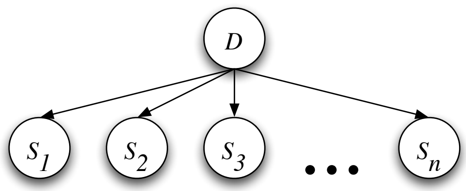
CSE 150. Assignment 2 *Winter 2015*

Out: *Tue Jan 20*
Due: *Tue Jan 27*
Reading: RN, Chapter 14; KN, Chapter 2.

2.1 Probabilistic reasoning

A patient is known to have contracted a rare disease which comes in two forms, represented by the values of a binary random variable $D \in \{0, 1\}$. Symptoms of the disease are represented by the binary random variables $S_k \in \{0, 1\}$, and knowledge of the disease is summarized by the belief network:



The conditional probability tables (CPTs) for this belief network are as follows. In the absence of evidence, both forms of the disease are equally likely, with prior probabilities: $P(D\!=\!0) = P(D\!=\!1) = \frac{1}{2}$. In the first form of the disease $(D\!=\!0)$, the first symptom occurs with probability one,

$$P(S_1\!=\!1|D\!=\!0) = 1,$$

while the k^{th} symptom (with $k\!\geq\!2$) occurs with probability

$$P(S_k\!=\!1|D\!=\!0) = \frac{f(k-1)}{f(k)},$$

where the function $f(k)$ is defined by

$$f(k) = 2^k + (-1)^k.$$

By contrast, in the second form of the disease $(D\!=\!1)$, all the symptoms are uniformly likely to be observed, with $P(S_k\!=\!1|D\!=\!1) = \frac{1}{2}$ for all k.

Suppose that on the k^{th} day of the month, a test is done to determine whether the patient is exhibiting the k^{th} symptom, and that each such test returns a positive result. Thus, on the k^{th} day, the doctor observes the patient with symptoms $\{S_1\!=\!1, S_2\!=\!1, \ldots, S_k\!=\!1\}$. Based on the cumulative evidence, the doctor makes a new diagnosis each day by computing the ratio:

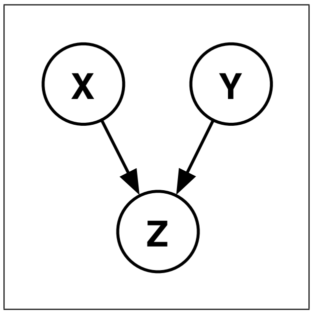
$$r_k = \frac{P(D\!=\!1|S_1\!=\!1, S_2\!=\!1, \ldots, S_k\!=\!1)}{P(D\!=\!0|S_1\!=\!1, S_2\!=\!1, \ldots, S_k\!=\!1)}.$$

If this ratio is greater than 1, the doctor diagnoses the patient with the $D\!=\!1$ form of the disease; otherwise, with the $D\!=\!0$ form.

(a) Compute the ratio r_k as a function of k. How does the doctor's diagnosis depend on the day of the month? Show your work.

(b) Does the diagnosis become more or less certain as more symptoms are observed? Explain.

2.2 Noisy-OR



Nodes:	$X \in \{0,1\}$, $Y \in \{0,1\}$, $Z \in \{0,1\}$
Noisy-OR CPT:	$P(Z=1\|X,Y) = 1 - (1-p_x)^X\,(1-p_y)^Y$
Parameters:	$p_x \in [0,1]$, $p_y \in [0,1]$, $p_y < p_x$

Suppose that the nodes in this network represent binary random variables and that the CPT for $P(Z|X,Y)$ is parameterized by a noisy-OR model, as shown above. Suppose also that

$$0 < P(X\!=\!1) < 1,$$
$$0 < P(Y\!=\!1) < 1,$$

while the parameters of the noisy-OR model satisfy:

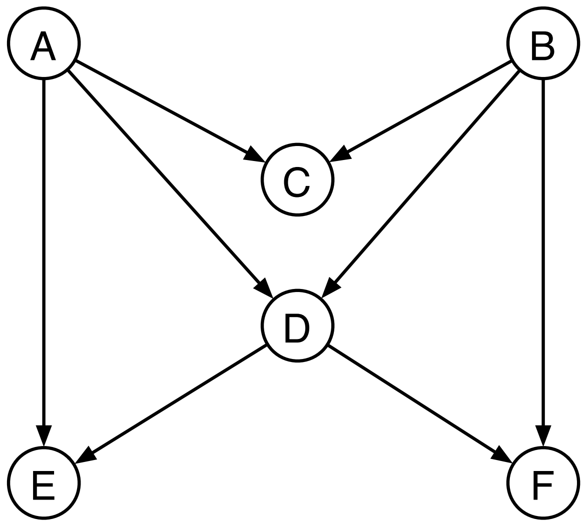
$$0 < p_y < p_x < 1.$$

Consider the following pairs of probabilities. In each case, indicate whether the probability on the left is equal $(=)$, greater than $(>)$, or less than $(<)$ the probability on the right.

(a)	$P(Z\!=\!1\|X\!=\!0,Y\!=\!0)$	$\square$	$P(Z\!=\!1\|X\!=\!0,Y\!=\!1)$
(b)	$P(Z\!=\!1\|X\!=\!0,Y\!=\!1)$	$\square$	$P(Z\!=\!1\|X\!=\!1,Y\!=\!0)$
(c)	$P(Z\!=\!1\|X\!=\!1,Y\!=\!1)$	$\square$	$P(Z\!=\!1\|X\!=\!1,Y\!=\!0)$
(d)	$P(X\!=\!1)$	$\square$	$P(X\!=\!1\|Z\!=\!1)$
(e)	$P(X\!=\!1)$	$\square$	$P(X\!=\!1\|Y\!=\!1)$
(f)	$P(X\!=\!1\|Y\!=\!1,Z\!=\!1)$	$\square$	$P(X\!=\!1\|Z\!=\!1)$
(g)	$P(X\!=\!1,Y\!=\!1,Z\!=\!1)$	$\square$	$P(X\!=\!1)\,P(Y\!=\!1)\,P(Z\!=\!1)$

2.3 Conditional independence

For the belief network shown below, indicate whether the following statements of (conditional) independence are **true (T)** or **false (F)**.



__________ $P(A, B) = P(A)\,P(B)$

__________ $P(E, F|D) = P(E|D)\,P(F|D)$

__________ $P(E, F|C, D) = P(E|C, D)\,P(F|C, D)$

__________ $P(E, F|A, B, D) = P(E|A, B, D)\,P(F|A, B, D)$

__________ $P(D|C) = P(D)$

__________ $P(D|A, B) = P(D|A, B, C)$

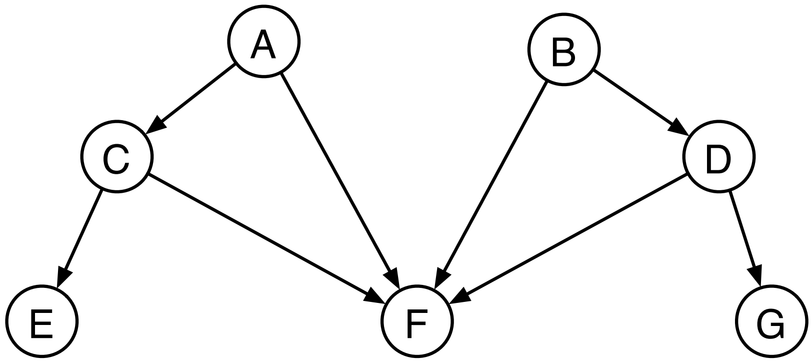
__________ $P(A|C, D) = P(A|C, D, F)$

__________ $P(A|B, C, D) = P(A|B, C, D, F)$

__________ $P(B|A, C, D, F) = P(B|A, C, D, F, E)$

__________ $P(B, F, A, E|C, D) = P(B, F|C, D)\,P(A, E|C, D)$

2.4 Subsets



Consider the following statements of (conditional) independence for the belief network shown above. Indicate the largest subset of nodes $\mathcal{S} \subset \{A, B, C, D, E, F, G\}$ for which each statement is true. Note that one possible answer is the empty set $\mathcal{S} = \emptyset$ or $\mathcal{S} = \{\}$ (whichever notation you prefer). The first has been done as an example.

$P(B) = P(B|\mathcal{S})$ $\quad\mathcal{S} = \{A, C, E\}$

$P(B|D) = P(B|\mathcal{S})$ ____________________

$P(B|D, F) = P(B|\mathcal{S})$ ____________________

$P(F) = P(F|\mathcal{S})$ ____________________

$P(F|C, D) = P(F|\mathcal{S})$ ____________________

$P(C|D) = P(C|\mathcal{S})$ ____________________

$P(C|F, G) = P(C|\mathcal{S})$ ____________________

$P(C|A, E, F) = P(C|\mathcal{S})$ ____________________

$P(E|F) = P(E|\mathcal{S})$ ____________________

$P(E|C) = P(E|\mathcal{S})$ ____________________

$P(A, B) = P(A, B|\mathcal{S})$ ____________________

2.5 Hangman

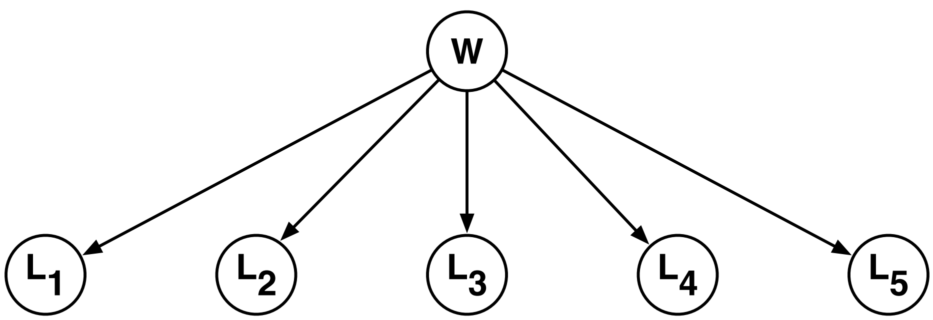
Consider the belief network shown below, where the random variable W stores a five-letter word and the random variable $L_i \in \{\mathtt{A}, \mathtt{B}, \ldots, \mathtt{Z}\}$ reveals only the word's ith letter. Also, suppose that these five-letter words are chosen at random from a large corpus of text according to their frequency:

$$P(W\!=\!w) \;=\; \frac{\text{COUNT}(w)}{\sum_{w'} \text{COUNT}(w')},$$

where $\text{COUNT}(w)$ denotes the number of times that w appears in the corpus and where the denominator is a sum over all five-letter words. Note that in this model the conditional probability tables for the random variables L_i are particularly simple:

$$P(L_i\!=\!\ell|W\!=\!w) \;=\; \begin{cases} 1 & \text{if } \ell \text{ is the } i\text{th letter of } w, \\ 0 & \text{otherwise.} \end{cases}$$

Now imagine a game in which you are asked to guess the word w one letter at a time. The rules of this game are as follows: after each letter (A through Z) that you guess, you'll be told whether the letter appears in the word and also where it appears. Given the *evidence* that you have at any stage in this game, the critical question is what letter to guess next.



Let's work an example. Suppose that after three guesses—the letters D, I, M—you've learned that the letter I does *not* appear, and that the letters D and M appear as follows:

$$\underline{\;\text{M}\;} \quad \underline{\quad} \quad \underline{\;\text{D}\;} \quad \underline{\quad} \quad \underline{\;\text{M}\;}$$

Now consider your next guess: call it ℓ. In this game the best guess is the letter ℓ that maximizes

$$P\Big(L_2\!=\!\ell \textbf{ or } L_4\!=\!\ell \;\Big|\; L_1\!=\!\mathtt{M}, L_3\!=\!\mathtt{D}, L_5\!=\!\mathtt{M}, L_2\!\not\in\!\{\mathtt{D},\mathtt{I},\mathtt{M}\}, L_4\!\not\in\!\{\mathtt{D},\mathtt{I},\mathtt{M}\}\Big).$$

In other works, pick the letter ℓ that is most likely to appear in the blank (unguessed) spaces of the word. For any letter ℓ we can compute this probability as follows:

$$\begin{aligned}
&P\Big(L_2\!=\!\ell \textbf{ or } L_4\!=\!\ell \;\Big|\; L_1\!=\!\mathtt{M}, L_3\!=\!\mathtt{D}, L_5\!=\!\mathtt{M}, L_2\!\not\in\!\{\mathtt{D},\mathtt{I},\mathtt{M}\}, L_4\!\not\in\!\{\mathtt{D},\mathtt{I},\mathtt{M}\}\Big) \\
&\quad= \sum_w P\Big(W\!=\!w, L_2\!=\!\ell \textbf{ or } L_4\!=\!\ell \;\Big|\; L_1\!=\!\mathtt{M}, L_3\!=\!\mathtt{D}, L_5\!=\!\mathtt{M}, L_2\!\not\in\!\{\mathtt{D},\mathtt{I},\mathtt{M}\}, L_4\!\not\in\!\{\mathtt{D},\mathtt{I},\mathtt{M}\}\Big), \quad \boxed{\textbf{marginalization}} \\
&\quad= \sum_w P(W\!=\!w|L_1\!=\!\mathtt{M}, L_3\!=\!\mathtt{D}, L_5\!=\!\mathtt{M}, L_2\!\not\in\!\{\mathtt{D},\mathtt{I},\mathtt{M}\}, L_4\!\not\in\!\{\mathtt{D},\mathtt{I},\mathtt{M}\}\Big)\; P(L_2\!=\!\ell \textbf{ or } L_4\!=\!\ell|W\!=\!w) \;\boxed{\textbf{product rule \& CI}}
\end{aligned}$$

where in the third line we have exploited the conditional independence (**CI**) of the letters L_i given the word W. Inside this sum there are two terms, and they are both easy to compute. In particular, the second term is more or less trivial:

$$P(L_2\!=\!\ell \textbf{ or } L_4\!=\!\ell|W\!=\!w) \;=\; \begin{cases} 1 & \text{if } \ell \text{ is the second or fourth letter of } w \\ 0 & \text{otherwise.} \end{cases}$$

And the first term we obtain from Bayes rule:

$$\begin{aligned} & P\Big(W\!=\!w|L_1\!=\!\texttt{M}, L_3\!=\!\texttt{D}, L_5\!=\!\texttt{M}, L_2\!\notin\!\{\texttt{D},\texttt{I},\texttt{M}\}, L_4\!\notin\!\{\texttt{D},\texttt{I},\texttt{M}\}\Big) \\ & = \frac{P\Big(L_1\!=\!\texttt{M}, L_3\!=\!\texttt{D}, L_5\!=\!\texttt{M}, L_2\!\notin\!\{\texttt{D},\texttt{I},\texttt{M}\}, L_4\!\notin\!\{\texttt{D},\texttt{I},\texttt{M}\}|W\!=\!w\Big)P(W\!=\!w)}{P(L_1\!=\!\texttt{M}, L_3\!=\!\texttt{D}, L_5\!=\!\texttt{M}, L_2\!\notin\!\{\texttt{D},\texttt{I},\texttt{M}\}, L_4\!\notin\!\{\texttt{D},\texttt{I},\texttt{M}\})} \qquad \boxed{\textbf{Bayes rule}} \end{aligned}$$

In the numerator of Bayes rule are two terms; the left term is equal to zero or one (depending on whether the evidence is compatible with the word w), and the right term is the prior probability $P(W = w)$, as determined by the empirical word frequencies. The denominator of Bayes rule is given by:

$$\begin{aligned} & P(L_1\!=\!\texttt{M}, L_3\!=\!\texttt{D}, L_5\!=\!\texttt{M}, L_2\!\notin\!\{\texttt{D},\texttt{I},\texttt{M}\}, L_4\!\notin\!\{\texttt{D},\texttt{I},\texttt{M}\}) \\ & = \sum_w P(W\!=\!w, L_1\!=\!\texttt{M}, L_3\!=\!\texttt{D}, L_5\!=\!\texttt{M}, L_2\!\notin\!\{\texttt{D},\texttt{I},\texttt{M}\}, L_4\!\notin\!\{\texttt{D},\texttt{I},\texttt{M}\}), \qquad \boxed{\textbf{marginalization}} \\ & = \sum_w P(W\!=\!w)P(L_1\!=\!\texttt{M}, L_3\!=\!\texttt{D}, L_5\!=\!\texttt{M}, L_2\!\notin\!\{\texttt{D},\texttt{I},\texttt{M}\}, L_4\!\notin\!\{\texttt{D},\texttt{I},\texttt{M}\}|W\!=\!w), \qquad \boxed{\textbf{product rule}} \end{aligned}$$

where again all the right terms inside the sum are equal to zero or one. Note that the denominator merely sums the empirical frequencies of words that are compatible with the observed evidence.

Now let's consider the general problem. Let E denote the evidence at some intermediate round of the game: in general, some letters will have been guessed correctly and their places revealed in the word, while other letters will have been guessed incorrectly and thus revealed to be absent. There are two essential computations. The first is the *posterior* probability, obtained from Bayes rule:

$$P(W\!=\!w|E) \;=\; \frac{P(E|W\!=\!w)\,P(W\!=\!w)}{\sum_{w'} P(E|W\!=\!w')\,P(W\!=\!w')}.$$

The second key computation is the *predictive* probability, based on the evidence, that the letter ℓ appears somewhere in the word:

$$P\Big(L_i\!=\!\ell \text{ for some } i\!\in\!\{1,2,3,4,5\}\Big|E\Big) \;=\; \sum_w P\Big(L_i\!=\!\ell \text{ for some } i\!\in\!\{1,2,3,4,5\}\Big|W\!=\!w\Big)P\Big(W\!=\!w\Big|E\Big).$$

Note in particular how the first computation feeds into the second. Your assignment in this problem is implement both of these calculations. **You may program in the language of your choice.**

(a) Download the file *hw2_word_counts_05.txt* that appears with the homework assignment. The file contains a list of 5-letter words (including names and proper nouns) and their counts from a large corpus of Wall Street Journal articles (roughly three million sentences). From the counts in this file compute the prior probability $P(w) = \textsc{count}(w)/\textsc{count}_{\text{total}}$. **As a sanity check, print out the ten <u>most frequent</u> 5-letter words, as well as the ten <u>least frequent</u> 5-letter words. Do your results make sense?**

(b) Consider the following stages of the game. For each of the following, indicate the best next guess—namely, the letter ℓ that is most likely (probable) to be among the missing letters. Also report the probability $P(L_i = \ell \text{ for some } i \in \{1, 2, 3, 4, 5\} | E)$ for your guess ℓ.

(i) **First guess:** ___ ___ ___ ___ ___

(ii) **None correctly guessed:** ___ ___ ___ ___ ___
Incorrectly guessed: `E`, `O`

(iii) **Correctly guessed:** `D` ___ ___ `I` ___
Incorrectly guessed: none

(iii) **Correctly guessed:** `D` ___ ___ `I` ___
Incorrectly guessed: `A`

(v) **Correctly guessed:** ___ `U` ___ ___ ___
Incorrectly guessed: `A`, `E`, `I`, `O`, `S`

(c) Turn in a **hard-copy printout** of your source code. Do not forget the source code: it is worth many points on this assignment.

(d) *No credit, totally optional*: if you'd like to create a similar program for longer words, you can also download the count files for words of length 6-10. Have fun.