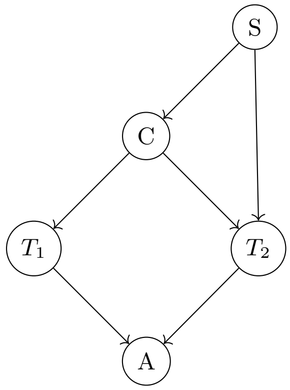


S	θ_S
male	0.55
female	0.45

S	C	$\theta_{c\|s}$
male	yes	0.05
male	no	0.95
female	yes	0.01
female	no	0.99

C	T_1	$\theta_{t_1\|c}$
yes	+ve	0.8
yes	-ve	0.2
no	+ve	0.2
no	-ve	0.8

S	C	T_2	$\theta_{t_2\|c,s}$
male	yes	+ve	0.80
male	yes	-ve	0.20
male	no	+ve	0.20
male	no	-ve	0.80
female	yes	+ve	0.95
female	yes	-ve	0.05
female	no	+ve	0.05
female	no	-ve	0.95

T_1	T_2	A	$\theta_{a\|t_1,t_2}$
+ve	+ve	yes	1
+ve	+ve	no	0
+ve	-ve	yes	0
+ve	-ve	no	1
-ve	+ve	yes	0
-ve	+ve	no	1
-ve	-ve	yes	1
-ve	-ve	no	0

Variables:
C — condition
T's — tests to detect the condition
S — sex of the patient

We first compute the marginal distributions to discuss probabilities.
These come in two major forms:

T_1	0.2192
$\neg T_1$	.7808

Prior Marginal
(pre-testing)

C	0.4533
$\neg C$	.5467

Posterior Marginal
(assuming $T_1 = T_2 = 1$)

We then use this information to find the **Most Probable Explanation**:
We assume the final consequence (A) and search for the most probable query
Setting A moves us from 32 $\rightarrow$ 8 states;
result: {C=no, S=fe, T1=-ve, T2=-ve} = 47%

This is not applicable in every situation; we generalize into the **Maximum a Posteriori Hypothesis**:
this is more complex and less efficient, but more often applicable
we find a subset of variables & find the MPE
(ex. S=C | A $\rightarrow$ {C=no, S=male} $|approx$ 49.3%

When we seek to make inferences, algorithms fall into two major categories:

1. variable elimination
2. conditioning

In both situations, complexity is tied to the topology of the Bayesian Network
the actual property is **tree width** — it is roughly analogous to connectivity
MPH = $O(nd^w)$ given n=#var, d=#val, w=width
We will not define tree width, as it is complex, but we observe a few special cases:
Trees have width 1 (one path from any node to node)
Poly-trees (>1 parent ok) have width = the maximum parent count of any node
These are both singly connected networks; multiply connected leads to a DAG

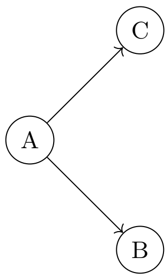
Our first method of performing inference is

Weighted Model Counting

A	B	C	w
t	t	t	0.08
t	t	f	0.04
t	f	t	0.10
t	f	f	0.10
f	t	t	0.20
f	t	f	0.00
f	f	t	0.42
f	f	f	0.06

Consider the statement $\Delta = (A \vee B) \wedge \neg C$
This has 3 variables and suggests 8 worlds
Given an nd-DNNF circuit, we can solve in O(N)!
Consider the example on the right; it is trivial!
WMC = 0.04 + 0.10 + 0.00 = 0.14
We thus only need a compiler which would transform $\Delta \rightarrow$ sd-DNNF
Unfortunately, in the general case this is intractable

There are tractable subsets of this, though:



We can use probability is our weights! $\rightarrow WMC(\Delta \wedge \alpha) = Pr(\alpha)$
So we need to convert $\Delta \rightarrow$boolean circuit.
We can see that Δ holds in $w_1, w_4, \& w_7$
We thus say $W(A) = W(\neg A) = ... = W(\neg C) = 1$
We then let $W(P_i) = \theta_i \& W(\neg P_i) = 0,$ so $W(A) = W(P_1)W(P_4)W(P_7)$

Thus we can solve probabilistic reasoning by symbol manipulations!
Properties of the algorithm:

1. there are many possible representations
2. it is not sensitive to tree width
3. it is not only applicable to Bayesian Networks

We can see that the size(Bayes) = size(CPT); though size(Bayes) = O(nd^{k+1}), size(joint-table) = O(D^n)!
This shows that Bayesian Networks are much more space efficient!

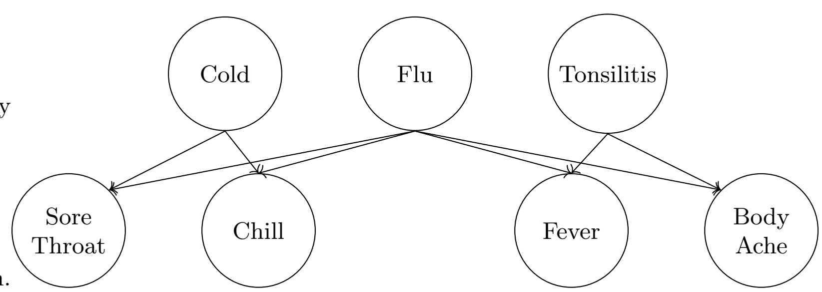
The process of modeling logic as a Bayesian Network has 3 steps:

1. define variables & values
2. define edges
3. specify CPT

Variables will then be labeled either query or evidence variables depending on the query.
This is a common approach used in early spam filters and Google ad Rephil.

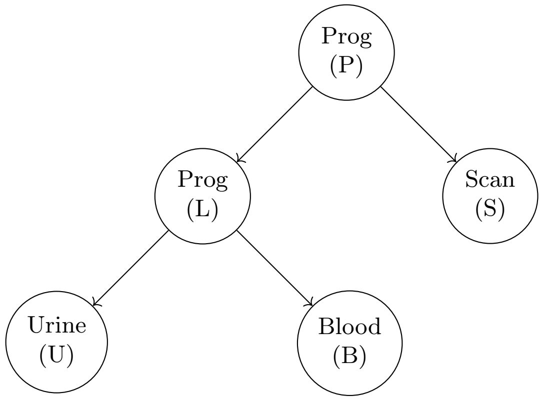
Consider the following statements

- The cold causes a sore throat/chill
- The flu causes a sore throat/chill/fever/body ache
- Tonsillitis can show itself in fever/body ache



This network forms a bipartite graph.
We could have used one multivariable disease node.
We use this to give probabilities of a given condition.

If we have complete information, we can model a system as a Bayesian Network.
For incomplete information, however, we must use Expectation Maximization.
For example, suppose the following environment:



P(P|S) = 0.87
$\mathrm{P}(S|\neg P) = 0.01$ & $\mathrm{P}(\neg S|P) = 0.1$
$\mathrm{P}(U|\neg P) = 0.1$ & $\mathrm{P}(\neg U|P) = 0.3$
$\mathrm{P}(B|\neg P) = 0.1$ & $\mathrm{P}(\neg B|P) = 0.2$

P(p)	0.87
P(¬p)	0.13

S	P	P(S, P)
1	1	0.9
1	0	0.1
0	1	0.01
0	0	0.99

What is the marginal $(P|\neg S, \neg B, \neg U)$? – 10.21%!
WHAT?? why is that so high?
It turns out this is because of the false negative rate of the scanning test.
We want a false negative rate of below 5%.
We can address this in one of three ways:

1. get a better scanning test — a false (-) for S of 4.63% meets our requirement
2. lower the success of the procedure — 75.59% would meet our requirement
3. increase P(L|P) — 99.67% meets our requirement

(b) and (c) turn out to be either impractical or in-economical, so our standard approach is to pay!