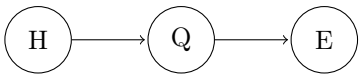
1 Inference Conceptual Review

1. When would we want to use inference?

   Inference is used to calculate some useful quantity from a joint probability distribution. For example, we can use it in speech recognition (the example given in class). Here, we could answer a question such as what is the most probable next word given the audio for the next word and the fact that the first word is "artificial". In this case, inference tells us which words were most likely to be said in the audio clip.

2. Suppose we are given binary random variables Q, H, E (query, hidden, evidence).
   We want to query $P(q \mid e)$.



   (a) **Enumeration**
   Perform inference on a joint distribution. Use the Bayes net above to break down joint into CPT factors.

   Note: You may use a proportionality constant α in your answer.

   Inference on a joint distribution:
   $P(q \mid e) = \alpha P(q, e)$

   $= \alpha \sum_{h \in (h_1, h_2)} P(q, h, e)$

   Using Bayes net to break down joint in to CPT factors:
   $P(q \mid e) = \alpha \sum_{h \in (h_1, h_2)} P(h) P(q \mid h) P(e \mid q)$

   $= \alpha [P(h_1) P(q \mid h_1) P(e \mid q) + P(h_2) P(q \mid h_2) P(e \mid q)]$

   (b) **Variable Elimination**
   Rewrite your answer to enumeration by moving summations inwards as far as possible.

   Note: You may use a proportionality constant α in your answer.

   $P(q \mid e) = \alpha P(e \mid q) \sum_{h \in (h_1, h_2)} P(h) P(q \mid h)$

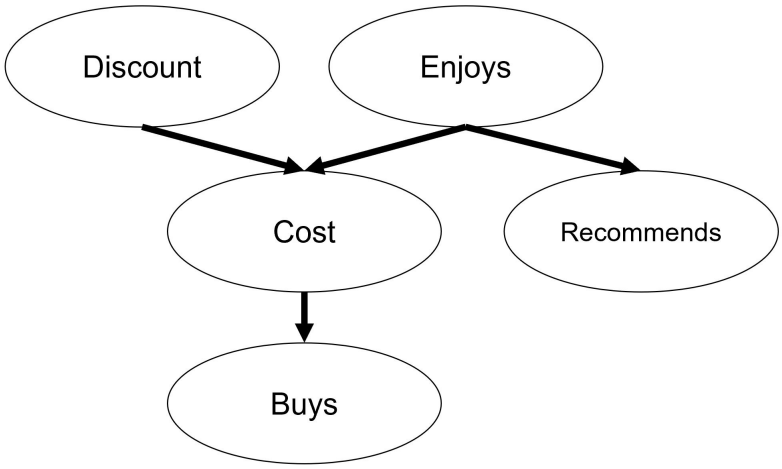
   $= \alpha P(e \mid q) [P(h_1) P(q \mid h_1) + P(h_2) P(q \mid h_2)]$

   (c) Based on 2a and 2b, why is variable elimination more efficient than enumeration?
   We sum entries from the joint distribution (these entries are obtained from a Bayes Net by multiplying conditional probabilities) in order to do inference by enumeration. This involves multiple repeated sub-expressions. We can move the summations inwards as far as possible in order to eliminate repeated computation. This is called variable elimination and is thus more efficient.

2 Inference

Realizing that students aren't particularly fond of reading the textbook, the 281 course staff have developed a software that automatically scans the textbook and outputs key points for each individual chapter. However, since the development of the software requires time and computational resources, the 281 staff decides to offer a free one month trial to students, after which a paid subscription is necessary to keep using the software. The following network and variables are used to represent the problem:

- $Discount(D)$: $+d$ if a discount is offered, $-d$ otherwise
- $Enjoys(E)$: $+e$ if a student enjoys the software, $-e$ otherwise
- $Cost(C)$: $+c$ if the software cost is < 20, $-c$ otherwise
- $Recommends(R)$: $+s$ if the student recommends the software to a friend, $-s$ otherwise
- $Buys(B)$: $+b$ if the student buys a software subscription, $-b$ otherwise



D	$P(D)$
$+d$	0.7
$-d$	0.3

E	$P(E)$
$+e$	0.1
$-e$	0.9

C	D	E	$P(C \mid D, E)$
$+c$	$+d$	$+e$	0.2
$-c$	$+d$	$+e$	0.8
$+c$	$+d$	$-e$	0.3
$-c$	$+d$	$-e$	0.7
$+c$	$-d$	$+e$	0.9
$-c$	$-d$	$+e$	0.1
$+c$	$-d$	$-e$	0.5
$-c$	$-d$	$-e$	0.5

R	E	$P(R \mid E)$
$+r$	$+e$	0.8
$-r$	$+e$	0.2
$+r$	$-e$	0.5
$-r$	$-e$	0.5

B	C	$P(B \mid C)$
$+b$	$+c$	0.7
$-b$	$+c$	0.3
$+b$	$-c$	0.5
$-b$	$-c$	0.5

1. How can we represent the probability that a student buys and recommends the software using the conditional probabilities at each node?
   $P(+b, +r) = \sum_{c,d,e} P(+b|c)P(c|d,e)P(d)P(e)P(+r|e)$

   This sum is equivalent to summing out the hidden variables in the joint distribution: $\sum_{c,d,e} P(d, e, c, +r, +b)$.

2. The staff has surveyed students and collected data on whether the students enjoyed the software or not. With this information, we want to perform inference on a joint distribution where the query variable is *Buys* (B).

   (a) How can we represent the probability expression in terms of conditional probabilities from the network?
   $P(B|E) = \alpha P(B, E) = \alpha \sum_{d,c,r} P(B, E, d, c, r) = P(E)\alpha \sum_{d,c,r} P(d)P(c|d, E)P(B|c)P(r|E)$

   Note: Equation 13.9 on page 493 of the TB goes into detail about why we use α. In short, when we are calculating conditional probabilities, α acts as a normalization constant. However, we can proceed with calculating the conditional probabilities even without knowing the value of α because relative proportions remain the same without normalization (e.g. relative proportions of $P(+b|E)$ and $P(-b|E)$ remain the same without knowing the exact value of $\alpha = 1/P(E)$).

   (b) What are the hidden and evidence variable(s)?
   The hidden variables are D, C, R, and the evidence variable is E.

3. Using the probability expression from the previous part, we want to compute the query B given evidence that the student enjoys the software. Assume the variable ordering is in alphabetical order.

   (a) How many factors are there, and what are the dimensions of each factor?
   Our expression is: $P(B| + e) = \alpha P(+e) \sum_{r,d,c} P(d)P(c|d, +e)P(B|c)P(r| + e)$

   $= \alpha P(+e) \sum_r P(r| + e) \sum_d P(d) \sum_c P(c|d, +e)(B|c)$

   Each conditional probability corresponds to an individual factor, so there are 5 factors total. The factor for $P(d)$ and $P(r| + e)$ each have dimension 2×1, the factors for $P(c|d, +e)$ and $P(B|c)$ each have dimension 2×2, and the factor for $P(+e)$ is a one-element vector.

   (b) Run the variable elimination algorithm to eliminate repeated computations for the expression $P(B| + e)$, and fill in the factor table as necessary for each variable eliminated.

   Eliminating C:

D	B	$f_1(D, B)$
$+d$	$+b$	
$-d$	$+b$	
$+d$	$-b$	
$-d$	$-b$	

   Eliminating D:

B	$f_2(B)$
$+b$	
$-b$	

   All factors: $P(D), P(+e), P(C|D, +e), P(B|C), P(R| + e)$

   - Choose C: The relevant factors are $P(C|D, +e), P(B|C)$. We sum out C to get $f_1(D, B) = \sum_c P(C = c|D, +e)P(B|C = c)$.

     Expression: $P(B| + e) = \alpha P(+e) \sum_{d,r} P(D = d)P(R = r| + e) \times f_1(D, B)$
   - Choose D: We sum out the relevant factors $P(D)$ and $f_1(D, B)$ to get $f_2(B) = \sum_d f_1(D = d, B)P(d)$.

     Expression: $P(B| + e) = \alpha P(+e) \sum_r P(R = r| + e) \times f_2(B)$
   - Choose R: We sum out the relevant factor $P(R| + e)$ to get $\sum_r P(R = r| + e) = 1$. We discard this variable since it is irrelevant and no other factors depend on it.

Expression: $P(B|+e) = \alpha P(+e) \times f_2(B)$

Eliminating C:

D	B	$f_1(D, B)$
$+d$	$+b$	0.54
$-d$	$+b$	0.68
$+d$	$-b$	0.46
$-d$	$-b$	0.32

Eliminating D:

B	$f_2(B)$
$+b$	0.582
$-b$	0.418

(c) How does the resulting expression change if the variable ordering is instead in reverse alphabetical order? Similarly, fill in the factor table as necessary for each variable eliminated.

Eliminating D:

C	$f_2(C)$
$+c$	
$-c$	

Eliminating C:

B	$f_3(B)$
$+b$	
$-b$	

- Choose R: We sum out relevant factor $P(R|+e)$ to get $\sum_r P(R=r|+e)=1$. We can discard this variable since it is irrelevant.
- Choose D: We sum out relevant factors $P(D)$, $P(C|D,+e)$ to get $f_2(C)=\sum_d P(C|D=d,+e)\times P(D=d)$.

  Expression: $P(B|+e)=\alpha P(+e)\sum_c P(B|c)\times f_2(C=c)$
- Choose C: We sum out relevant factors $P(B|c)$ and $f_2(C)$ to get $f_3(B)=\sum_c P(B|C=c)\times f_2(C=c)$.

  $P(B|+e)=\alpha P(+e)f_3(B)$

Eliminating D:

C	$f_2(C)$
$+c$	0.41
$-c$	0.59

Eliminating C:

B	$f_3(B)$
$+b$	0.582
$-b$	0.418

(d) How do the two orderings compare with respect to time and space complexity?

When the terms were ordered in alphabetical order, the largest factor had 2 variables. When the terms were ordered in reverse alphabetical order, the largest factor had 1 variable. Since the size of the largest factor determines the space/time complexity, the second ordering performs better.

(e) Describe a heuristic that could be useful in determining a variable ordering to minimize the size of the largest factor.

Potential ideas:

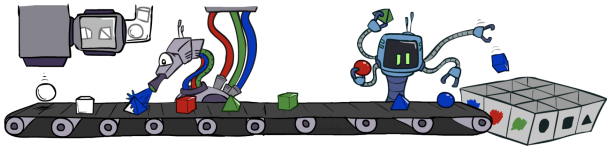
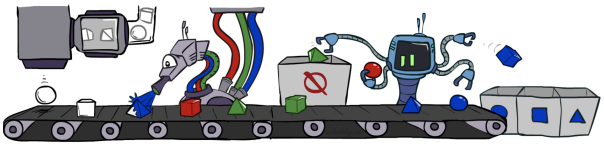
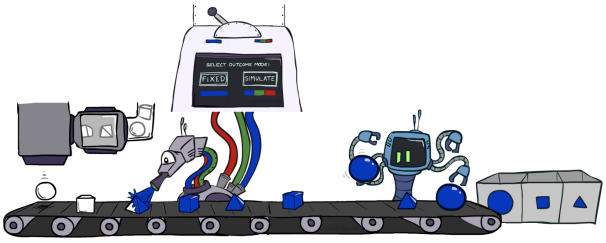
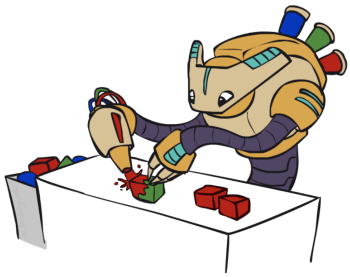
- Eliminate whichever variable minimizes the size of the next factor to be constructed.
- Eliminate the variable with the fewest dependent variables

3 Candy Sampling

1. In the year 2020, the Oompa Loompas at Charlie's Chocolate Factory have decided that they want to try a new automated way of sampling their candies for quality assurance. However, they have spilled chocolate sauce on their only copy of the user manual! Help out the Oompa Loompas by filling in the blanks below with the names of the four different types of sampling methods we've discussed in lecture, and then match each one to the corresponding image, probability distribution, and algorithm from the tables below.

Sampling Method Name	Image and Distribution (A-D)	Algorithm (1-4):

Sampling Method Name	Image and Distribution (A-D)	Algorithm (1-4):
Prior Sampling	A	2
Rejection Sampling	B	4
Likelihood Weighting	C	1
Gibbs Sampling	D	3

Name:	Images and Corresponding Probability Distributions:
(A)	 **P(Q, E)**
(B)	 **P(Q | e)**
(C)	 **P(Q, e)**
(D)	 **P(Q | e)**

Name:	Algorithms:

(1)	`def _____(evidence instantiation):`
	`  w = 1.0`
	`  for i = 1,2,...n:`
	`  if(X_i is an evidence variable):`
	`  X_i = observation x_i for X_i`
	`  w = w * P(x_i | Parents(X_i))`
	`  else:`
	`  Sample x_i from P(X_i | Parents(X_i))`
	`  return ((x_1, x_2,...x_n), w)`

(2)	**function** _____(bn) **returns** an event sampled from the prior specified by bn **inputs**: bn, a Bayesian network specifying joint distribution $\mathbf{P}(X_1, \ldots, X_n)$ $\mathbf{x} \leftarrow$ an event with n elements **foreach** variable X_i **in** $X_1, \ldots, X_n$ **do** $\mathbf{x}[i] \leftarrow$ a random sample from $\mathbf{P}(X_i \mid parents(X_i))$ **return x**

(3)	**function** _____(X, **e**, bn, N) **returns** an estimate of $\mathbf{P}(X\|\mathbf{e})$
	local variables: **N**, a vector of counts for each value of X, initially zero
	Z, the nonevidence variables in bn
	x, the current state of the network, initially copied from **e**
	
	initialize **x** with random values for the variables in **Z**
	for j = 1 to N **do**
	for each Z_i in **Z do**
	set the value of Z_i in **x** by sampling from $\mathbf{P}(Z_i\|mb(Z_i))$
	$\mathbf{N}[x] \leftarrow \mathbf{N}[x] + 1$ where x is the value of X in **x**
	return NORMALIZE(**N**)

(4)	**function** _____(X, **e**, bn, N) **returns** an estimate of $\mathbf{P}(X\|\mathbf{e})$
	inputs: X, the query variable
	e, observed values for variables **E**
	bn, a Bayesian network
	N, the total number of samples to be generated
	local variables: **N**, a vector of counts for each value of X, initially zero
	
	for j = 1 to N **do**
	$\mathbf{x} \leftarrow$ PRIOR-SAMPLE(bn)
	if x is consistent with **e then**
	$\mathbf{N}[x] \leftarrow \mathbf{N}[x]$+1 where x is the value of X in **x**
	return NORMALIZE(**N**)

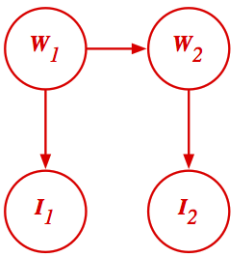
4 Sampling Practice

1. Compared to other sampling methods (rejection, likelihood weighting, Gibbs), what kind of information can prior sampling not use (that other methods can)?
   Other methods can compute probabilities with fixed evidence, while the network for prior sampling has no evidence associated.

2. How does rejection sampling work on a high level, and what is its biggest/immediate weakness?
   It generates samples from the given prior distribution, rejects all samples that do not match the evidence, and then derives the probability (# times the desired value appears in the remaining samples).

   Its biggest weakness is potential inefficiency when evidence is rare. Most samples would then be rejected, so all this information would be thrown away despite being calculated.

The diagram below describes a person's ice-cream eating habits based on the weather. The nodes W_i stand for the weather on a day i, which can either be **s** (sunny) or **r** (rainy). The nodes I_i represent whether the person ate ice-cream on day i, which can either be **t** (true) or **f** (false).



W_1	$P(W_1)$
s	0.6
r	0.4

I	W	$P(I\|W)$
t	s	0.9
f	s	0.1
t	r	0.2
f	r	0.8

W_2	W_1	$P(W_2\|W_1)$
s	s	0.7
r	s	0.3
s	r	0.5
r	r	0.5

Assume we generate the following six samples given the evidence $I_1 = t$ and $I_2 = f$ using **Likelihood Weighted Sampling**:

$$(W_1, I_1, W_2, I_2) = < s, t, r, f >, < r, t, r, f >, < s, t, r, f >, < s, t, s, f >, < s, t, s, f >, < r, t, s, f >$$

Using these samples, we will complete the following table:

(W_1, I_1, W_2, I_2)	Count/N	w	Joint
s, t, s, f	2/6	0.09	0.03
s, t, r, f	/6		
r, t, s, f	/6		
r, t, r, f	/6		

1. What is the weight of the sample (s, t, r, f) above? Recall that the weight given to a sample in likelihood weighting is:

$$w = \prod_{\text{Evidence variables } e} P(e|\text{Parents}(e)).$$

   In this case, the evidence is $I_1 = t, I_2 = f$. The weight of the first sample is therefore

   $w = Pr(I_1 = t|W_1 = s) \cdot Pr(I_2 = f|W_2 = r) = 0.9 \cdot 0.8 = 0.72$

2. What is the estimate of P(s, t, r, f) given the samples?
   The estimate of the joint probability is simply $Count/N * w = 2/6 * 0.72 = 0.24$.

3. Compute the rest of the entries in the table. Use the estimated joint probabilities to estimate $P(W_2 = r|I_1 = t, I_2 = f)$.

(W_1, I_1, W_2, I_2)	Count/N	w	Joint
s, t, s, f	2/6	0.09	0.03
s, t, r, f	2/6	0.72	0.24
r, t, s, f	1/6	0.02	0.003
r, t, r, f	1/6	0.16	0.027

   To compute the probabilities, we sum out variables as usual:
   P$(W_2 = r|I_1 = t, I_2 = f)$ = P$(I_1 = t, W_2 = r, I_2 = f)$ / P$(I_1 = t, I_2 = f)$
   We sum over W_1 using the rows from the table:
   $P(W_2 = r, I_1 = t, I_2 = f) = \sum_{w_1} P(W_1 = w_1, I_1 = t, W_2 = r, I_2 = f) = 0.24 + 0.027 = 0.267$
   Since all the rows in the table have $I_1 = t, I_2 = f$, the probability is just the sum of all the joint probabilities.
   P$(I_1 = t, I_2 = f)$ = 0.03 + 0.24 + 0.003 + 0.027 = 0.3
   So P$(W_2 = r|I_1 = t, I_2 = f)$ = 0.267 / 0.3 = 0.89.

4. What is a weakness of likelihood weighing sampling? How does Gibbs sampling work, and how does it address this limitation?
   Likelihood weighing only conditions on upstream evidence (so evidence only influences the choice of downstream variables).

   Gibbs sampling starts with an arbitrary instantiation of a complete sample (consistent with evidence), and then samples on one variable at a time, conditioned on all the rest, while keeping evidence consistent. This way, both upstream and downstream variables condition on evidence.