it.

The π evidence for the child node is calculated by multiplying the link matrix by the π message from the parent. In the case of node C this other evidence for the node consists of its prior probability and the λ evidence from its other child F. Suppose for a given picture we calculate the λ evidence for C from F as:

$$\boldsymbol{\lambda_F(C)} = [\lambda_F(c_1), \lambda_F(c_2)] = [0.3, 0.2]$$

and the prior probability of C is:

$$P(C) = [0.6, 0.4]$$

the total evidence for C, excluding any evidence from E, is found by multiplying the individual evidence values:

$$\boldsymbol{\pi_E(C)} = [0.3 \times 0.6, 0.2 \times 0.4] = [0.18, 0.08]$$

This is the π message to E from C. The π evidence for E, which is written as $\boldsymbol{\pi(E)}$ is found as follows:

$$\boldsymbol{\pi(E)} = \boldsymbol{P(E|C)\pi_E(C)}$$

$$\boldsymbol{\pi(E)} = \begin{bmatrix} P(e_1|c_1) & P(e_1|c_2) \\ P(e_2|c_1) & P(e_2|c_2) \\ P(e_3|c_1) & P(e_3|c_2) \end{bmatrix} \begin{bmatrix} 0.18 \\ 0.08 \end{bmatrix} = \begin{bmatrix} 0.18P(e_1|c_1) + 0.08P(e_1|c_2) \\ 0.18P(e_2|c_1) + 0.08P(e_2|c_2) \\ 0.18P(e_3|c_1) + 0.08P(e_3|c_2) \end{bmatrix}$$

Notice that the only difference to previous cases is that we are using evidence, not probabilities. We can combine the evidence at a node by multiplying the λ and π messages received at that node. We can normalise it into a posterior probability of the node (ie the probability of each state of the node given the evidence from the rest of the network). Propagation of evidence in a tree does not require normalisation to take place until all the evidence is accumulated, because probabilities are always combined by multiplication. Probabilities can be found as a final step when we wish to make an inference about a variable.

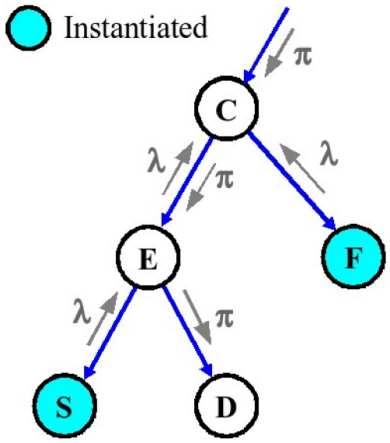
At C, as at other nodes, the evidence can always be normalised to a posterior probability distribution indicated, in this case, by $\boldsymbol{P'(C)}$. We could equally well compute $\pi_E(C)$ by first computing a posterior probability of C, and then dividing the evidence from E out of it. In scalar form this means, for state c_j:

$$P'(c_j) = \alpha P(c_j)\lambda_F(c_j)\lambda_E(c_j)$$

$$\pi_E(C) = P'(c_j)/\lambda_E(c_j)$$

Where α is a normalising constant making $\sum_i P'(c_i) = 1$. The magnitude of the π message will be different, but the resulting evidence passed to E is the same.

It is important to note that it does not matter whether we normalise evidence during propagation in a tree or not. Providing we normalise the evidence of the hypothesis node(s) when all the evidence is collected we will get the same result. The point is that the magnitude of the evidence for a particular state of a node is irrelevant. It is the relative magnitudes of the evidence values for all the states of the nodes that carry significant information.
Propagation of evidence in a tree is both upwards and downwards. The real power of a Bayesian network is in its generality. We can instantiate any given subset of the nodes, and then calculate probability distributions over the unknown nodes, regardless of whether they are root nodes, leaf nodes or intermediate nodes. The adjacent figure shows how evidence is propagated when we instantiate S and F and calculate the evidence for nodes CE and D.



In summary evidence received from a nodes parent is written as a vector equation: $\boldsymbol{\pi(E) = P(E|C)\pi_E(C)}$, where $\boldsymbol{\pi_E(C)}$ means all the evidence for C excluding anything from E. In scalar form it is written:

$$\pi(e_i) = \sum_j [P(e_i|c_j)\pi(c_j) \prod_{k \setminus E} \lambda_k(c_j)]$$

Priors and Likelihood in Networks

For a single node in a network we can associate the notion of prior and likelihood with the evidence being propagated. $\lambda(B_i)$ is the likelihood evidence for B_i. It is derived from its children, and from measurements made from the data. $\pi(B_i)$ is the prior evidence for B_i. It derives from the prior probabilities of the root nodes, and from other evidence in the tree. This notion of prior and likelihood is slightly different from our previous usage and reflects the fact that the network itself (together with the prior probability of the root node(s)) represents our prior knowledge for inference.

Incorporating new information

One of the best features of Bayesian Networks is that we can incorporate new nodes as the data becomes available. Recall that we had information from the computer vision process as to how 'likely' the extracted circles were. This could simply be treated as another node in the network. Specifically we could say that it is providing likelihood evidence for the presence of eyes, so it would appear as a child node of eyes.
The only new parameters that must be found are the elements of the link matrix $P(L|E)$. All other parameters are unchanged. Notice that if we do not have any evidence of this kind, and we set the λ evidence entries for L to be all equal to 1, the network will be identical to the previous inference network.