10708 Graphical Models: Homework 1

Due October 1st, beginning of class

September 29, 2008

Instructions: There are five questions on this assignment. The last question involves coding, which should be done in MATLAB. Do *not* attach your code to the writeup. Instead, copy your implementation to

`/afs/andrew.cmu.edu/course/10/708/Submit/your_andrew_id/HW1`

Refer to the web page for policies regarding collaboration, due dates, and extensions.

1 Conditional Probability [23] [Dhruv]

1.1 [4 pts]

Let $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$ be three disjoint sets of variables such that $\mathcal{S} = \mathcal{X} \cup \mathcal{Y} \cup \mathcal{Z}$. Prove that $P \models (\mathcal{X} \perp \mathcal{Y} | \mathcal{Z})$ if and only if we can write P in the form: $P(\mathcal{S}) = f(\mathcal{X}, \mathcal{Z}) g(\mathcal{Y}, \mathcal{Z})$

1.2 [5 pt]

Is it possible for both f and g above to be probability distributions over their respective sets of variables? Formally, is it possible for every distribution P over $(\mathcal{X} \cup \mathcal{Y} \cup \mathcal{Z})$ with the independency above, to be expressed as a product of a distribution over $(\mathcal{X} \cup \mathcal{Z})$ and a distribution over $(\mathcal{Y} \cup \mathcal{Z})$? Justify your answer.
(*Hint*: look at the marginal probability of $\mathcal{Z}$; you may assume that the variables are binary if you wish.)

1.3 [3 pts]

Prove or disprove (by providing a counter-example) each of the following properties of independence:

1. $(X \perp Y, W|Z)$ implies $(X \perp Y|Z)$.
2. $(X \perp Y|Z)$ and $(X, Y \perp W|Z)$ imply $(X \perp W|Z)$.
3. $(X \perp Y, W|Z)$ and $(Y \perp W|Z)$ imply $(X, W \perp Y|Z)$.

1.4 [3 pts]

Provide an example of a distribution $P(X_1, X_2, X_3)$ where for each $i \neq j$, we have that $(X_i \perp X_j) \in \mathcal{I}(P)$, but we also have that $(X_1, X_2 \perp X_3) \notin \mathcal{I}(P)$.

1.5 [8 pts]

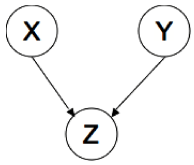


Figure 1: Graphical Model for Prob. 1.5

Let X, Y, Z be binary random variables with joint distribution given by the graphical model shown above (v-structure). We define the following shorthands:

$$a \triangleq P(X = t);\ b \triangleq P(X = t \mid Z = t);\ c \triangleq P(X = t, \mid Z = t, Y = t)$$

1. For all the following cases, provide examples of conditional probability tables (CPTs) (and compute the quantities, a, b, c), which make the statements true:
   (a) a > c
   (b) a < c < b
   (c) b < a < c
2. Think of X, Y as causes and Z as a common effect, and for all the above cases summarize (in a sentence or two) why the statements are true for your examples.

(Hint: Think about positive and negative correlations along edges)

2 Graph Independencies [12 pts] [Dhruv]

2.1 [4 pts]

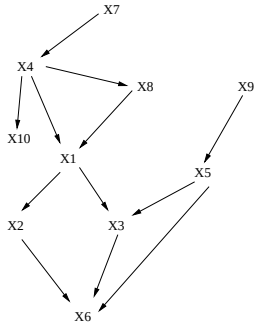


Figure 2: Graphical Model for Prob. 2

Let $\mathbf{X} = \{X_1, \ldots, X_n\}$ be a random vector with distribution given by the graphical model in Figure 2. Consider variable X_1. What is the minimal subset of the variables, $\mathbf{A} \subseteq \mathcal{X} - \{X_1\}$, such that X_1 is independent of the rest of the variables, $\mathcal{X} - \mathbf{A} \cup \{X_1\}$, given $\mathbf{A}$? Justify your answer.

2.2 [8 pts]

Now, let the distribution of $\mathbf{X}$ be given by some graphical model instance $\mathbf{B} = (\mathcal{G}, P)$. Consider variable X_i. What is the minimal subset of the variables, $\mathbf{A} \subseteq \mathcal{X} - \{X_i\}$, such that X_i is independent of the rest of the variables, $\mathcal{X} - \mathbf{A} \cup \{X_i\}$, given $\mathbf{A}$? Prove that this subset is necessary and sufficient.

(Hint: Think about the variables that X_i cannot possibly be conditionally independent of, and then think some more)

2.3 *Extra Credit* [8 pts]

Show how you could efficiently compute the distribution over a variable X_i given some assignment to all the other variables in the network: $P(X_i | x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$.
Your procedure should not require the construction of the entire joint distribution $P(X_1, \ldots, X_n)$.
Specify the computational complexity of your procedure.

3 Factorization [15 pts] [Dhruv]

Let $\mathcal{G}$ be a bayesian network graph over a set of random variables $\mathcal{X}$ and let P be a joint distribution over the same space. Show that if P factorizes according to $\mathcal{G}$, then $\mathcal{G}$ is an $I-$map for P.
(Hint: See example in Section 3.2.3.3 of Koller and Friedman)

4 Marginalization [15 pts] [Amr]

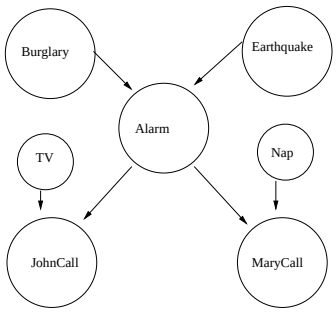


Figure 3: Burglar Alarm Network

1. Consider the *Burglar Alarm* network shown in Figure 3. Construct a Bayesian network over all of the nodes except for *Alarm*, which is a minimal $I-$map for the marginal distribution over those variables defined by the above network. Be sure to get all dependencies that remain from the original network.

   (*Hint*: Consider all active trails ,$\langle X_1 \rightleftharpoons X_2 \cdots \rightleftharpoons X_n \rangle$,that go through *Alarm* , make sure that there still an active trail under the same conditions —i.e. observed variables — between X_1 and X_n in the marginalized network.)

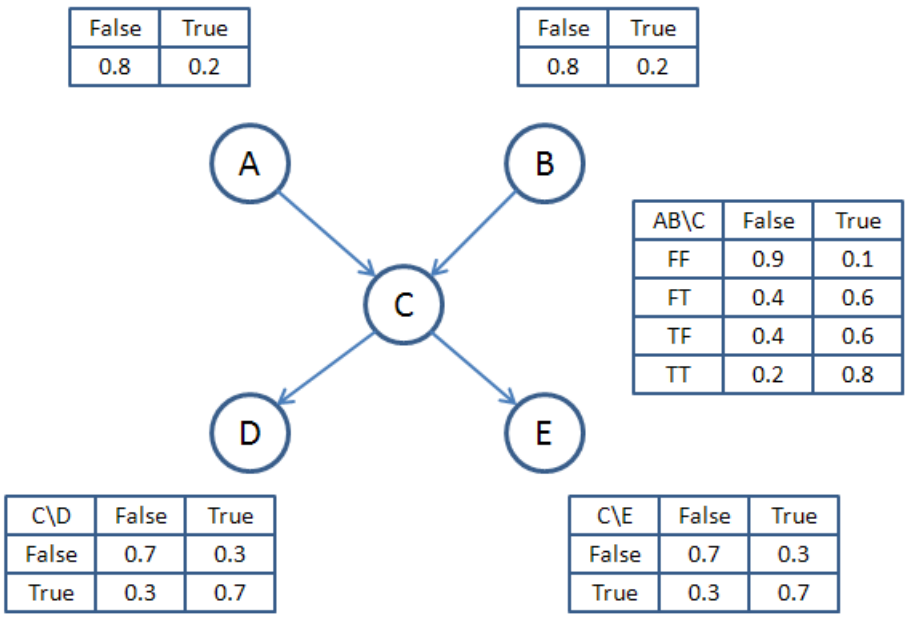
2. Generalize the procedure you used to solve the above into a node-elimination algorithm. That is, define an algorithm that transforms the structure of $\mathcal{G}$ into $\mathcal{G}'$ such that one of the nodes X_ι of $\mathcal{G}$ is not in $\mathcal{G}'$ and $\mathcal{G}'$ is an $I-$map of the marginal distribution over the remaining variables as defined by $\mathcal{G}$.

   (*Hint:*: Consider the relationship between the variables you added edges to in part 1 and the node being marginalized. Now, can you devise a set of generic rules over these affected variables? It would be helpful to think about different local configurations around X_ι)

5 [35 pts] Learning PDAGs [Amr]

Given samples from a probability distribution $\mathcal{P}$, we would like to learn a graph $\mathcal{G}$ which is a $\mathcal{P}$-map for $\mathcal{P}$. A PDAG is a compact way of representing all $\mathcal{P}$-maps for a given distribution. In this question, you will implement an algorithm for learning a PDAG from samples from $\mathcal{P}$ and examine its behavior in details.

Note: You are not allowed to use any code that is not given to you in the homework or that is not part of a standard Matlab distribution.



False	True
0.8	0.2

False	True
0.8	0.2

AB\C	False	True
FF	0.9	0.1
FT	0.4	0.6
TF	0.4	0.6
TT	0.2	0.8

C\D	False	True
False	0.7	0.3
True	0.3	0.7

C\E	False	True
False	0.7	0.3
True	0.3	0.7

Figure 4: Network 1

5.1 [2 points]

Consider the network shown in Figure 4. Draw its skeleton and PDAG. How many different graphs are encoded in this PDAG?

5.2 Implementation

Implement the PDAG learning algorithm discussed in class and in Figure 3.21 in Koller and Friedman . You need to implement the following steps:

- *Build-Skeleton*: An algorithm that constructs an undirected graph $\mathcal{S}$ that contains an edge $X - Y$ if X and Y are adjacent in $\mathcal{G}$.
- *Mark-Immoralities*: An algorithm that detects immoralities and directs their edges appropriately in $\mathcal{S}$. **Note:** when examining a potential immorality of the form $X - Y - Z$ with no edge between $X - Z$, DO consider all possible sets U, of bounded size up to $2 * d$, that contain Y. $X - Y - Z$ is an immorality if $\neg\exists U, y \in U, X \perp Z|U$. This is in contrast to the implementation in Koller and Friedman in which only the witness separator between X and Z is examined — we will come to this point later.
- *Orient-Edges*: An algorithm that applies the rules in Figure 3.20 to propagate the constraints imposed by the discovered immoralities and direct some edges in $\mathcal{S}$ to avoid adding cycles and/or additional immoralities. **Note**: you ONLY need to implement rules 1 and 2.
- *Testing for (conditional) independence*: Since we only have access to the original distribution through its samples, we need to empirically answer independence queries like $X \perp Y|Z$. We will use Mutual Information (MI) defined as follows:

$$\hat{\mathrm{I}}(X;Y|Z) = \sum_{x,y,z} \hat{p}(x,y,z)\log\frac{\hat{p}(x,y|z)}{\hat{p}(x|z)\hat{p}(y|z)}$$

  where $\hat{p}$ is the empirical probability. In addition, we define a threshold t, and we declare that $X \perp Y|Z$ if $\hat{\mathrm{I}}(X;Y|Z) \leq t$.

Submit your implementation to your AFS code directory. Answer the following questions in your writeup

1. *[10 points]* Use the function *genSamples_net1.m* to generate 1000 samples from this network. Apply your learning algorithm on these samples using $t = .02$, $t = .06$ and $t = .07$, and draw the resulting skeleton and PDAG in each case. What do you observe?
2. *[5 points]* The value of the threshold t is important in recovering the correct structure. To understand this point further, compute the following empirical mutual information values:
   - $I(A;C|\{D,E\})$ and $I(B;C|\{D,E\})$
   - $I(D;C|\{A,B\})$ and $I(E;C|\{A,B\})$
   - $I(A;B|\{C\})$

   Can you now explain the behavior you observed by varying t in part 2?
3. *[8 points]* The number of samples used to estimate the empirical probabilities can introduce another source of error when answering independence queries, and thus affect

the final learnt PDAG. fix $t = .02$ and vary the number of samples along the range $[10, 50, 100, 300]$. Draw the resulting skeletons and PDAGs in each case.

4. *[10 points]* Consider the network in Figure 5, where α determines the strength of dependencies in the network. The higher the value of α, the more dependent the variables, and the easier it is to identify the correct structure. Using the function $genSample\_net2(\alpha, N)$, fix $N = 5000$, $t = .007$ and vary α in the range $[.4, .7, .9]$. For each setting of α apply your code and draw the resulting skeleton and PDAG. what do you observe and what can you conclude?

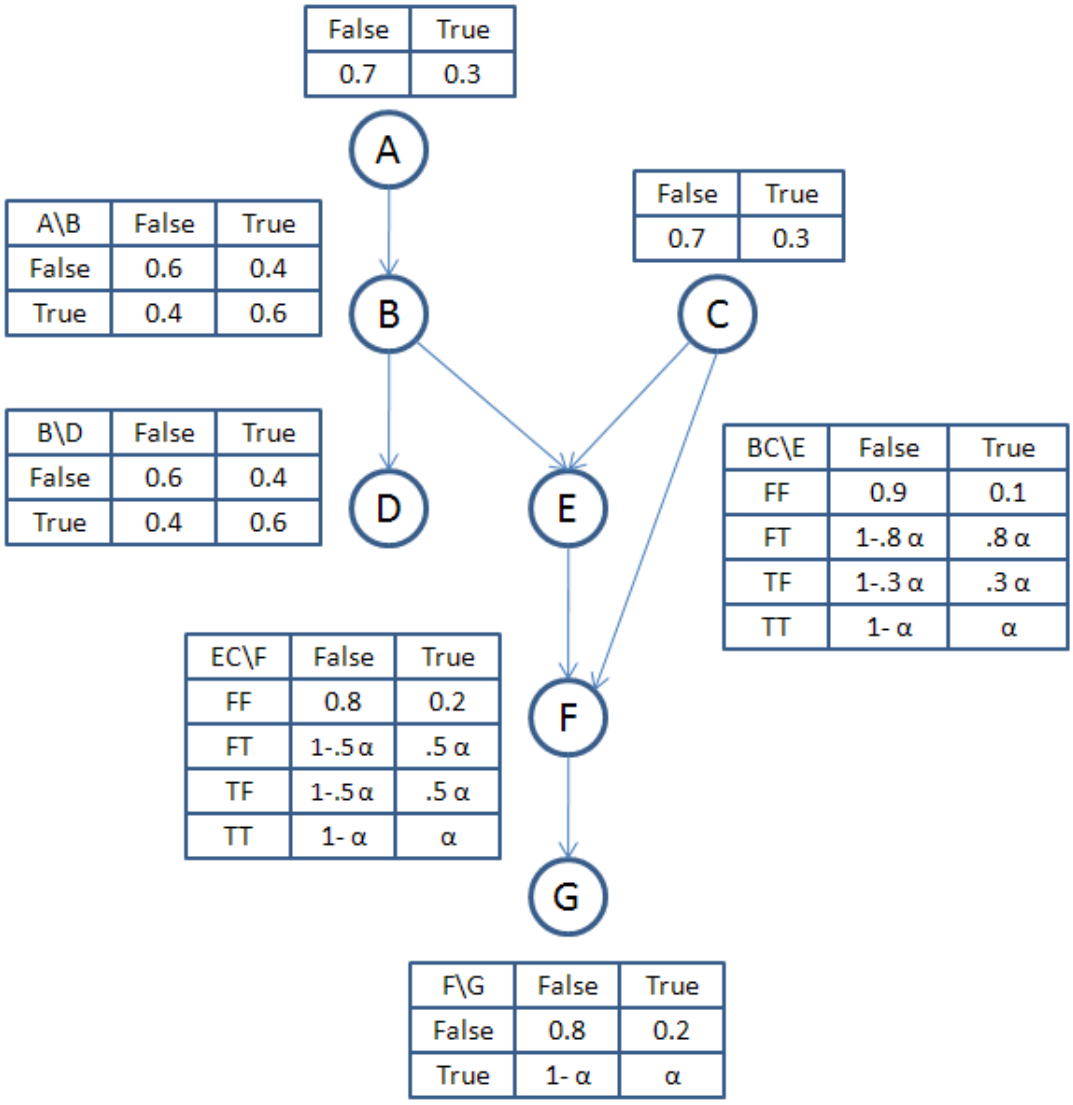


Figure 5: Network 2

5.3 [Extra Credit]: Efficient Implementation and Robustness

1. *[5 points]* Modify your implementation of *Mark-Immoralities* to follow the reading in Figure 3.18 in Koller and Friedman. In other words, only examine the separator that was recorded as a witness for the removal of the edge between X and Z. Now using network 1 draw $N = 1000$ samples and fix the threshold at $t = .02$, then apply your PDAG learning algorithm and draw the resulting skeleton and PDAG. What do you observe? NOTE: results here depends on the way you traverse the subsets U in *build-skeleton*— we are assuming that you visit them in increasing size and stop iterating once a witness is found, if you followed another scheme, please clearly indicate it in your witting.

2. *[2 points]* Using the setting in 5.3.1, vary t until you recover the correct PDAG and record t.

3. The extra step in part 5.3.1 indeed results in a more efficient computation and is sound if independence queries are answered directly from $\mathcal{P}$ rather than being estimated from the data:

   - *[2 points]* Prove that claim. *hint:* you may make use of the result of Lemma 3.4.8 in Koller and Friedman.

   - *[6 points]* Can you explain the behavior you observed in part 5.3.1? *hint*: examine the recorded witness separator that caused the wrong behavior and its associated induced mutual information. What happened when you lowered t in part 5.3.2? Why your implementation in part 5.2 does not suffer from this problem?