Lecture 20

Inference in Belief Network (cont'd)

In this last lecture, we will go over a couple of examples using the belief network to compute posterior probabilities.

We will also have a few words about the exam.

Reference:

- Textbook Chapter 15, section 15.4

The algorithm

Let's have some revision first. The algorithm consists of two recurrences:

- $P(X \mid E) = \alpha P(E_X^- \mid X)\sum_{\mathbf{u}} P(X|\mathbf{u})\prod_i P(U_i = u_i|E_{U_i \setminus X})$
- $P(E_X^- \mid X)$

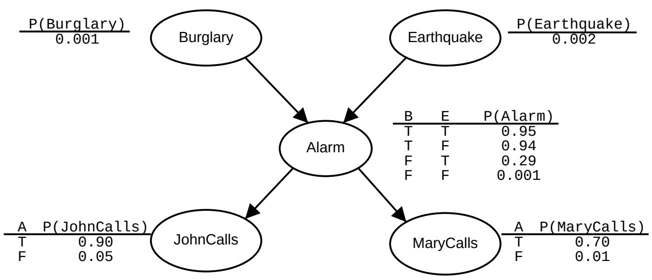
  $= \beta \prod_i \sum_{y_i} P(E_{Y_i}^- \mid Y_i = y_i) \sum_{\mathbf{z}} P(Y_i = y_i \mid X, \mathbf{z}) \prod_j P(Z_{ij} = z_{ij} \mid E_{Z_{ij} \setminus Y_i})$
- In each of the formulas, the first term is a **recursion** to the second formula, the second term is a **CPT lookup**, the third term is a **recursion** to the first formula.
- We will basically ignore the β in the second recurrence, expecting that the normalization of the first recurrence will deal with it.
- The recursion terminates when it **hits an evidence variable**, and when it **reaches the top or the bottom of the network**.

The terminating cases

- For the first (upstream) formula (for $P(X \mid E_X^+)$):
  - If X is itself an **evidence** variable: return **1** or **0** accordingly, no recursion needed.
  - If X has **no parent** (no U variable): use **prior** probability of X, recurse on other downstream branches if needed.
- For the second (downstream) formula (for $P(E_X^- \mid X)$):
  - If E_X^- is **empty**: return **1**, no recursion needed. (Empty proposition is always true.)
  - If E_X^- **contains** one of the Y: we will not need to list out all the possible values of Y in the outermost sum. Instead, just use the known value of Y.

    The algorithm in the book miss this case. You can go to the web page of the book to find a corrected version.

Let use the same example again!



We want two probabilities:

- $P(B \mid J, M)$, the one we have calculated last time.
- $P(J \mid B, M)$, a mixed reasoning.

What we need

Let's try to use the two recurrence for the old probability $P(B \mid J, M)$…

- Use the **upstream** formula, $P(B \mid J, M) = \alpha P(J, M \mid B)P(B)$.

  Special case: at top of tree, use prior probability.
- Now we need to find $P(J, M \mid B)$, a **downstream** probability. We invoke the downstream formula:

  $P(J, M \mid B) = \beta_1 \sum_a P(J, M \mid A = a) \sum_e P(A = a \mid B, E = e)P(E = e)$

  a and e are both just true or false.
- Here we need the **downstream** probability $P(J, M \mid A = a)$. We invoke the downstream formula again. This time we have 2 branches, multiplied together:

  $P(J, M \mid A = a) = \beta_2 \big(P(J \mid A = a)\big)\big(P(M \mid A = a)\big)$
- And… we are basically done! We just need to plug in the values…

Putting in the values

- $P(J, M \mid A) = \beta_2 P(J \mid A)P(M \mid A) = \beta_2 \times 0.9 \times 0.7 = \beta_2 \times 0.63$
- $P(J, M \mid \neg A) = \beta_2 P(J \mid \neg A)P(M \mid \neg A) = \beta_2 \times 0.05 \times 0.01 = \beta_2 \times 0.0005$
- $P(J, M \mid B) = \beta_1 \big(P(J, M \mid A)(\sum_e P(A \mid B, E = e)P(E = e))$
  $+ P(J, M \mid \neg A)(\sum_e P(\neg A \mid B, E = e)P(E = e))\big)$

  $= \beta_1\beta_2(0.63 \times (P(A \mid B, E)P(E) + P(A \mid B, \neg E)P(\neg E))$
  $+ 0.0005 \times (P(\neg A \mid B, E)P(E) + P(\neg A \mid B, \neg E)P(\neg E)))$

  $= \beta_1\beta_2(0.63 \times (0.95 \times 0.002 + 0.94 \times 0.998)$
  $+ 0.0005 \times (0.05 \times 0.002 + 0.06 \times 0.998))$

  $= \beta_1\beta_2 \times 0.59224259$

  This looks very complicated. But in fact it is very simple: we are given B, so we list out all the possibilities of A and E, and add all of them up.

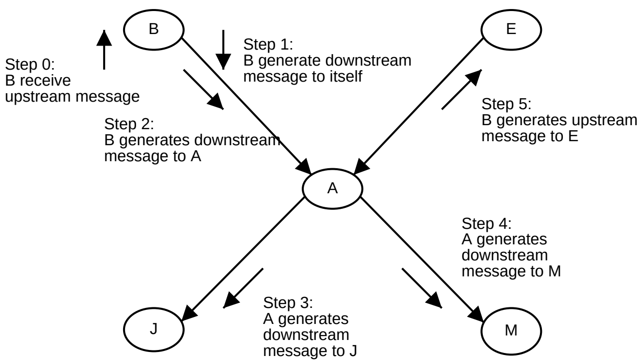
Putting in the values (cont'd)

- $P(J, M \mid \neg B) = \beta_1\big(P(J, M \mid A)(\sum_e P(A \mid \neg B, E = e)P(E = e))$
  $+ P(J, M \mid \neg A)(\sum_e P(\neg A \mid \neg B, E = e)P(E = e))\big)$
  $= \beta_1\beta_2(0.63 \times (P(A \mid \neg B, E)P(E) + P(A \mid \neg B, \neg E)P(\neg E))$
  $+ 0.0005 \times (P(\neg A \mid \neg B, E)P(E) + P(\neg A \mid \neg B, \neg E)P(\neg E)))$
  $= \beta_1\beta_2(0.63 \times (0.29 \times 0.002 + 0.0001 \times 0.998)$
  $+ 0.0005 \times (0.71 \times 0.002 + 0.999 \times 0.998))$
  $= \beta_1\beta_2 \times 0.001493351$
- $P(B \mid J, M) = \alpha \times 0.001 \times \beta_1\beta_2 \times 0.59224259 = \alpha\beta_1\beta_2 \times 0.00059224259.$
- $P(\neg B \mid J, M) = \alpha \times 0.999 \times \beta_1\beta_2 \times 0.001493351$
  $= \alpha\beta_1\beta_2 \times 0.001491857649$
- Now, normalization give us $P(B \mid J, M) = 28.4\%$.

The value propagation view

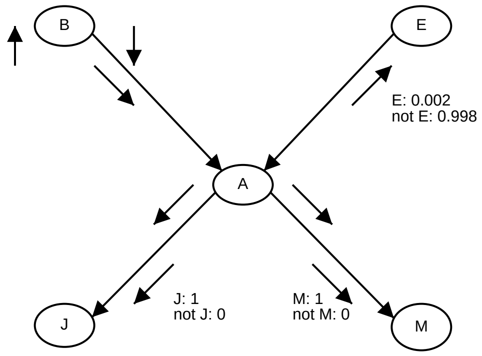
- The above calculation is sound, but it can be **visualized** better if we do the calculation directly on the network.
- We can view the calculation as **nodes sending messages** upstream and downstream to obtain upstream and downstream probability.
  A message is a request for either the upstream or downstream probability.
- At the beginning we put an **upstream** message to the **query** variable.
- When a node receives an upstream message, it will send upstream messages to its **parents**, and send downstream message to **itself**.
- When a node receives a downstream message, it will send downstream messages to its its **children**, and upstream messages to **parents of its children**.
  Except that the upstream message is not sent back to where the downstream message comes from.

Repeating our example the last time…



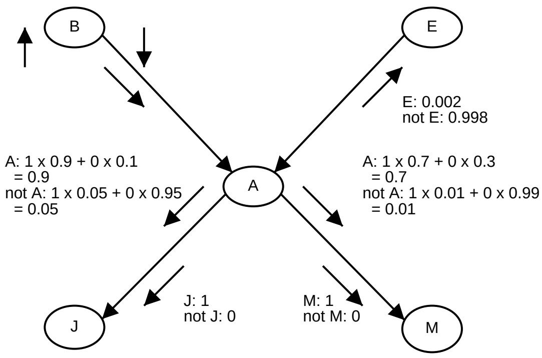
The calculations…

Let's list out those values that can be found directly…



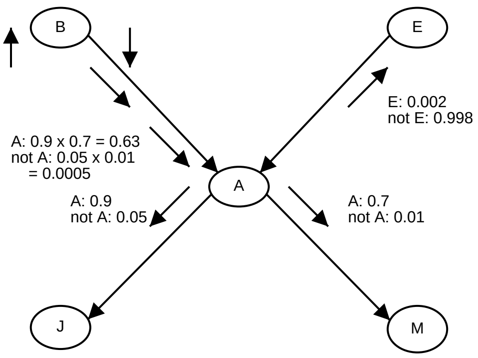
The calculations (2)…

A computes the contributions of the downstream probabilities…



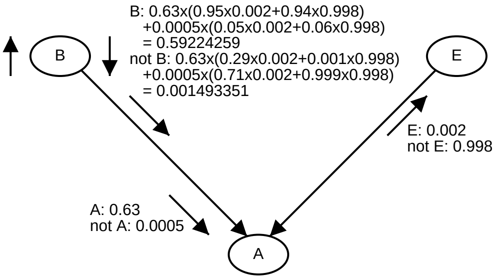
The calculations (3)…

Contributions are combined, and put back up the network.



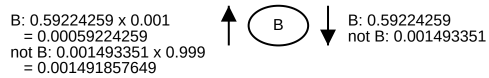
The calculations (4)…

B computes the contributions of the downstream probabilities…



The calculations (5)…

There is only 1 contribution of *B*, so combining it gives the same result. Now we find the upstream probability.



So all the calculations are just the same, except that we are much more clear about where we are.

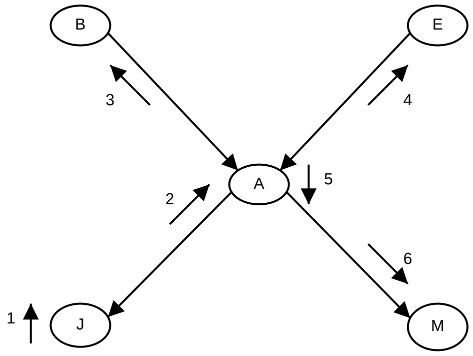
Now let's repeat the exercise for $P(J \mid B, \neg M)$

Illustrating how upstream probabilities are propagated, and how to deal with negated evidence.

This time we will just draw the network view.

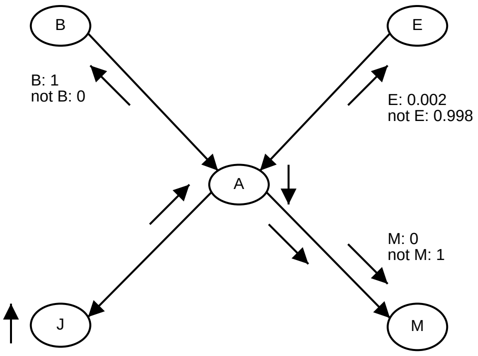
The messages

These messages are sent through the network…



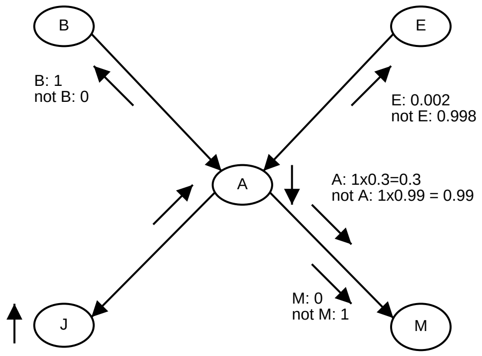
Our new exercise (1)

Again, we first show the things we can obtain directly.



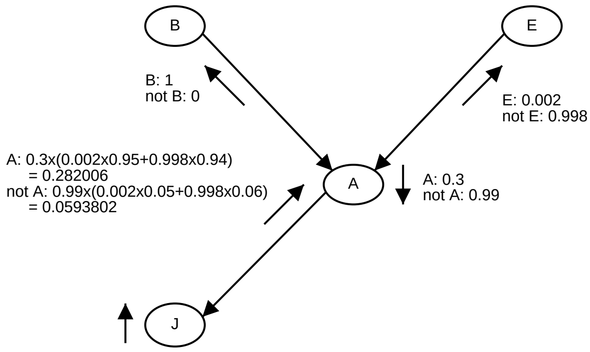
Our new exercise (2)

A use the downstream probability of *M* to calculate its contribution to the downstream probability of *A*.



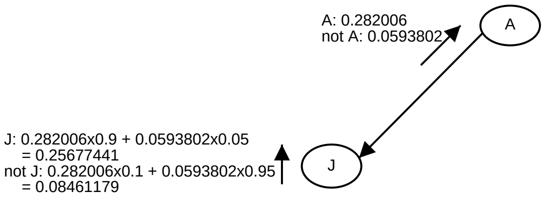
Our new exercise (3)

There is no other children, so it is also the downstream probability of *A*. It is combined with the upstream prob to find upstream prob of *A*.



Our new exercise (4)

With upstream probability of *A*, we find the upstream probability of *J*…



Then we do normalization to find the probability 75.2%.

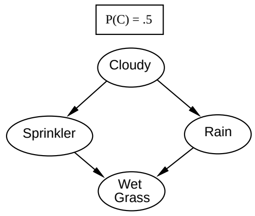
So actually we don't need to refer to the two recurrences once we know what we need to do.

One handy place to go to verify your answer…

`http://www-2.cs.cmu.edu/~javabayes/index.html`

Dealing with multiply connected network

- A multiply connected network looks like this:



C	P(S)
T	.10
F	.50

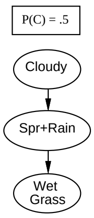
C	P(R)
T	.80
F	.20

S	R	P(W)
T	T	.99
T	F	.90
F	T	.90
F	F	.00

- Unluckily, the problem of probabilistic inference becomes **NP-complete** if the network is multiply connected.

Clustering

- One solution: merge offending nodes into a "**mega-node**".



C	P(S+R=x) TT	TF	FT	FF
T	.08	.02	.72	.18
F	.40	.10	.40	.10

S+R	P(W)
T T	.99
T F	.90
F T	.90
F F	.00

- It is essential that the mega-node is small enough: the size of the resulting network is exponential to the size of the number of nodes represented by the mega-node.
  Other approaches has been suggested, but we will not cover them.

A few words about exam…

- Includes lecture 2–20.
- Includes 7 multi-part questions, each 10% of the marks.
- Marks of top 3 scoring questions will be double-counted.
  So it is important to get a couple of them to score completely.
- Include questions closely related to assignment, and include some other questions.
- You can bring 1A4 paper into the exam hall, with any written or printed materials. But 1A4 paper means 1A4 paper, not 1A4 paper with many smaller-sized paper sticked onto it.
  Last year my Algorithm course has one student doing so, and the whole sheet of paper is taken away from him.