BAYESIAN NETWORKS

Introduction

- Last week we talked about using probability theory to represent uncertainty in an agent's knowledge of the world.
- With a full joint probability distribution over all the state variables
  - which we can either measure directly

    $$P(toothache, cavity, \neg catch)$$

    or we can compute from conditionals

    $$P(catch|toothache, cavity)$$

  we can compute any specific values we want.
- Computationally this is awkward.
- Bayesian networks are how we make the computation tractable.

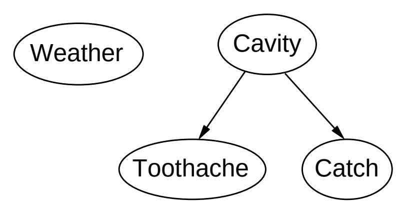
Bayesian networks

- A simple, graphical notation for conditional independence assertions and hence for compact specification of full joint distributions
- Syntax:
  - a set of nodes, one per variable
  - a directed, acyclic graph (link ≈ "directly influences") a conditional distribution for each node given its parents

    $$\mathbf{P}(X_i|Parents(X_i))$$

- In the simplest case, conditional distribution represented as a *conditional probability table* (CPT) giving the distribution over X_i for each combination of parent values

- Topology of network encodes conditional independence assertions:

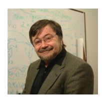


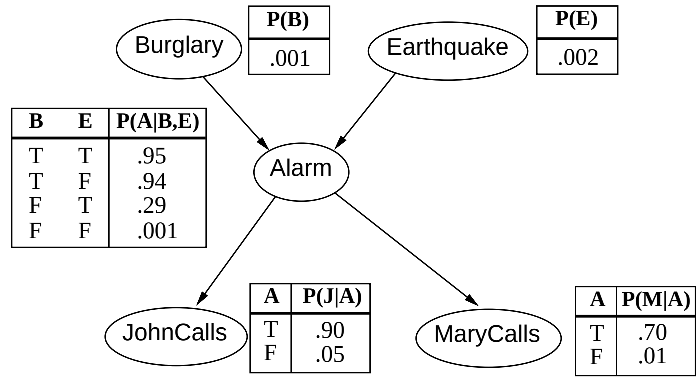
- *Weather* is independent of the other variables
- *Toothache* and *Catch* are conditionally independent given *Cavity*

- An example (from California):

  I'm at work, neighbor John calls to say my alarm is ringing, but neighbor Mary doesn't call. Sometimes it's set off by minor earthquakes. Is there a burglar?

- Variables: *Burglar*, *Earthquake*, *Alarm*, *JohnCalls*, *MaryCalls*
- Network topology reflects "causal" knowledge:
  - A burglar can set the alarm off
  - An earthquake can set the alarm off
  - The alarm can cause Mary to call
  - The alarm can cause John to call





P(B)
.001

P(E)
.002

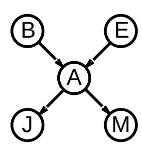
B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J|A)
T	.90
F	.05

A	P(M|A)
T	.70
F	.01

Compactness

- A CPT for Boolean X_i with k Boolean parents has 2^krows for the combinations of parent values
- Each row requires one number p for $X_i = true$ (the number for $X_i = false$ is just $1 - p$)
- If each variable has no more than k parents, the complete network requires $O(n \cdot 2^k)$ numbers
  - grows linearly with n, vs. $O(2^n)$ for the full joint distribution
- For burglary net, $1+1+4+2+2 = 10$ numbers (vs. $2^5 - 1 = 31$)



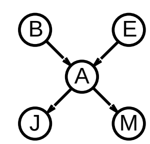
Global semantics

- *Global* semantics defines the full joint distribution as the product of the local conditional distributions:

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i | parents(X_i))$$

- $P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$

  $=$



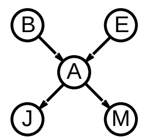
Global semantics

- *Global* semantics defines the full joint distribution as the product of the local conditional distributions:

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i | parents(X_i))$$

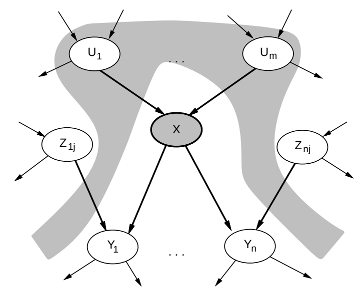
- $P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$

$$= P(j|a)P(m|a)P(a|\neg b, \neg e)P(\neg b)P(\neg e)$$
$$= 0.9 \times 0.7 \times 0.001 \times 0.999 \times 0.998$$
$$\approx 0.00063$$



Local semantics

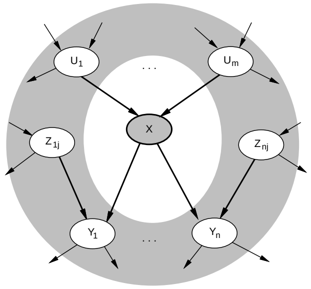
- *Local* semantics: each node is conditionally independent of its nondescendants given its parents



- Theorem: Local semantics $\Leftrightarrow$ global semantics

Markov blanket

- Each node is conditionally independent of all others given its *Markov blanket*: parents + children + children's parents



Andrey Markov

Compact conditional distributions

- CPT grows exponentially with number of parents
  - Use *canonical* distributions that are defined compactly
- *Deterministic* nodes are the simplest case.
- $X = f(Parents(X))$ for some function f
  - Boolean functions:

$$NorthAmerican \Leftrightarrow Canadian \vee US \vee Mexican$$

  - Numerical relationships among continuous variables

$$\frac{\partial Level}{\partial t} = \text{inflow + precipitation - outflow - evaporation}$$

- *Noisy-OR* distributions model multiple noninteracting causes
  1. Parents $U_1 \ldots U_k$ include all causes (can add *leak node*)
  2. Independent failure probability q_i for each cause alone
     $\Rightarrow P(X|U_1 \ldots U_j, \neg U_{j+1} \ldots \neg U_k) = 1 - prod_{i=1}^{j} q_i$

Cold	*Flu*	*Malaria*	$P(Fever)$	$P(\neg Fever)$
F	F	F	**0.0**	1.0
F	F	T	0.9	**0.1**
F	T	F	0.8	**0.2**
F	T	T	0.98	$0.02 = 0.2 \times 0.1$
T	F	F	0.4	**0.6**
T	F	T	0.94	$0.06 = 0.6 \times 0.1$
T	T	F	0.88	$0.12 = 0.6 \times 0.2$
T	T	T	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

- Number of parameters *linear* in number of parents

Inference tasks

- *Simple queries*: compute posterior marginal $\mathbf{P}(X_i|\mathbf{E}=\mathbf{e})$

  $P(NoGas|Gauge=empty, Lights=on, Starts=false)$
- *Conjunctive queries*

  $\mathbf{P}(X_i, X_j|\mathbf{E}=\mathbf{e}) = \mathbf{P}(X_i|\mathbf{E}=\mathbf{e})\mathbf{P}(X_j|X_i, \mathbf{E}=\mathbf{e})$
- *Optimal decisions*: decision networks include utility information; probabilistic inference required for $P(outcome|action, evidence)$
- *Value of information*: which evidence to seek next?
- *Sensitivity analysis*: which probability values are most critical?
- *Explanation*: why do I need a new starter motor?

Inference by enumeration

- Simplest approach to evaluating the network is to do just as we did for the dentist example.
- Difference is that we use the structure of the network to tell us which sets of joint probabilities to use.
  - Thanks Professor Markov
- Gives us a slightly intelligent way to sum out variables from the joint without actually constructing its explicit representation

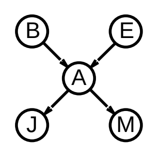
- Simple query on the burglary network.

  $\mathbf{P}(B|j, m)$

  $\mathbf{P}(B, j, m)/P(j, m)$

  $= \alpha\mathbf{P}(B, j, m)$

  $= \alpha\ \Sigma_e\ \Sigma_a\ \mathbf{P}(B, e, a, j, m)$



- Rewrite full joint entries using product of CPT entries:

  $\mathbf{P}(B|j, m)$

  $= \alpha\ \Sigma_e\ \Sigma_a\ \mathbf{P}(B)P(e)\mathbf{P}(a|B, e)P(j|a)P(m|a)$

  $= \alpha\mathbf{P}(B)\ \Sigma_e\ P(e)\ \Sigma_a\ \mathbf{P}(a|B, e)P(j|a)P(m|a)$

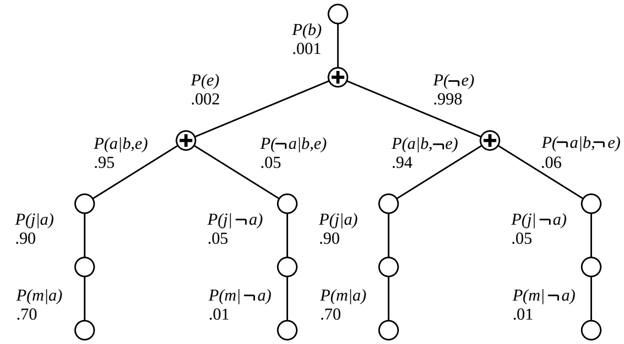
- We evaluate this expression

$$P(B|j,m) = \alpha \mathbf{P}(B) \sum_e P(e) \sum_a \mathbf{P}(a|B,e)P(j|a)P(m|a)$$

by going through the variables in order, multiplying CPT entries along the way.

- At each point, we need to loop through the possible values of the variable.
- Involves a lot of repeated calculations.

Evaluation tree



- Enumeration is inefficient: repeated computation
  - Computes $P(j|a)P(m|a)$ for each value of e

Enumeration algorithm

function ENUMERATION-ASK(X, **e**, bn) **returns** a distribution over X
  inputs: X, the query variable
    e, observed values for variables $\exists$
    bn, a Bayesian network with variables $\{X\} \cup \exists \cup \mathbf{Y}$
  $\mathbf{Q}(X) \leftarrow$ a distribution over X, initially empty
  for each value x_i of X **do**
    extend **e** with value x_i for X
    $\mathbf{Q}(x_i) \leftarrow$ ENUMERATE-ALL(VARS[bn], **e**)
  return NORMALIZE($\mathbf{Q}(X)$)

function ENUMERATE-ALL($vars$, **e**) **returns** a real number
  if EMPTY?($vars$) **then return** 1.0
  $Y \leftarrow$ FIRST($vars$)
  if Y has value y in **e**
    then return $P(y \mid Pa(Y)) \times$ ENUMERATE-ALL(REST($vars$), **e**)
    else return $\Sigma_y\ P(y \mid Pa(Y)) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}_y$)
      where $\mathbf{e}_y$ is **e** extended with $Y = y$

Other exact approaches

- We can improve on enumeration.
- *Variable elimination* evaluates the enumeration tree bottom up, remembering intermediate values.
  - Simple and efficient for single queries
- *Clustering algorithms* can be more efficient for multiple queries
- However, all exact inference can be computationally intractable.

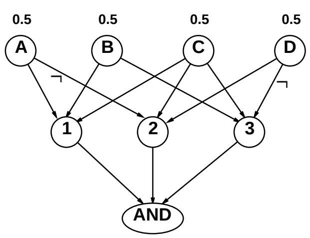
Complexity of exact inference

- *Singly connected* networks (or *polytrees*)
  - any two nodes are connected by at most one (undirected) path
  - time and space cost of variable elimination are $O(d^k n)$
    k parents, d values.

Complexity of exact inference

- *Multiply connected* networks:
  - can reduce 3SAT to exact inference $\Rightarrow$ NP-hard
  - equivalent to *counting* 3SAT models $\Rightarrow$ #P-complete

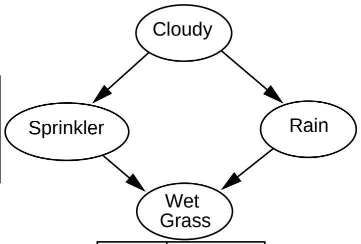
1. A v B v C
2. C v D v ¬A
3. B v C v ¬D



Inference by stochastic simulation

- Basic idea:
  1. Draw N samples from a sampling distribution S
  2. Compute an approximate posterior probability $\hat{P}$
  3. Show this converges to the true probability P

P(C)
.50

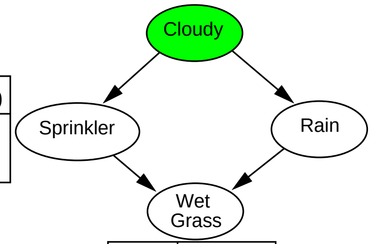


C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

P(C)
.50

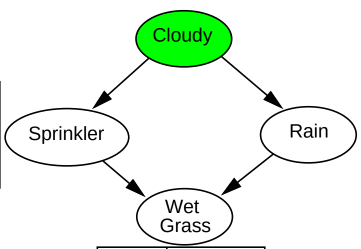


C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

P(C)
.50

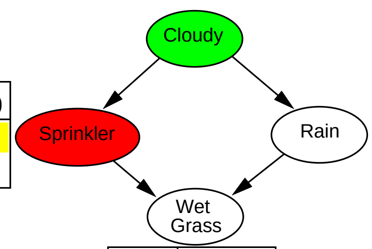


C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

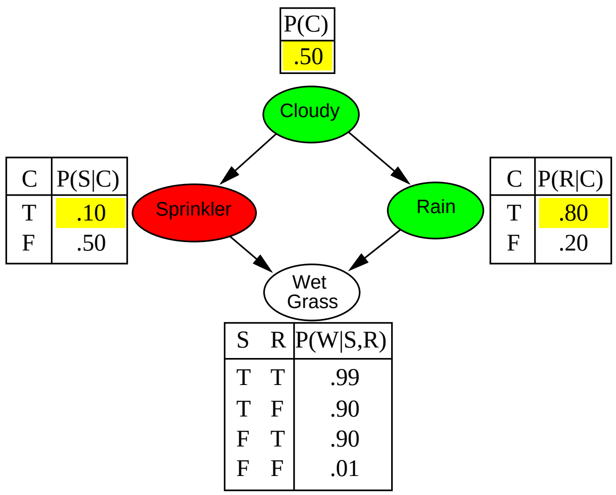
P(C)
.50



C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

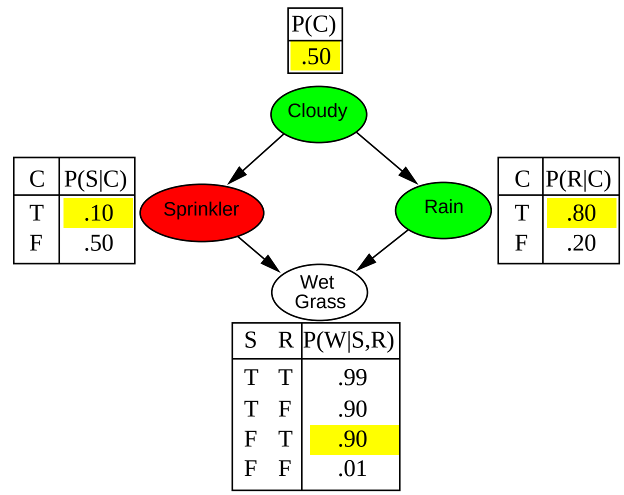


P(C)
.50

C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

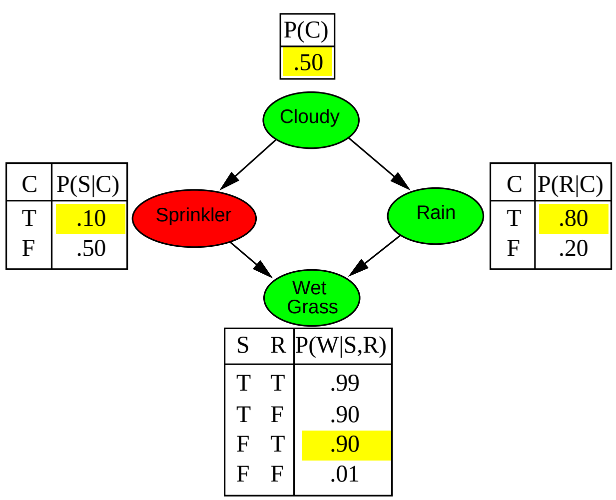


P(C)
.50

C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01



P(C)
.50

C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

- So, this time we get the event

$$[true, false, true, true]$$

- If we repeat the process many times, we can count the number of times $[true, false, true, true]$ is the result.
- The proportion of this to the total number of runs is:

$$\mathbf{P}(c, \neg s, r, w)$$

- The more runs, the more accurate the probability.

- This algorithm:

function PRIOR-SAMPLE(*bn*) **returns** an event sampled from *bn*
  inputs: *bn*, a belief network specifying joint distribution $\mathbf{P}(X_1, \ldots, X_n)$

  $\mathbf{x} \leftarrow$ an event with n elements
  for $i = 1$ **to** n **do**
    $x_i \leftarrow$ a random sample from $\mathbf{P}(X_i \mid parents(X_i))$
      given the values of $Parents(X_i)$ in $\mathbf{x}$
  return $\mathbf{x}$

captures the *no evidence* case, which is what we just looked at.

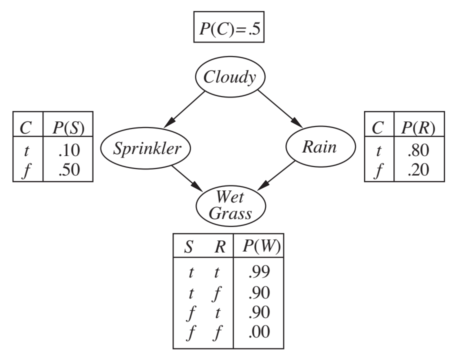
- To get values with evidence, we need conditional probabilities

$$\mathbf{P}(X|\mathbf{e})$$

- Could just compute the joint probability and sum out the conditionals but that is inefficient.
- Better is to use *rejection sampling*
  - Sample from the network but reject samples that don't match the evidence.
  - If we want $\mathbf{P}(w|c)$ and our sample picks $\neg c$, we stop that run immediately.
  - For unlikely events, may have to wait a long time to get enough matching samples.
- Still inefficient.

- Likelihood weighting:
  - Version of *importance sampling*.
  - Fix evidence variable to *true*, so just sample relevant events.
  - Have to weight them with the likelihood that they fit the evidence.
  - Use the probabilities we know to weight the samples.

- Consider we have the following network:



- Say we want to establish $\mathbf{P}(Rain|Cloudy = true, WetGrass = true)$

- We want $\mathbf{P}(Rain|Cloudy = true, WetGrass = true)$
- We pick a variable ordering, say *Cloudy*, *Sprinkler*, *Rain*, *WetGrass*.
- Set the weight to 1 and generate an event.
- *Cloudy* is true, so:

$$w \leftarrow w \times P(Cloudy = true) = 0.5$$

- *Sprinkler* is not an evidence variable, so sample from

$$\mathbf{P}(Sprinkler|Cloudy = true) = \langle 0.1, 0.9 \rangle$$

  Let's assume this returns *false*.

- *Rain* is not an evidence variable, so sample from

$$\mathbf{P}(Rain|Cloudy = true) = \langle 0.8, 0.2 \rangle$$

  Let's assume this returns *true*.
- *WetGrass* is an evidence variable with value *true*, so we set:

$$w \leftarrow w \times P(WetGrass = true|Sprinkler = false, Rain = true) = 0.45$$

- So we end with the event $[true, false, true, true]$ and weight 0.45.
- To find a probability we tally up all the relevant events, weighted with their weights.
- The one we just calculated would tallyup under $Rain = true$

From probability to decision making

- What we have covered allows us to compute probabilities of interesting events.
- But *beliefs* alone are not so interesting to us.
- In the WW don't care so much if there is a pit in (2, 2), so much as we care whether we should go left or right.
- This is complicated because the world is uncertain.
  - Don't know the outcome of actions.
  - Non-deterministic as well as partially observable

DA MAYOR: Mookie.

MOOKIE: Gotta go.

DA MAYOR: C'mere, Doctor.

DA MAYOR: Doctor, this is Da Mayor talkin'.

MOOKIE: OK. OK.

DA MAYOR: Doctor, always try to do the right thing.

MOOKIE: That's it?

DA MAYOR: That's it.

MOOKIE: I got it.

(Spike Lee, Do the Right Thing)

- I offer you the chance to take part in this gamble:
  - \$0 one time in one hundred;
  - \$1 89 times in one hundred;
  - \$5 10 times in one hundred.
- Would you prefer this to \$1.00?

- I offer you the chance to take part in this gamble:
  - \$0 one time in one hundred;
  - \$1 89 times in one hundred;
  - \$5 10 times in one hundred.
- Would you prefer this to \$1.50?

- I offer you the chance to take part in this gamble:
  - \$0 one time in one hundred;
  - \$1 89 times in one hundred;
  - \$5 10 times in one hundred.
- Would you prefer this to \$1.20?

- I offer you the chance to take part in this gamble:
  - \$0 one time in one hundred;
  - \$1 89 times in one hundred;
  - \$5 10 times in one hundred.
- Would you prefer this to \$1.40?

- We can't make this choice without thinking about how likely outcomes are.
- Although the first option is attractive, it isn't necessarily the best course of action (especially if the choice is iterated).
- Decision theory gives us a way of analysing this kind of situation.

- Consider being offered a bet in which you pay \$2 if an odd number is rolled on a die, and win \$3 if an even number appears.
- To analyse this prospect we need a *random variable* X, as the function:

$$X : \Omega \mapsto \Re$$

from the sample space to the values of the outcomes. Thus for $\omega \in \Omega$:

$$X(\omega) = \begin{cases} 3, & \text{if } \omega = 2, 4, 6 \\ -2, & \text{if } \omega = 1, 3, 5 \end{cases}$$

- The probability that X takes the value 3 is:

$$\begin{aligned} \Pr(\{2,4,6\}) &= \Pr(\{2\}) + \Pr(\{4\}) + \Pr(\{6\}) \\ &= 0.5 \end{aligned}$$

- How do we analyse how much this bet is worth to us?

- To do this, we need to calculate the *expected value* of X.
- This is defined by:

$$E(X) = \sum_k k \Pr(X = k)$$

where the summation is over all values of k for which $\Pr(X = k) \neq 0$.

- Here the expected value is:

$$E(X) = 0.5 \times 3 + 0.5 \times -2$$

- Thus the expected value of X is \$0.5, and we take this to be the value of the bet.
  - Not the value you will get.

- What is the expected value of this event:
  - \$0 one time in one hundred;
  - \$1 89 times in one hundred;
  - \$5 10 times in one hundred.
- Would you prefer this to \$1?

- And now we can make a first stab at defining what rational action is.
- Rational action is the choice of actions with the greatest expected value for the agent in question.
- The problem is then to decide what "value" is.

Decision theory

- One obvious way to define "value" is in terms of money.
- This has obvious applications in writing programs to trade stocks, or programs to play poker.
- The problem is that the value of a given amount of money to an individual is highly subjective.
- In addition, using monetary values does not take into account an individual's attitude to risk.

- As an example, consider a transaction which offered the following payoffs:
  - $0 one time in one hundred;
  - $1 million 89 times in one hundred;
  - $5 million 10 times in one hundred.
- Would you prefer this to a guaranteed $1 million?

- Utilities are a means of solving the problems with monetary values.
- Utilities are built up from preferences, and preferences are captured by a preference relation $\preceq$ which satisfies:

$$a \preceq b \quad \text{or} \quad b \preceq a$$
$$a \preceq b \ \text{ and } \ b \preceq c \Rightarrow a \preceq c$$

- You have to be able to state a preference.
- Preferences are transitive.

- A function:

$$u : \Omega \mapsto \Re$$

is a utility function representing a preference relation $\preceq$ if and only if:

$$u(a) \leq u(b) \;\leftrightarrow\; a \preceq b$$

- With additional assumptions on the preference relation (to do with preferences between lotteries) Von Neumann and Morgenstern identified a sub-class of utility functions.

- These "Von Neumann and Morgenstern utility functions"are such that calculating expected utility, and choosing the action with the maximum expected utility is the "best" choice according to the preference relation.
- This is "best" in the sense that any other choice would disagree with the preference order.
- This is why the *maximum expected utility* decision criterion is said to be rational.

- To relate this back to the problem of an agent making a rational choice, consider an agent with a set of possible actions A available to it.
- Each $a \in A$ has a sample space Ω_a associated with it, and a set of possible outcomes s_a where $s_a \subseteq S_a$ and $S_a = 2^{\Omega_a}$.
- (This is a simplification since each s_a will usually be conditional on the state of the environment the agent is in.)

- The action a^* which a rational agent should choose is that which maximises the agent's utility.
- In other words the agent should pick:

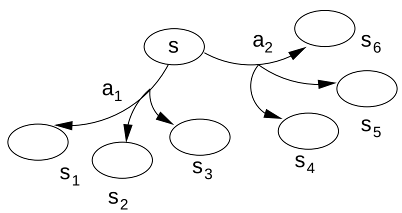
$$a^* = arg \max_{a \in A} u(s_a)$$

- The problem is that in any realistic situation, we don't know which s_a will result from a given a, so we don't know the utility of a given action.
- Instead we have to calculate the expected utility of each action and make the choice on the basis of that.

- In other words, for the set of outcomes s_a of each action each a, the agent should calculate:

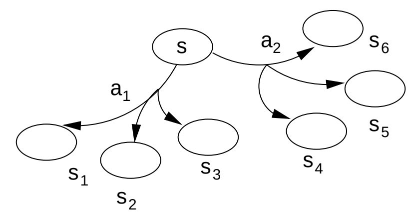
$$E(u(s_a)) = \sum_{s' \in s_a} u(s'). \Pr(s_a = s')$$

and pick the best.



- Thus to be rational, an agent needs to choose a^* such that:

$$a^* = arg \max_{a \in A} \sum_{s' \in s_a} u(s'). \Pr(s_a = s')$$



- As an example, consider an agent which has to choose between tossing a coin, rolling a die, or receiving a payoff of $ 1.
- If the coin is chosen, then the agent gets $1.50 a head and $0.5 for a tail.
- If the die is chosen, the agent gets $5 if a six is rolled, $1 if a two or three is rolled, and nothing otherwise.
- What is the rational choice, assuming that the agent's preferences are (for once) modelled by monetary value?

- Well, we need to calculate the expected outcome of each choice.
- For doing nothing, we have a_1 = "receive payoff", s_{a_1} = {"get $1"}, $u(\text{"get \$1}) = 1$ and $\Pr(s_{a_1} = \text{"get \$1}) = 1$.
- Thus:

$$E(u(s_{a_1})) = 1$$

- If the coin is chosen, we have a_2 = "coin", $s_{a_2} = \{\text{head}, \text{tail}\}$,

$$\begin{aligned} u(\text{head}) &= \$1.50 \\ u(\text{tail}) &= \$0.5 \end{aligned}$$

and

$$\begin{aligned} \Pr(s_{a_2} &= \text{head}) = 0.5 \\ \Pr(s_{a_2} &= \text{tail}) = 0.5 \end{aligned}$$

- Thus the expected utility is:

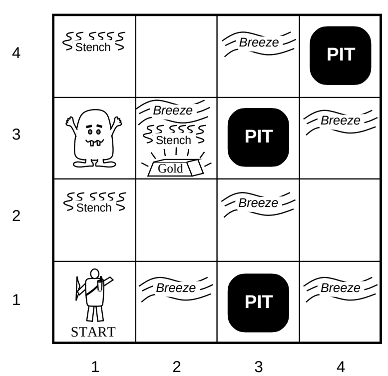
$$\begin{aligned} E(u(s_{a_2})) &= 0.5 \times 1.5 + 0.5 \times 0.5 \\ &= 1 \end{aligned}$$

- Action a_3, rolling the die, can be analysed in a similar way, giving:

$$E(u(s_{a_3})) = 1.17$$

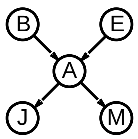
- Choosing to roll the die is the rational choice.

Decisions in the WW



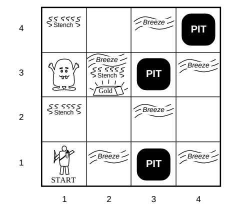
- Actions have a range of outcomes.
- Forward has some probability of moving sideways
  - Not so silly with a robot
- Probabilities across action outcomes.
  - Given an action, probability of getting to some states
- Utilities for states.

- Given what we know about Bayesian networks, we can clearly deal with complex situations as far as probability is concerned.



- Should I go home given that John calls and Mary doesn't?

- But what about more complex decisions?



- What is the best sequence of actions to carry out to get the gold?
- Next time.

Summary

- This lecture started with probabilistic inference.
  - Inference by enumeration
  - Inference by stochastic simulation
- Then we went on to talk about utilities.
- We now know how to make a decision about the best action to carry out.
  - But we can only choose one action at a time.
- Next time we'll look at sequential decision problems.