17-0: **Probabilistic Reasoning**

- Given:
  - Set of conditional probabilities ($P(t1|d)$, etc)
  - Set of prior probabilities ($P(d)$)
  - Conditional independence information ($P(t1|d, t2) = P(t1|d)$)
- We can calculate any quantity that we like
- Problems:
  - Hard to know exactly what data we need
  - Even given sufficient data, calculations can be complex – especially dealing with conditional independence

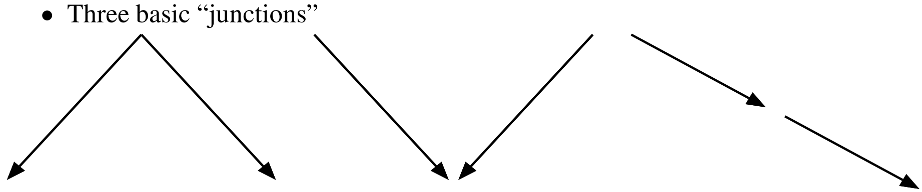
17-1: **Bayesian Networks**

Bayesian Networks are:

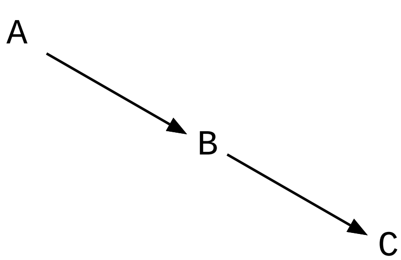
- Clever encoding of conditional independence information
- Mechanical, "turn the crank" method for calculation
  - Can be done by a computer

Nothing "magic" about Bayesian Networks 17-2: **Directed Acyclic Graphs**

- We will encode conditional independence information using Directed Acyclic Graphs (or DAGs)
- While we will use causal language to give intuitive justification, these DAGs are *not necessarily* causal (more on this later)
- Three basic "junctions"

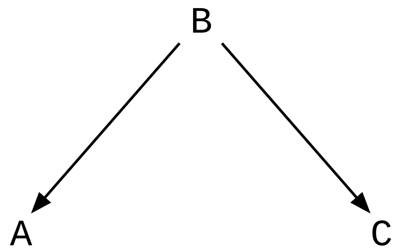


17-3: **Head-to-Tail**



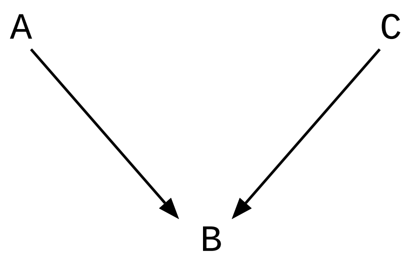
- "Causal Chain"
- Rain → Wet Pavement → Slippery Pavement
  - $(A \not\perp\!\!\!\perp C)$
  - $(A \perp\!\!\!\perp C|B)$

17-4: **Tail-to-Tail**



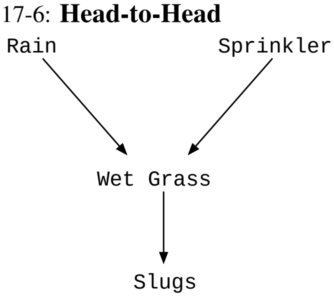
- "Common Cause"
- Reading Ability ← Age → Shoe Size
  - $(A \not\perp\!\!\!\perp C)$
  - $(A \perp\!\!\!\perp C|B)$

17-5: **Head-to-Head**



- "Common Effect"
- Rain → Wet Grass ← Sprinkler
  - $(A \perp\!\!\!\perp C)$
  - $(A \not\perp\!\!\!\perp C|B)$

17-6: **Head-to-Head**



- Also need to worry about descendants of head-head junctions.
- (Rain $\perp\!\!\!\perp$ Sprinkler)
- (Rain $\not\perp\!\!\!\perp$ Sprinkler | Slugs)

17-7: **Markovian Parents**

- V is an ordered set of variables $X_1, X_2, \ldots X_n$.
- $P(V)$ is a joint probability distribution over V
- Define the set of Markovian Parents of variable X_j, PA_j as:
  - Minimal set of predecessors of X_j such that
  - $P(X_j|X_1, \ldots X_{j-1}) = P(X_j|PA_j)$
- The Markovian Parents of a variable X_j are often *(but not always)* the direct causes of X_j

17-8: **Markovian Parents & Joint**

- For any set of variables $X_1, \ldots X_n$, we can calculate any row of the joint:
  - $P(x_1, ...x_n) = P(x_1)P(x_2|x_1)P(x_3|x_1, x_2) \ldots$
    $P(x_n|x_1, x_2, \ldots x_{n-1})$
- Using Markovian parents
  - $P(x_1, ...x_n) = P(x_1)P(x_2|PA_2)P(x_3|PA_3) \ldots$
    $P(x_n|PA_n)$

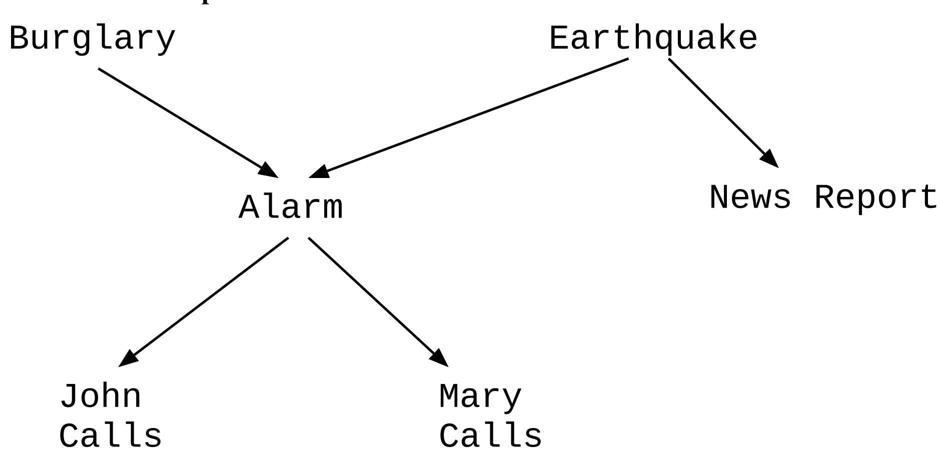
17-9: **Markovian Parents & DAGs**

- We can create a DAG which represents conditional independence information using Markovian parents.
  - Each variable is a node in the graph
  - For each variable X_j, add a directed link from all elements in PA_j to X_j

17-10: **Burglary Example**

- I want to know if my house has been robbed
- I install an alarm
  - Have two neighbors, John & Mary, who call me if they hear my alarm
- Small earthquakes could also set off the alarm
- Sometimes, small earthquakes are reported on the radio
- Variables:
  - Burglary, Earthquake, News Report, Alarm, John Calls, Mary Calls

17-11: **DAG Example**

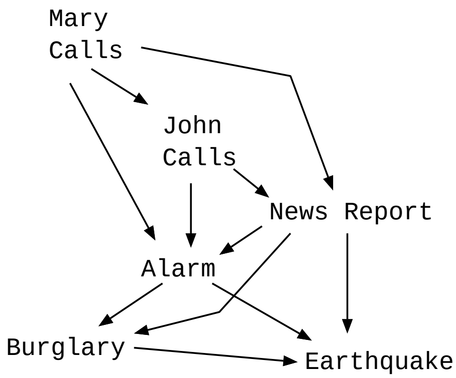


17-12: **Markovian Parents & DAGs**

- The order that we consider variables is important!
- Causal ordering gives "best" DAGS, but non-causal works, too
- Example: Mary Calls, John Calls, News Report, Alarm, Burglary, Earthquake

17-13: **DAG Example**

- Order: Mary Calls, John Calls, News Report, Alarm, Burglary, Earthquake

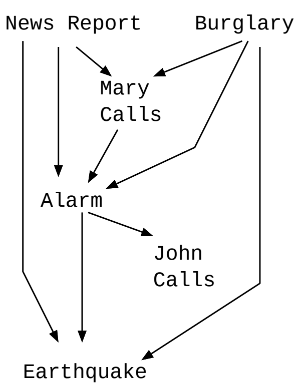


17-14: **Markovian Parents & DAGs**

- The order that we consider variables is important!
- Causal ordering gives "best" DAGS, but non-causal works, too
- Example: News Report, Burglary, Mary Calls, Alarm, John Calls, Earthquake

17-15: **DAG Example**

- Order: News Report, Burglary, Mary Calls, Alarm, John Calls, Earthquake

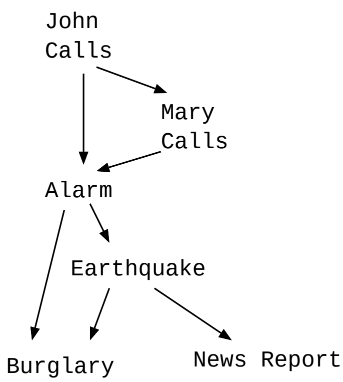


17-16: **Markovian Parents & DAGs**

- The order that we consider variables is important!
- Causal ordering gives "best" DAGS, but non-causal works, too
- Example: John Calls, Mary Cals, Alarm, Earthquake, News Report, Burglary,

17-17: **DAG Example**

- Order: John Calls, Mary Cals, Alarm, Earthquake, News Report, Burglary,



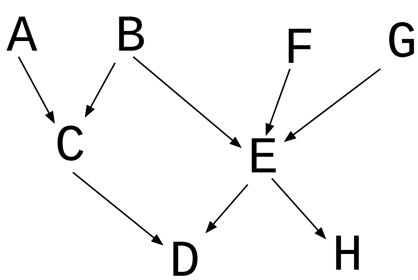
17-18: **DAGs & Cond. Independence**

- Given a DAG of Markovian Parents, we know that every variable X_i is independent of its ancestors, given its parents
- We also know quite a bit more

17-19: **d-separation** To determine if a variable X is conditionally independent of Y given a set of variables Z:

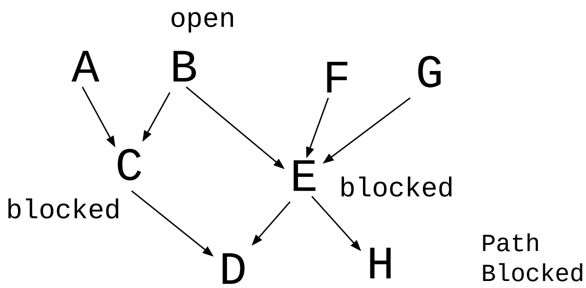
- Examine all paths between X and Y in the graph
- Each node along a path can be “open” or “blocked”
  - A node at a head-to-tail or tail-to-tail junction is open if the node is not in Z, and closed otherwise.
  - A node at a head-to-head junction is open if the node *or any of its descendants* is not in Z, and closed otherwise.

17-20: **d-separation Examples**



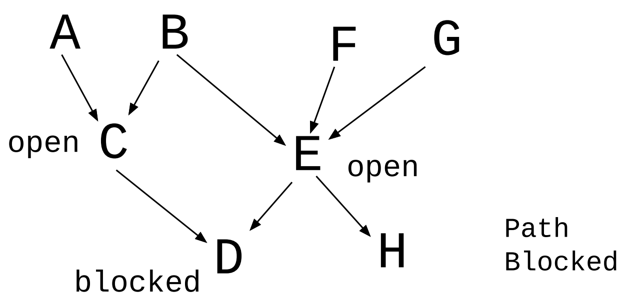
$(A \perp\!\!\!\perp G)$?

17-21: **d-separation Examples**



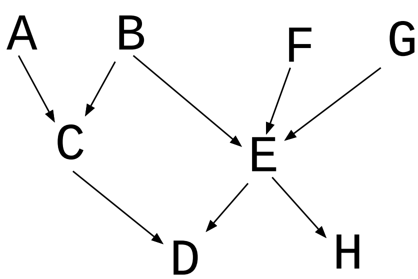
$(A \perp\!\!\!\perp G)?$

17-22: **d-separation Examples**



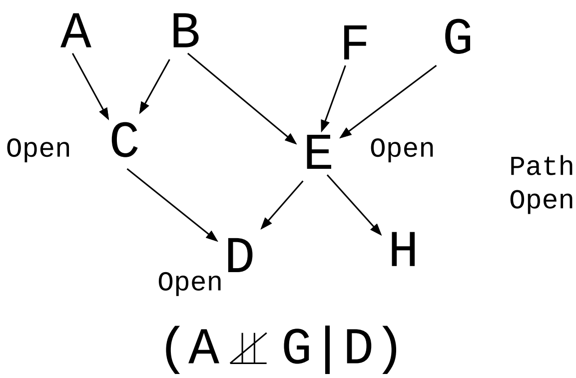
$(A \perp\!\!\!\perp G)!$

17-23: **d-separation Examples**

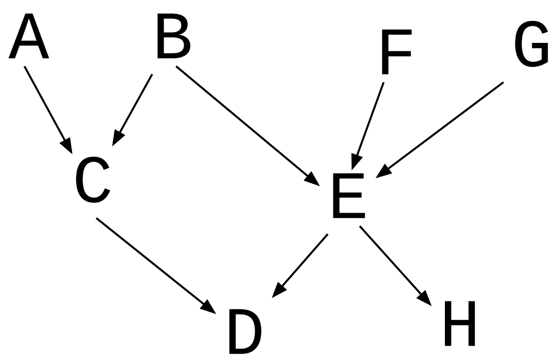


$(A \perp\!\!\!\perp G | D)?$

17-24: **d-separation Examples**



17-25: **d-separation Examples**



17-26: **Bayesian Networks**

To build a Bayesian Network:

- Select variables
- Order variables
    - Normally want a *causal* ordering
- Compute Markovian parents for each variable
- Compute $P(X_i|PA_i)$ for each variable

17-27: **Test / Courier Example**

P(D)	D = ~d	D = d
	0.999	0.001

Disease

P(T|D)	T = ~t	T = t
D = ~d	0.9	0.1
D = d	0.1	0.9

Test

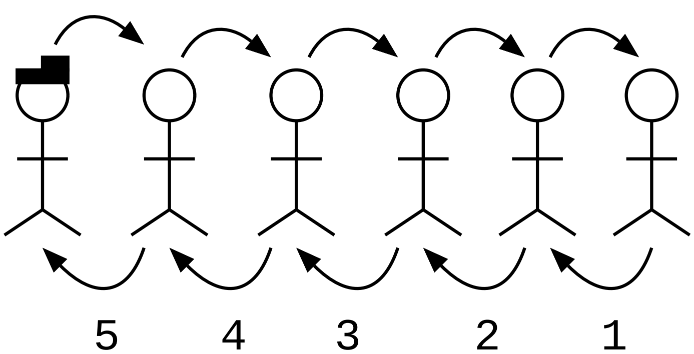
P(C|T)	C = ~c	C = c
T = ~t	0.95	0.05
T = t	0.1	0.9

Courier

17-28: **Message Passing**

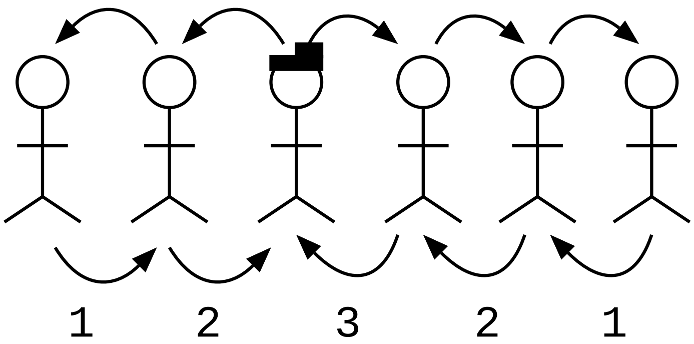
- Once we have our Bayesian Network, we will calculate probabilities using message passing
- Example:
  - Leader of a group of troops wants to know how many soldiers are in the group
  - Sends a "count" message down line of soldiers
  - Gets a count reply back

17-29: **Message Passing**

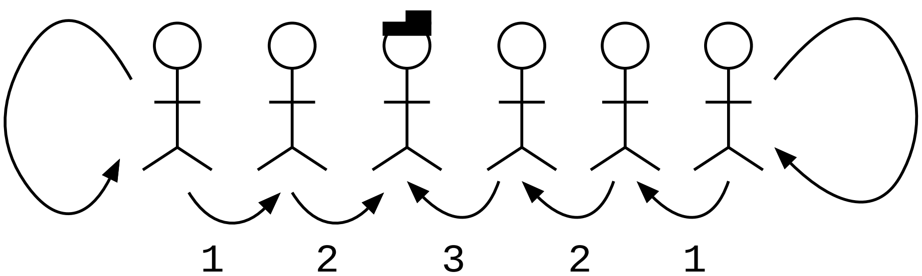


Platoon leader counting soldiers

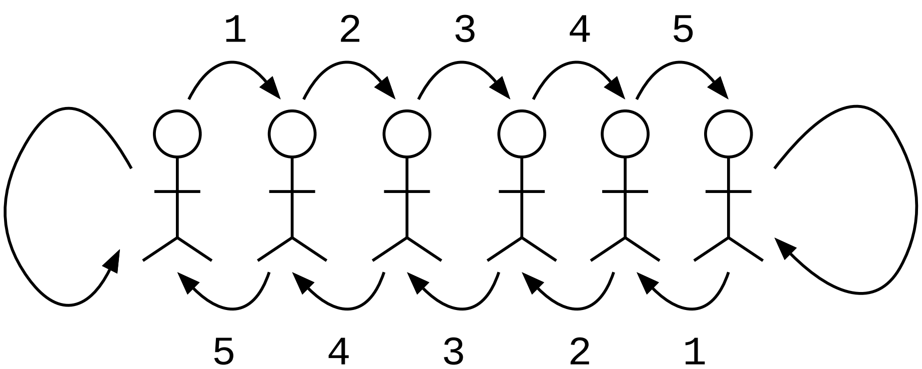
17-30: **Message Passing**



Platoon leader counting soldiers, from middle of line 17-31: **Message Passing**



Platoon leader counting soldiers, with self-generating count signal 17-32: **Message Passing**



Leaderless Counting 17-33: **Using Bayesian Networks**

- A patient receives a "positive" result from the courier. Does the patient have the disease?
- What is $P(d|c)$?
- In general, what is $P(d|e)$, where e is all the evidence that we have?

17-34: **Breaking Up Evidence**

- Break evidence e into two pieces
  - "causal evidence" or "causal support", e^+
  - "diagnostic evidence" or "evidential support" e^-

$$\begin{aligned} P(d|e_d^+, e_d^-) &= \frac{P(d|e_d^+)P(e_d^-|d, e_d^+)}{P(e_d^-)} \\ &= \frac{P(d|e_d^+)P(e_d^-|d)}{P(e_d^-)} \\ &= \alpha P(d|e_d^+)P(e_d^-|d) \end{aligned}$$

17-35: **Renaming**

$$\begin{aligned} P(d|e_d^+, e_d^-) &= \frac{P(d|e_d^+)P(e_d^-|d, e_d^+)}{P(e_d^-)} \\ &= \frac{P(d|e_d^+)P(e_d^-|d)}{P(e_d^-)} \\ &= \alpha P(d|e_d^+)P(e_d^-|d) \end{aligned}$$

- $\pi(x) = P(x|e_x^+)$
- $\lambda(x) = P(e_x^-|x)$

Thus, $P(d|e) = \alpha\pi(d)\lambda(d)$

17-36: **Renaming**

$$\begin{aligned} P(x|e_x^+, e_x^-) &= \alpha P(x|e_x^+)P(e_x^-|x) \\ &= \alpha\pi(x)\lambda(x) \end{aligned}$$

- $\pi(x)$ is the "message" from upstream.
- $\lambda(x)$ is the "message" from downstream.

17-37: **Calculating** $\pi(d)$

- $\pi(d)$ is the probability that $D = d$, given upstream evidence for D
- All we have for upstream evidence is the prior probability for D
- $\pi(d)$ = Prior Probability on $d = P(d)$!

17-38: **Calculating** $\lambda(d)$

$$\begin{aligned} \lambda(d) &= P(e_d^-|d) \\ &= \sum_{t \in T} P(e_d^-|d, t)P(t|d) \\ &= \sum_{t \in T} P(e_t^-|t)P(t|d) \\ &= \sum_{t \in T} \lambda(t)P(t|d) \end{aligned}$$

17-39: **Calculating** $\lambda(d)$

$$\lambda(d) = \sum_{t \in T} \lambda(t)P(t|d)$$

$$\begin{aligned} \lambda(\neg d) &= \lambda(\neg t)P(\neg t|\neg d) &+ \lambda(t)P(t|\neg d) \\ \lambda(d) &= \lambda(\neg t)P(\neg t|d) &+ \lambda(t)P(t|d) \end{aligned}$$

17-40: **Calculating** $\lambda(d)$

$$\lambda(d) = \sum_{t \in T} \lambda(t)P(t|d)$$

$\lambda(D) = [\lambda(\neg d), \lambda(d)]$

17-41: **Calculating** $\lambda(D)$

$$\lambda(d) = \sum_{t \in T} \lambda(t)P(t|d)$$

$$\begin{aligned} \lambda(D) &= [\lambda(\neg d), \lambda(d)] \\ &= [\lambda(\neg t)P(\neg t|\neg d) + \lambda(t)P(t|\neg d), \lambda(\neg t)P(\neg t|d) + \lambda(t)P(t|d)] \end{aligned}$$

17-42: **Calculating** $\lambda(D)$

$$\lambda(d) = \sum_{t \in T} \lambda(t)P(t|d)$$

$$\begin{aligned} \lambda(D) &= [\lambda(\neg d), \lambda(d)] \\ &= [\lambda(\neg t)P(\neg t|\neg d) + \lambda(t)P(t|\neg d), \lambda(\neg t)P(\neg t|d) + \lambda(t)P(t|d)] \\ &= \left[\begin{array}{cc} P(\neg t|\neg d) & P(t|\neg d) \\ P(\neg t|d) & P(t|d) \end{array} \right] \left[\begin{array}{c} \lambda(\neg t) \\ \lambda(t) \end{array} \right] \end{aligned}$$

17-43: **Calculating** $\lambda(D)$

$$\lambda(d) = \sum_{t \in T} \lambda(t)P(t|d)$$

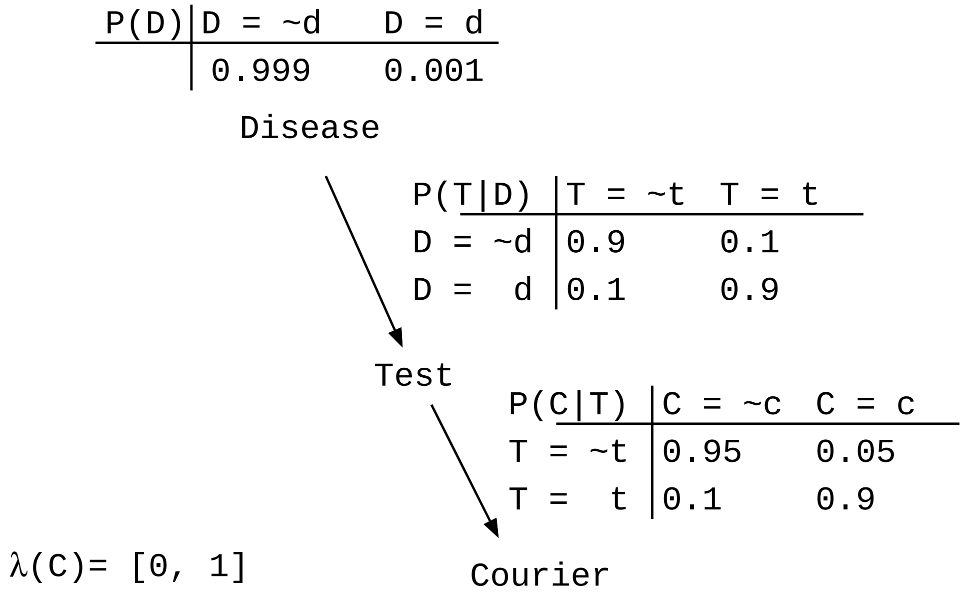
$$\begin{aligned} \lambda(D) &= [\lambda(\neg d), \lambda(d)] \\ &= [\lambda(\neg t)P(\neg t|\neg d) + \lambda(t)P(t|\neg d), \lambda(\neg t)P(\neg t|d) + \lambda(t)P(t|d)] \\ &= \left[\begin{array}{cc} P(\neg t|\neg d) & P(t|\neg d) \\ P(\neg t|d) & P(t|d) \end{array} \right] \left[\begin{array}{c} \lambda(\neg t) \\ \lambda(t) \end{array} \right] \\ &= P(T|D)\lambda(T) \\ &= M_{T|D}\lambda(T) \end{aligned}$$

17-44: **Calculating** $\lambda(D)$

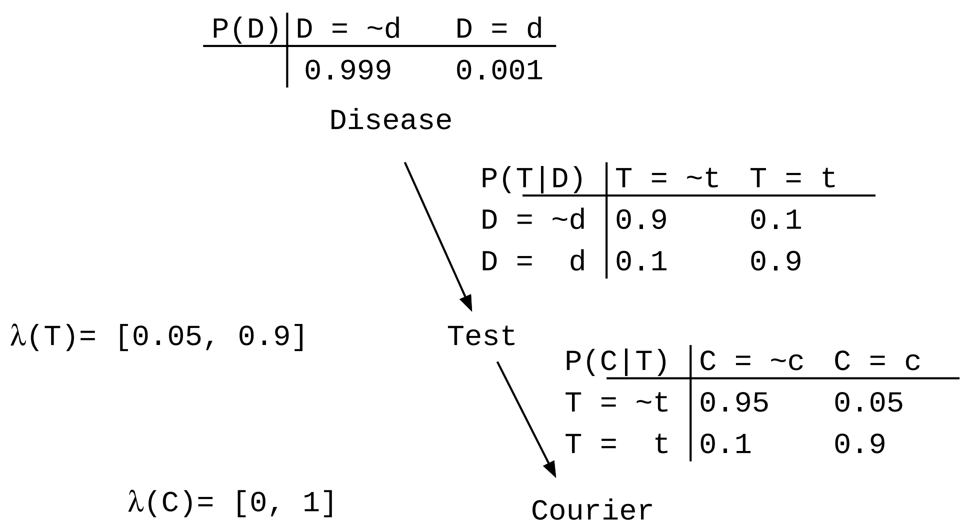
- $\lambda(D) = M_{T|D}\lambda(T)$
- $\lambda(T) = M_{C|T}\lambda(C)$
- $\lambda(C) = ?$
  - What is the evidence that $C = \neg c, C = c$?

- We know that $C = c$
- $\lambda(C) = [0, 1]$

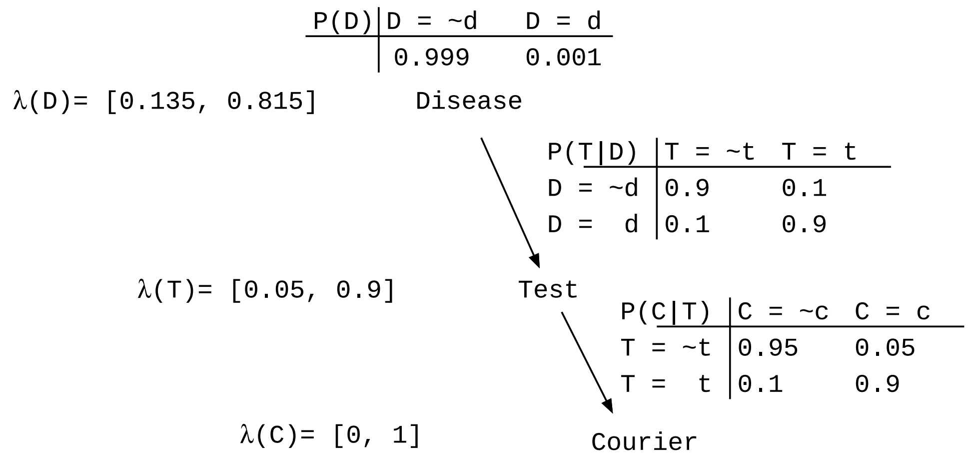
17-45: **Test / Courier Example**



17-46: **Test / Courier Example**



17-47: **Test / Courier Example**



17-48: **Calculating** $P(D|e)$

- $\lambda(C) = [0, 1]$
- $\lambda(T) = M_{C|T}\lambda(C) = [0.05, 0.9]$
- $\lambda(D) = M_{T|D}\lambda(T) = [0.135, 0.815]$

From before, $\pi(D) = P(D) = [0.999, 0.001]$

- $P(D|e) = \alpha\pi(D)\lambda(D)$
- $P(D|e) = \alpha[0.999, 0.001][0.135, 0.815]$
- $P(D|e) = \alpha[0.134865, 0.000815]$
  - $\alpha = 1/0.13568$
- $P(D|e) = [0.993993, 0.006007]$

17-49: **Calculating** $P(T|e)$

- What if we wanted to calculate the probability that the test actually was positive, given that the courier delivered a positive result?
- $P(T|e) = \alpha\pi(T)\lambda(T)$
- We know $\lambda(T)$ from before
- What is $\pi(T)$?

17-50: **Calculating** $\pi(t)$

$$\begin{aligned}
\pi(t) &= P(t|e_t^+) \\
&= \sum_{d \in D} P(t|d, e_t^+)P(d|e_t^+) \\
&= \sum_{d \in D} P(t|d, e_d^+)P(d|e_d^+) \\
&= \sum_{d \in D} P(t|d)P(d|e_d^+) \\
&= \sum_{d \in D} P(t|d)\pi(d)
\end{aligned}$$

17-51: **Calculating $\pi(t)$**

$$
\begin{aligned}
\pi(t) &= P(t|e_t^+) \\
&= \sum_{d \in D} P(t|d, e_t^+) P(d|e_t^+) \\
&= \sum_{d \in D} P(t|d, e_d^+) P(d|e_d^+) \\
&= \sum_{d \in D} P(t|d) P(d|e_d^+) \\
&= \sum_{d \in D} P(t|d) \pi(d)
\end{aligned}
$$

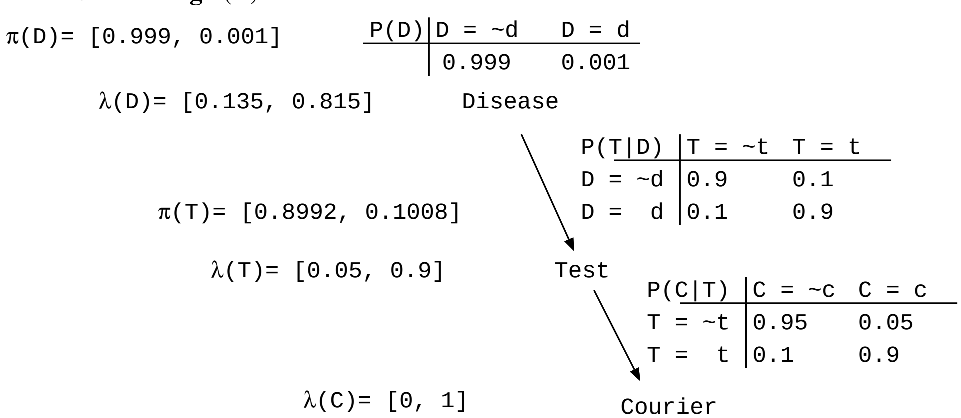
$\pi(\neg t) = P(\neg t|\neg d)P(\neg d|e_d^+) + P(\neg t|d)P(d|e_d^+)$
$\pi(t) = P(t|\neg d)P(\neg d|e_d^+) + P(t|d)P(d|e_d^+)$

17-52: **Calculating $\pi(T)$**

$$\pi(t) = \sum_{d \in D} P(t|d)\pi(d)$$

$\pi(T) = [\pi(\neg t), \pi(t)]$
$= [P(\neg t|\neg d)\pi(\neg d) + P(\neg t|d)\pi(d), P(t|\neg d)\pi(\neg d) + P(t|d)\pi(d)]$
$= [\pi(\neg d), \pi(d)] \begin{bmatrix} P(\neg t|\neg d) & P(t|\neg d) \\ P(\neg t|d) & P(t|d) \end{bmatrix}$

17-53: **Calculating $\pi(T)$**



17-54: **Calculating $BEL(T) = P(T|e)$**

- $BEL(T) = \alpha\pi(T)\lambda(T)$
  - $\lambda(T) = [0.05, 0.9]$
  - $\pi(T) = [0.8992, 0.1008]$
  - $\pi(T)\lambda(T) = [0.04496, 0.09072]$
  - $\alpha = 1/(0.04496 + 0.09072) = 1/(0.13568)$
- $BEL(T) = [0.331368, 0.668632]$

17-55: **Computation for Chains**

- Calculating π messages:
  - π(root) = Prior on root
  - For any other variable X with parent P,
    $\pi(X) = \pi(P)M_{X|P}$
- Calculating λ messages:
  - λ(leaf) = evidence for leaf
    - ([1, 1, . . . , 1] if no evidence)
  - For any other variable X with child C, $\lambda(X) = M_{C|X}\lambda(C)$

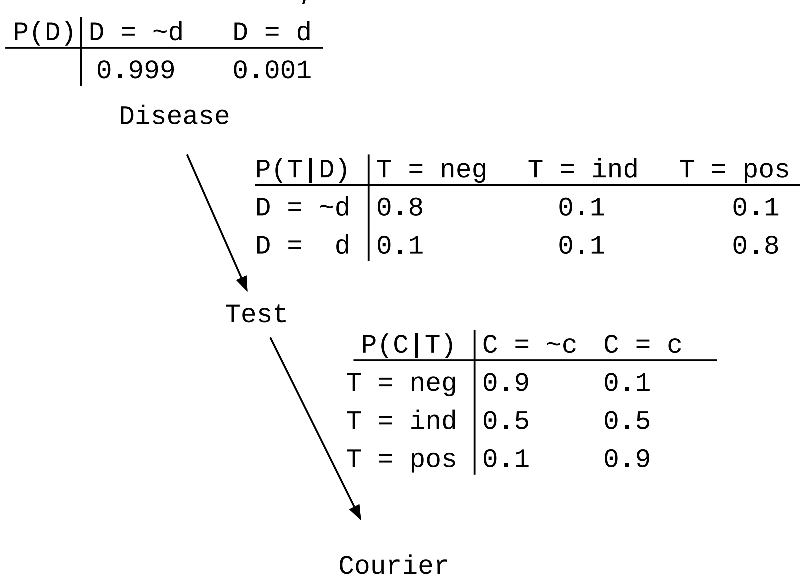
17-56: **Computation for Chains**

- Send π messages down
- Send λ messages up
- For any variable X, we can calculate $BEL(X) = P(X|e)$ by multiplying the messages together, and normalizing
  - $P(X|e) = \alpha\lambda(X)\pi(X)$
    - (Pairwise multiplication)

17-57: **Variable # of Values / Variables**

- Of course, variables can have > 2 values
- Each variable can have a different number of values
- Disease Example
  - Doctor test for a disease
  - Test can be positive, indeterminate, or negative
  - Doctor discusses the result with the courier
  - Courier delivers result

17-58: **Variable # of Values / Variables**

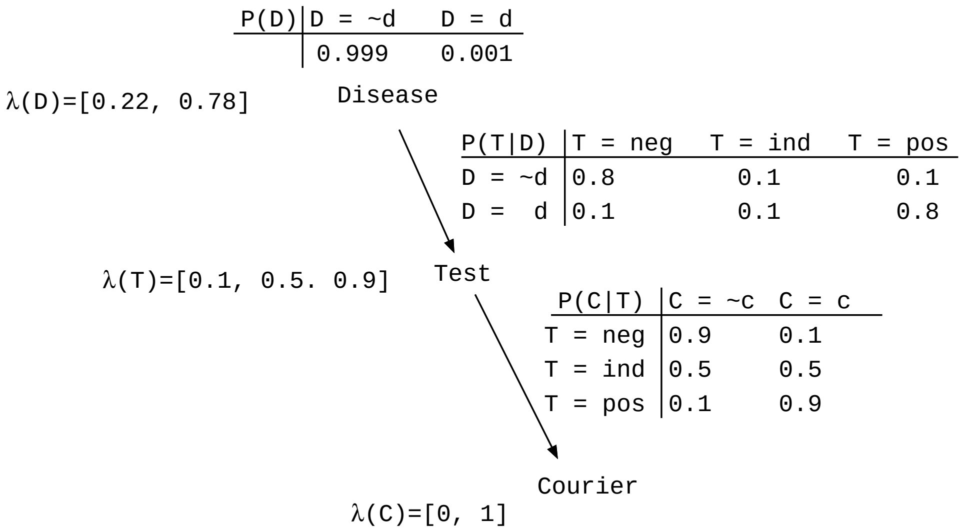


P(D)	D = ~d	D = d
	0.999	0.001

P(T|D)	T = neg	T = ind	T = pos
D = ~d	0.8	0.1	0.1
D = d	0.1	0.1	0.8

P(C|T)	C = ~c	C = c
T = neg	0.9	0.1
T = ind	0.5	0.5
T = pos	0.1	0.9

17-59: **Variable # of Values / Variables**



P(D)	D = ~d	D = d
	0.999	0.001

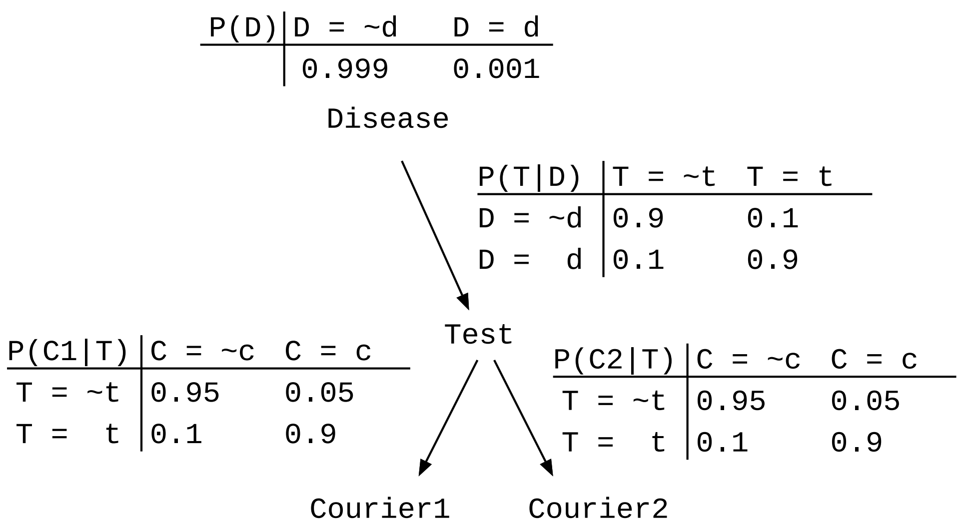
P(T|D)	T = neg	T = ind	T = pos
D = ~d	0.8	0.1	0.1
D = d	0.1	0.1	0.8

P(C|T)	C = ~c	C = c
T = neg	0.9	0.1
T = ind	0.5	0.5
T = pos	0.1	0.9

17-60: **Computation for Trees**

- What if some of the nodes have > 1 child?
- Example: Send message via two different couriers

17-61: **Computation for Trees**



P(D)	D = ~d	D = d
	0.999	0.001

P(T|D)	T = ~t	T = t
D = ~d	0.9	0.1
D = d	0.1	0.9

P(C1|T)	C = ~c	C = c
T = ~t	0.95	0.05
T = t	0.1	0.9

P(C2|T)	C = ~c	C = c
T = ~t	0.95	0.05
T = t	0.1	0.9

17-62: **Computation for Trees**

- How do we send λ messages in trees?
- Courier example: What is $\lambda(T)$, which is the probability of the downstream evidence given the test, if both couriers give a positive response?
- We will need to combine the messages that we get from each child into a single λ message
  - Use this λ message to compute $BEL(T)$
  - Use this λ message to send a message to D

17-63: **Calculating $\lambda(t)$**

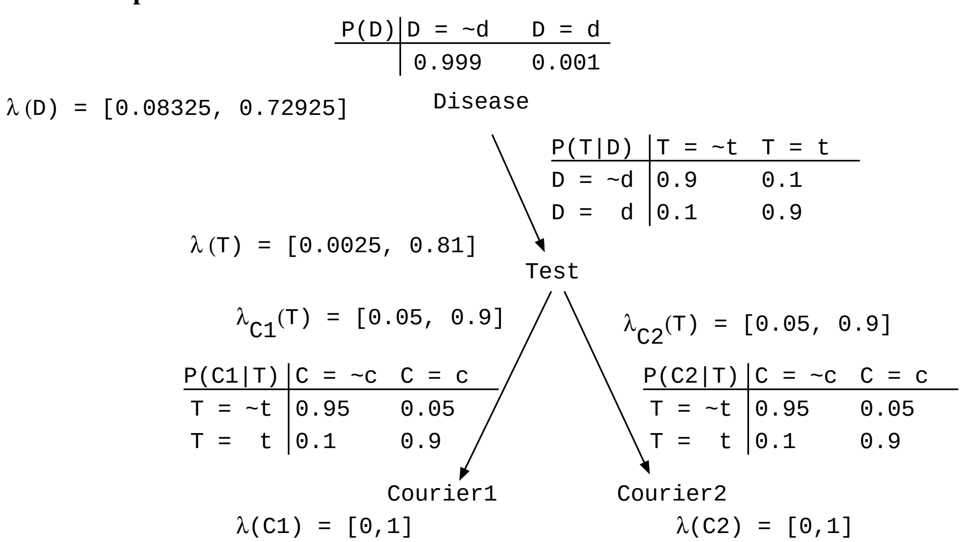
$$\begin{aligned}
\lambda(t) &= P(e_t^-|t) \\
&= P(e_{C1}^-, e_{C2}^-|t) \\
&= P(e_{C1}^-|t)P(e_{C2}^-|t) \\
&= \sum_{c_1 \in C1} P(e_{C1}^-|c_1, t)P(c_1|t) \sum_{c_2 \in C2} P(e_{C2}^-|c_2, t)P(c_2|t) \\
&= \sum_{c_1 \in C1} P(e_{C1}^-|c_1)P(c_1|t) \sum_{c_2 \in C2} P(e_{C2}^-|c_2)P(c_2|t) \\
&= \sum_{c_1 \in C1} \lambda(c_1)P(c_1|t) \sum_{c_2 \in C2} \lambda(c_2)P(c_2|t)
\end{aligned}$$

17-64: **Calculating $\lambda(T)$**

$$\lambda(t) = \sum_{c_1 \in C1} \lambda(c_1)P(c_1|t) \sum_{c_2 \in C2} \lambda(c_2)P(c_2|t)$$

$$\begin{aligned}
\lambda(T) &= M_{C1|T}\lambda(C1) * M_{C2|T}\lambda(C2) \\
&= \lambda_{C1}(T) * \lambda_{C2}(T)
\end{aligned}$$

17-65: **Computation for Trees**



17-66: **Computation for Trees**

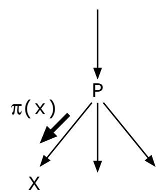
- $BEL(D) = \alpha\pi(D)\lambda(D)$
  - $\pi(D) = [0.999, 0.001]$
  - $\lambda(D) = [0.08325, 0.72925]$

- $\pi(D)\lambda(D) = [0.0831667, 0.00072925]$
- $\alpha = 1/(0.08389595)$

- $BEL(D) = [0.991308, 0.008692]$

17-67: **Sending π Messages in Trees**

- $\pi(x) = P(x|e_x^+)$
- That is, $\pi(x)$ is $P(X = x)$, given all upstream evidence from X



- $\pi(X) = P(P|e_X^+)P(X|P)$
- $\pi(P) * \lambda_{\text{other children of P}}(P)M_{X|P}$
- $(BEL(P)/\lambda_X(P))M_{X|P}$
  - Pairwise division

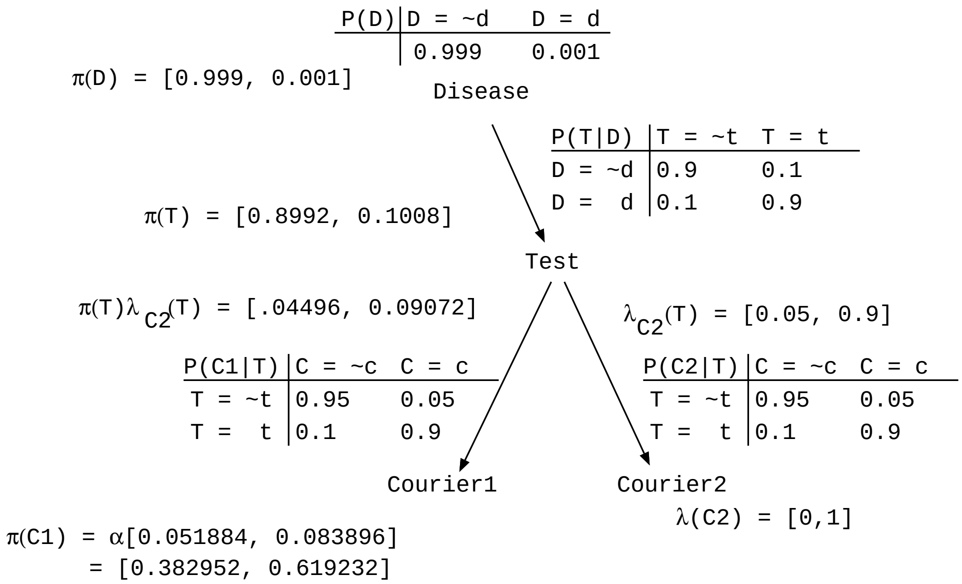
17-68: **Sending π Messages in Trees**

- What is the probability that Courier 1 will give a positive result, given that Courier 2 gave a positive result?
- $P(C1|e)$
  - Evidence e is the prior probability for disease, and the fact that Courier 2 gave a positive result

17-69: **Sending π Messages in Trees**

- What is the probability that Courier 1 will give a positive result, given that Courier 2 gave a positive result?
- $P(C1|e)$
  - Evidence e is the prior probability for disease, and the fact that Courier 2 gave a positive result
- $\pi(C1) = \alpha\pi(T) * \lambda_{C2}(T)M_{C1|T}$

17-70: **Computation for Trees**



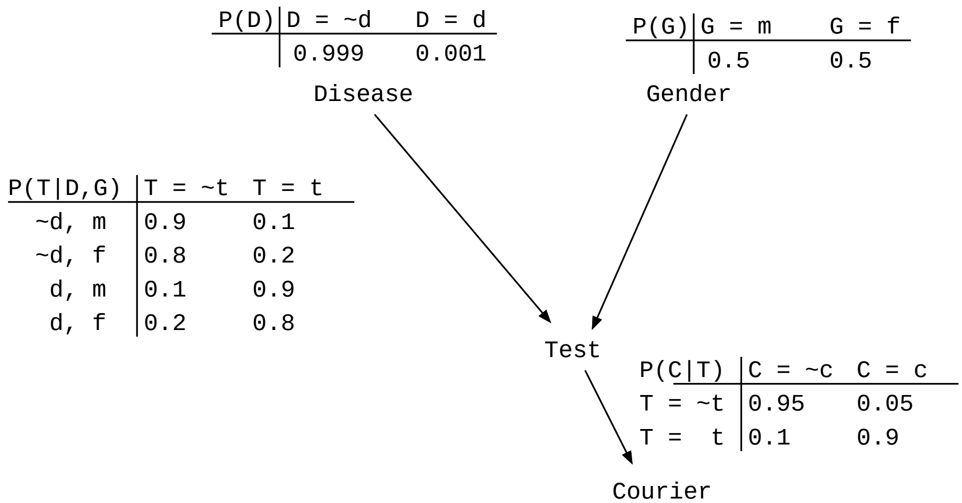
17-71: **Computation for Trees**

- For root variable R, $\pi(R)$ = Prior on R
- For unobserved leaf variables L, $\lambda(L) = [1, 1, \ldots, 1]$
- For leaf variables L observed to have the value l_k, $\lambda(L) = [0, \ldots, 0, 1, 0, \ldots 0]$ – the k^{th} element is 1, all others are 0
- Pass π and λ messages through the tree
  - Multiply π message by λ messages from other childen, them multiply the result by the link matrix
  - Multiply link matrix by λ messages
    - Multiple Children – multiply λ messages

17-72: **Multiple Parents (Polytrees)**

- Add a gender variable
- Test for disease depends upon gender, as well as disease state
- Need to expand link matrix for test to include gender
  - Need $P(t|g, d)$ for all values of t, g, d

17-73: **Multiple Parents (Polytrees)**



P(D)	D = ~d	D = d
	0.999	0.001

P(G)	G = m	G = f
	0.5	0.5

P(T|D,G)	T = ~t	T = t
~d, m	0.9	0.1
~d, f	0.8	0.2
d, m	0.1	0.9
d, f	0.2	0.8

P(C|T)	C = ~c	C = c
T = ~t	0.95	0.05
T = t	0.1	0.9

17-74: **Calculating $\pi()$ in Polytrees**

- For each parent X, we have $P(X|e^+)$
  - $P(D) = [0.999, 0.001]$, $P(G) = [0.5, 0, 5]$
- We need the probabilities for all combinations of parents
  - $P(\neg d, m), P(\neg d, f), P(d, m), P(d, f)$
- Parents are *independent* given upstream evidence
  - $P(\neg d, m) = P(\neg d)P(m)$

17-75: **Calculating $\pi()$ in Polytrees**

- We have $[P(\neg d), P(d)]$ and $[P(m), P(f)]$
- We need $[P(\neg d, m), P(\neg d, f), P(d, m), P(d, f)]$
  - $P(\neg d, m) = P(\neg d)P(m)$, $P(\neg d, f) = P(\neg d)P(f)$, etc.
- $P(\neg d, m) = 0.999 * 0.5$, $P(\neg d, f) = 0.999 * 0.5$, $P(d, m) = 0.001 * 0.5$, $P(d, f) = 0.001 * 0.5$
- $P(D, G) = [0.4995, 0.4995, 0.0005.0.0005]$
- $\pi(T) =$

$$\left[\begin{array}{cccc} \pi(\neg d, m) & \pi(\neg d, f) & \pi(d, m) & \pi(d, f) \end{array}\right] \left[\begin{array}{ll} P(\neg t|\neg d, m) & P(t|\neg d, m) \\ P(\neg t|\neg d, f) & P(t|\neg d, f) \\ P(\neg t|d, m) & P(t|d, m) \\ P(\neg t|d, f) & P(t|d, f) \end{array}\right]$$

17-76: **Calculating $\pi(T)$**

$$\left[\begin{array}{cccc} \pi(\neg d, m) & \pi(\neg d, f) & \pi(d, m) & \pi(d, f) \end{array}\right] \left[\begin{array}{ll} P(\neg t|\neg d, m) & P(t|\neg d, m) \\ P(\neg t|\neg d, f) & P(t|\neg d, f) \\ P(\neg t|d, m) & P(t|d, m) \\ P(\neg t|d, f) & P(t|d, f) \end{array}\right]$$

$=$

$$\begin{bmatrix} 0.4995 & 0.4995 & 0.0005 & 0.0005 \end{bmatrix} \begin{bmatrix} 0.9 & 0.1 \\ 0.8 & 0.2 \\ 0.1 & 0.9 \\ 0.2 & 0.8 \end{bmatrix}$$

$$= \begin{bmatrix} 0.8493 & 0.1507 \end{bmatrix}$$

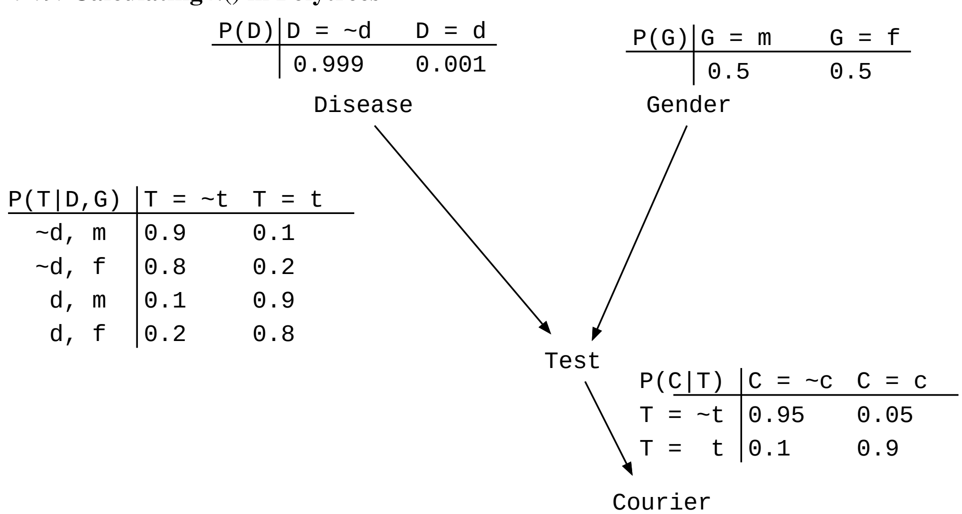
17-77: **Calculating** $BEL(T)$

- What is our belief that the test actually is positive, given that the courier delivers a positive message?
  - $\pi(T) = [0.8493, 0.1507]$
  - $\lambda(T) = \begin{bmatrix} 0.95 & 0.05 \\ 0.1 & 0.9 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix}$
  - $\lambda(T) = [0.05, 0.9]$
- $BEL(T) = \alpha[0.42465, 0.13565]$ ($\alpha = 1/0.5603$)
- $BEL(T) = [0.757898, 0.242102]$

17-78: **Calculating** $\pi()$ **in Polytrees**

- To calculate $\pi(X)$, when X has multiple parents m:
  - For each parent Y_k of X, calculate $P(Y_k|e^+_X)$
    (Define message from Y_k to X, $\pi_X(Y_k) = (Y_k|e^+_X)$
    - If X is the only child of Y_k, $\pi_x(Y_k) = \pi(Y_k)$
    - If Y_k has children $C_1 \ldots C_j$ other than X, then $\pi_X(Y_k) = \pi(Y_k) \prod_{i=i \ldots j} \lambda_{C_i}(Y)$
    - (That is, $\pi_X(Y_k) = BEL(Y)/\lambda_X(Y)$)
  - Combine the π_X messages from all the parents, and multiply the result by the link matrix $M_{X|Y_1 \ldots Y_m}$ to get $\pi(X)$

17-79: **Calculating** $\lambda()$ **in Polytrees**



17-80: **Calculating** $\lambda()$ **in Polytrees**

- How do we send a λ message up to Disease, given the combined link matrix for Disease and Gender? $\begin{bmatrix} P(\neg t|\neg d, m) & P(t|\neg d, m) \\ P(\neg t|\neg d, f) & P(t|\neg d, f) \\ P(\neg t|d, m) & P(t|d, m) \\ P(\neg t|d, f) & P(t|d, f) \end{bmatrix}$

- If we knew that the gender was definitely male, then we could select the appropriate two rows, to create a 2x2 matrix: $\begin{bmatrix} P(\neg t|\neg d, m) & P(t|\neg d, m) \\ P(\neg t|d, m) & P(t|d, m) \end{bmatrix}$

17-81: **Calculating $\lambda()$ in Polytrees**

- How do we send a λ message up to Disease, given the combined link matrix for Disease and Gender? $\begin{bmatrix} P(\neg t|\neg d, m) & P(t|\neg d, m) \\ P(\neg t|\neg d, f) & P(t|\neg d, f) \\ P(\neg t|d, m) & P(t|d, m) \\ P(\neg t|d, f) & P(t|d, f) \end{bmatrix}$
- If we knew that the gender was definitely female, then we could select the appropriate two rows, to create a 2x2 matrix: $\begin{bmatrix} P(\neg t|\neg d, f) & P(t|\neg d, f) \\ P(\neg t|d, f) & P(t|d, f) \end{bmatrix}$

17-82: **Calculating $\lambda()$ in Polytrees**

- If we knew the value of Gender, we could select the correct rows to build the appropriate link matrix to send the lambda message.
- We don't know for certain the value of Gender, but we *do* know the probability G, given evidence upstream of T:
  - $P(G|e_T^+) = \pi_T(G) = \pi(G) = \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}$
- We can then average the rows: $\begin{bmatrix} P(\neg t|\neg d, m) * P(m) + P(\neg t|\neg d, f)P(f) & P(t|\neg d, m)P(m) + P(t|\neg d, f)P(f) \\ P(\neg t|d, m) * P(m) + P(\neg t|d, f)P(f) & P(t|d, m)P(m) + P(t|d, f)P(f) \end{bmatrix}$

17-83: **Calculating $\lambda()$ in Polytrees**

Original Link Matrix $M_{T|D,C}$

$P(T\|D, C)$	$T = \neg t$	$T = t$
$\neg d, m$	0.9	0.1
$\neg d, f$	0.8	0.2
d, m	0.1	0.9
d, f	0.2	0.8

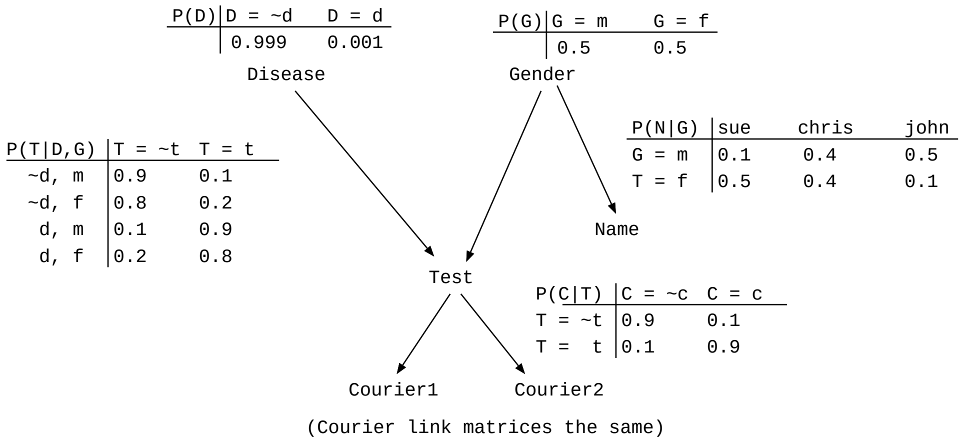
Revised Link Matrix $M_{T|D}$

$P(T\|D)$	$T = \neg t$	$T = t$
$\neg d$	0.85	0.15
d	0.15	0.85

17-84: **Calculating $BEL(D)$**

$$\begin{aligned} \lambda(D) &= \begin{bmatrix} 0.85 & 0.15 \\ 0.15 & 0.85 \end{bmatrix} \begin{bmatrix} 0.05 \\ 0.9 \end{bmatrix} \\ &= \begin{bmatrix} 0.1775 & 0.7725 \end{bmatrix} \\ \pi(D) &= \begin{bmatrix} 0.999 & 0.001 \end{bmatrix} \\ BEL(D) &= \alpha\pi(D)\lambda(D) \\ &= \alpha \begin{bmatrix} 0.177323 & 0.0007725 \end{bmatrix} \\ &= \begin{bmatrix} 0.99566 & 0.00434 \end{bmatrix} \end{aligned}$$

17-85: **Complete Polytree Example**



P(D)	D = ~d	D = d
	0.999	0.001

P(G)	G = m	G = f
	0.5	0.5

P(T|D,G)	T = ~t	T = t
~d, m	0.9	0.1
~d, f	0.8	0.2
d, m	0.1	0.9
d, f	0.2	0.8

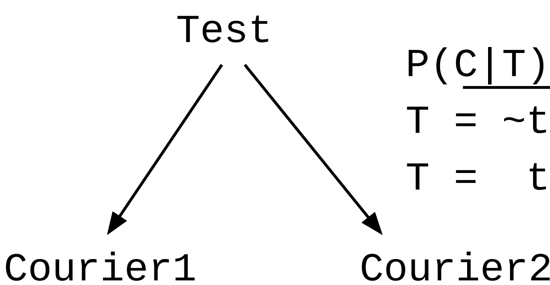
P(N|G)	sue	chris	john
G = m	0.1	0.4	0.5
T = f	0.5	0.4	0.1

P(C|T)	C = ~c	C = c
T = ~t	0.9	0.1
T = t	0.1	0.9

17-86: **Complete Polytree Example**

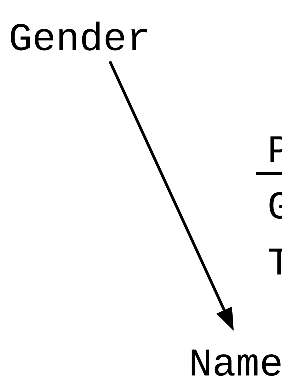
- Find $BEL(D)$, given that:
  - Both couriers return a positive result
  - Patients name is John

17-87: **Polytree Example: λs**



P(C|T)	C = ~c	C = c
T = ~t	0.9	0.1
T = t	0.1	0.9

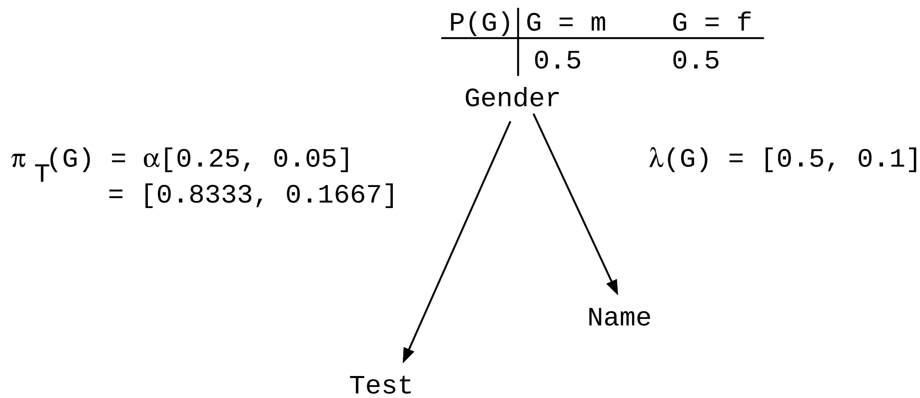
- $\lambda(C_1) = \lambda(C_2) = [0, 1]$
- $\lambda_{C_1}(T) = [0.1, 0.9]$
- $\lambda_{C_2}(T) = [0.1, 0.9]$
- $\lambda(T) = [0.01, 0.81]$



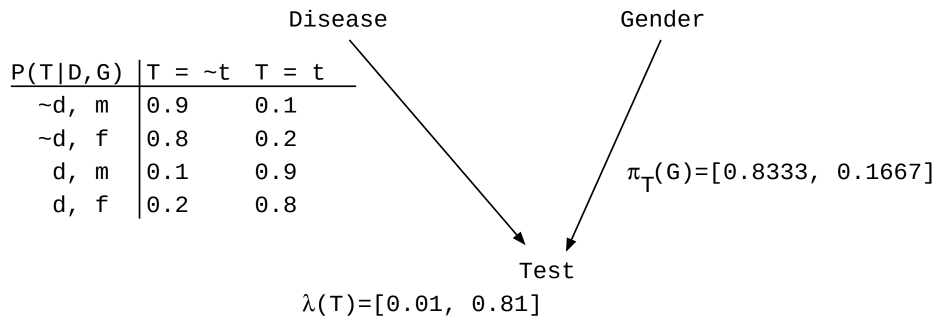
P(N|G)	sue	chris	john
G = m	0.1	0.4	0.5
T = f	0.5	0.4	0.1

17-88: **Polytree Example: λs**

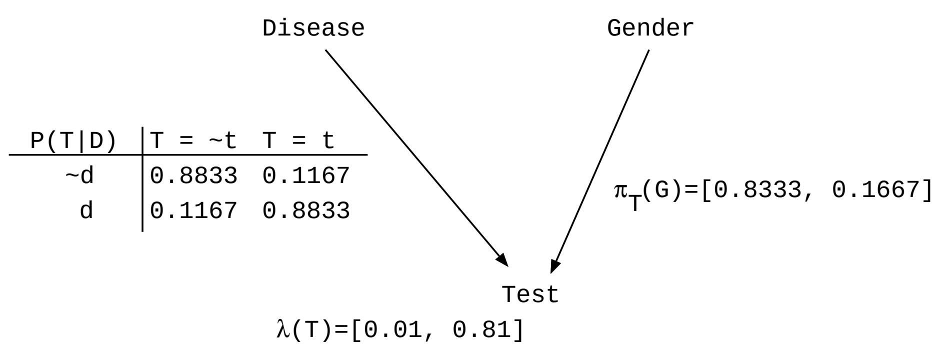
- $\lambda(N) = [0, 0, 1]$
- $\lambda(G) = [0.5, 0.1]$



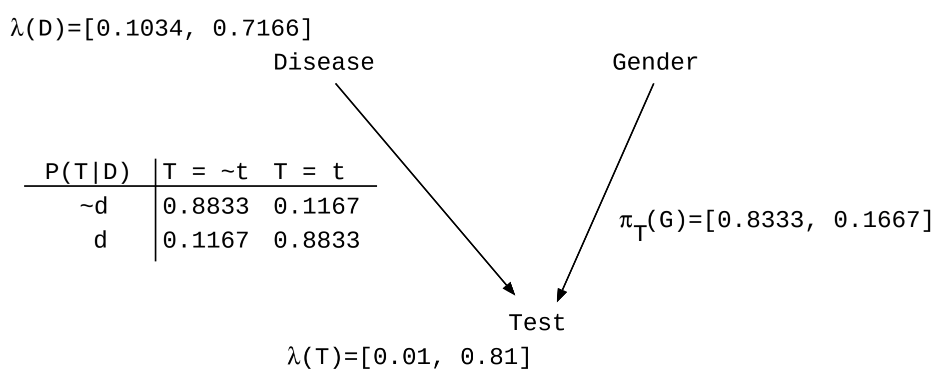
17-89: **Polytree Example: λs**



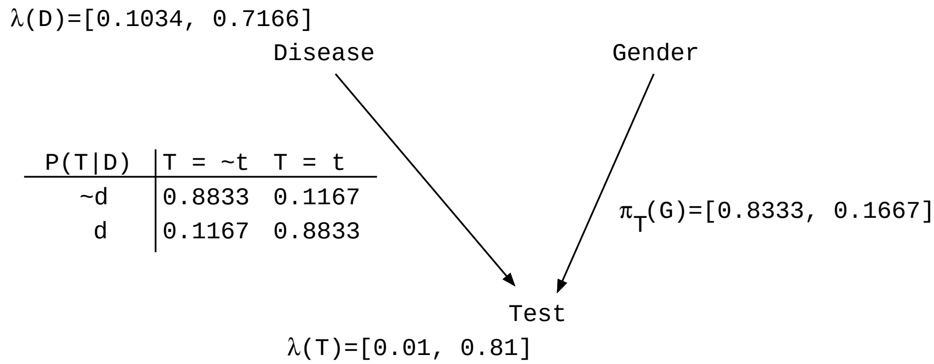
17-90: **Polytree Example: λs**



17-91: **Polytree Example: λs**



17-92: **Polytree Example: λs**



P(T|D)	T = ~t	T = t
~d	0.8833	0.1167
d	0.1167	0.8833

17-93: **Polytree Example: λs**

$$\begin{aligned} BEL(D) &= \alpha\pi(D)\lambda(D) \\ BEL(D) &= \alpha[0.999, 0.001][0.1034, 0.7166] \\ BEL(D) &= \alpha[0.1033, 0.0007] \\ BEL(D) &= \alpha[0.9933, 0.0067] \end{aligned}$$

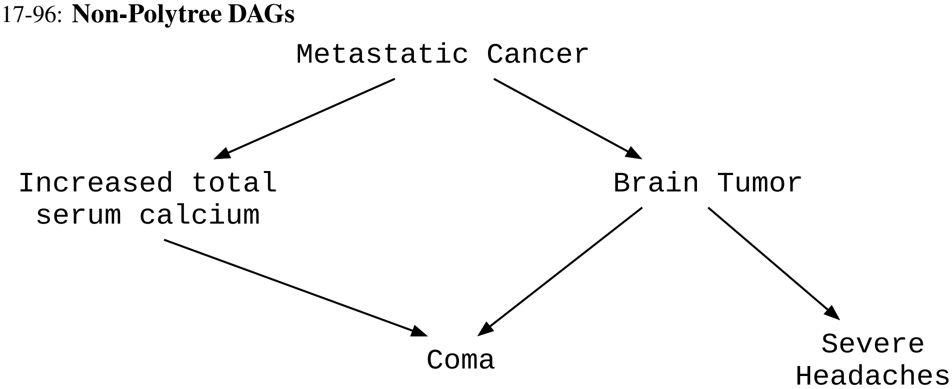
17-94: **Observing Non-Leaves**

- What if we observe a variable that is not a leaf?
  - For instance, we observe the actual test result
- Add a "phantom child"
- Set λ message from that child to $[0, \ldots, 0, 1, 0, \ldots, 0]$, where the 1 occurs at the observed value
- This λ message will override all other evidence for the node

17-95: **Bayesian Network Failures**

- Unfortunately, message passing only works for *polytrees* – DAGs whose underlying undirected graph has no cycles.
- There are systems that we would like to model (including many medical systems) whose Markovian DAG does not form a polytree.
- Message passing system is not guaranteed to produce correct results in non-polytrees.

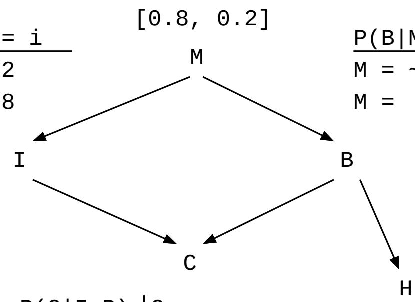
17-96: **Non-Polytree DAGs**



- We can still calculate $P(X_i|PA_i)\ldots$

17-97: **Non-Polytree DAGs**

P(I|M)	I = ~i	I = i
M = ~m	0.8	0.2
M = m	0.2	0.8



P(B|M)	b = ~b	B = b
M = ~m	0.95	0.05
M = m	0.8	0.2

P(H|B)	H = ~h	H = h
B = ~b	0.4	0.6
B = b	0.2	0.8

P(C|I,B)	C = ~c	c = c
~i,~b	0.95	0.05
~i, b	0.2	0.8
i,~b	0.2	0.8
i, b	0.2	0.8

- This is still enough information to answer queries – we just can't use the message passing scheme
  - why?

17-98: **Monte Carlo Method**

- For each root variable, pick a value for the variable according to the prior.
- For example:
  - X is a root variable
  - $\pi(X) = [0.3, 0.2, 0.5]$
  - $\Rightarrow$ Pick the value x_1 for X with probability 0.3, x_2 with probability 0.2, and x_3 with probability 0.5

17-99: **Monte Carlo Method**

- Once a value for all of the parents of a node Z have been chosen, pick a value for the node based on the value of the parents, and $P(Z|PA_Z)$
- For example:
  - If Z has a single parent W
  - $W = [0, 1, 0]$,
  - $P(Z|W) =$

$P(Z|W)$	z_1	z_2	z_3
w_1	0.1	0.2	0.8
w_2	0.3	0.4	0.3
w_3	0.9	0.1	0

  - $\Rightarrow$ Pick z_1 with probability 0.3, z_2 with probability 0.4, and z_3 with probability 0.3.

17-100: **Monte Carlo Method**

- Once values have been chosen for all variables in the network, we have a single trial
- Do repeated trials, collect frequency information, and use that information to determine the values of queries.

17-101: **Monte Carlo Method**

- Once values have been chosen for all variables in the network, we have a single trial
- Do repeated trials, collect frequency information, and use that information to determine the values of queries.
    - To determine $P(x|y)$, count the number of trials in which $X = x$ and $Y = y$,and the number of trials in which $Y = y$, and divide to get an estimate on $P(x|y)$

17-102: **Monte Carlo Method**

- Disadvantages of the Monte Carlo Method:

17-103: **Monte Carlo Method**

- Disadvantages of the Monte Carlo Method:
    - Not guaranteed to find an exact probability in finite time.
    - Can require exponential time to get good results.
    - Calculating $P(x|y)$ when both x and y are unlikely can require a very large number of iterations to get good data.

17-104: **Monte Carlo Method**

- Advantages of the Monte Carlo Method:

17-105: **Monte Carlo Method**

- Advantages of the Monte Carlo Method:
    - Does not require exponential space
    - Do not need to modify the network (no node collapsing)
    - Easy to implement
        - And easy to parallelize
    - Can get approximate answers "quickly", and can get better answers with more time

17-106: **Other Techniques**

- There are a plethora of other techniques for doing inference in non-polytrees
    - Combining nodes to remove cycles
    - Methods using undirected graphs
    - Leave those methods unexplored

17-107: **Applications of Bayesian Networks**

- Diagnosis (widely used in Microsoft's products)
- Medical diagnosis
- Spam filtering
- Expert systems applications (plant control, monitoring)
- Robotic control