Probabilistic Graphical Models

HWK#2 Part B
Assigned Sunday, Feb. 9, 2020
Due: Thursday, March 5, 2020

## Problem 7

*Consider the distribution*

$$p(y|x_1,\ldots,x_T)p(x_1)\prod_{t=2}^{T}p(x_t|x_{t-1})$$

*where all variables are binary.*

1. *Draw a junction tree for this distribution and explain the computational complexity of computing* $p(x_T)$*, as suggested by the junction tree algorithm.*

2. *By using an approach different from the plain JTA above, explain how* $p(x_T)$ *can be computed in time that scales linearly with* $T$*.*

# Problem 8. Inference in a Fully Observed Bayesian Network

In this question, you will implement and experiment with directed graphical models and briefly report your experiences in doing so. We provide you with data from a simulated "medical domain".

**Medical Domain Data** We have provided you with a joint probability distribution of symptoms, conditions and diseases. Certain diseases are more likely than others given certain symptoms, and a model such as this can be used to help doctors make a diagnosis (don't actually use this for diagnosis though!). The ground-truth joint probability distribution consists of twelve binary random variables and contains $2^{12}$ possible configurations (numbered 0 to 4095), which is small enough that you can enumerate them exhaustively. The variables are as follows:

0. **IsSummer** true if it is the summer season, false otherwise.
1. **HasFlu** true if the patient has the flu.
2. **HasFoodPoisoning** true if the patient has food poisoning.
3. **HasHayFever** true if patient has hay fever.
4. **HasPneumonia** true if the patient has pneumonia.
5. **HasRespiratoryProblems** true if the patient has problems in the respiratory system.
6. **HasGastricProblems** true if the patient has problems in the gastro-intestinal system.
7. **HasRash** true if the patient has a skin rash.
8. **Coughs** true if the patient has a cough.
9. **IsFatigued** true if the patient is tired and fatigued.
10. **Vomits** true if the patient has vomited.
11. **HasFever** true if the patient has a high fever.

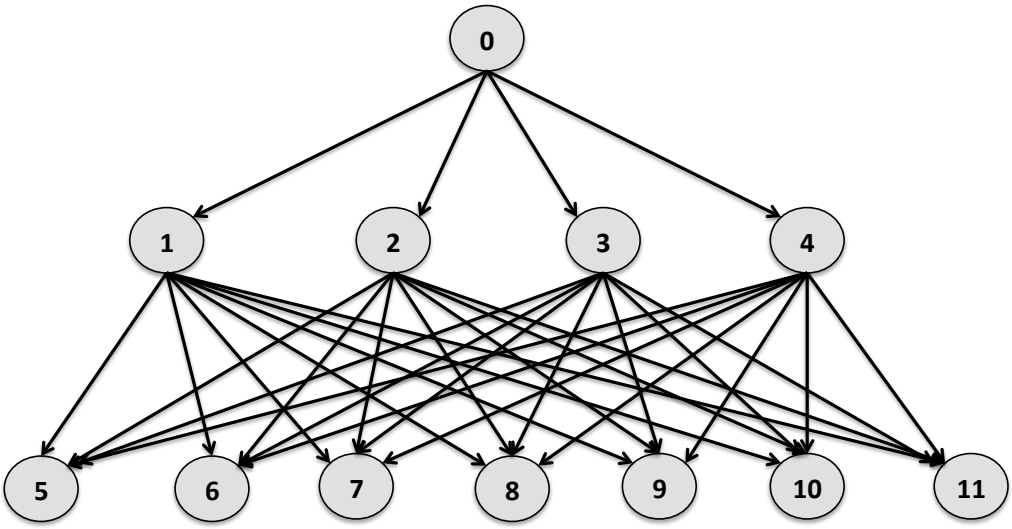


Figure 1: Directed GM for Problem 4.2. Here $0 \ldots 11$ are variables representing "IsSummer" ... "HasFever" as described in Problem 4.1

You can get the data at folder "data-hwk2" from canvas. The archive contains two files:

1. `joint.dat`: The true joint probability distribution over the twelve binary variables. Since each variable is binary, we can represent a full variable assignment as a bit string. This file lists all $2^{12}$ assignments (one in each line) as pairs "Integer Probability" where "Integer" is an integer encoding of the bit string. Specifically, assuming false=0 and true=1, an assignment to all variables results in a 12-bit binary number (with the index of the variables shown in parentheses above) which is converted to a decimal number. For example, assignment 0 represents all variables are false, 1 represents only IsSummer is true, 2 represents only HasFlu is true, and so on.

2. `dataset.dat`: The dataset consists of samples from the above probability distribution. Each line of the file contains a complete assignment to all the variables, encoded as an integer (as described above).

**Answer following questions** Use principled algorithms you learned in inference - Variable Elimination and Message Passing. Consider the following 3 queries and solve the follow up problems in parts (a) through (e):

- (query 1) What is the probability a patient has flu given they are coughing and have a high fever? (Observed Variables: HasFever=true, Coughs=true; Query Variable: HasFlu)
- (query 2) What is the probability distribution over the symptoms (HasRash, Coughs, IsFatigued, Vomits, and HasFever) given the patient has pneumonia?
- (query 3) What is the probability of vomiting in summer?
- **a) Variable Elimination** For query 1, show the moralized graph. Choose a good elimination ordering and write down the variable elimination procedure. Show the intermediate factors produced after eliminating each variable. What is the time and space complexity of the variable elimination algorithm?
  Note:- Do not substitute numbers for the marginal and conditional probabilities yet. You just have to describe the procedure for a good elimination ordering similar to lecture.

  Note: The true distribution is generated using the model described in Figure 1 but some small noise has been added to it. So your answers will not exactly match the answers using the true distribution but will be pretty close.
- **b) Message Passing on Clique Trees/Factor Graphs**

  (i) Now, assume that you were instead doing variable elimination on clique trees. For the first query, use the same elimination ordering you chose before. Construct a clique tree. What do the messages here correspond to? Again, you need not substitute any numbers. What is the time and space complexity of the resulting algorithm?

  (ii) Now consider, instead, the view of message passing on a factor graph for query 1. What do the factors here correspond to? Write down any 3 messages from variables to factors and any 3 messages from the factors to variables. Again, do not substitute any numbers. What is the time and space complexity of the resulting algorithm?
- **c) Estimating Parameters:** consider the Directed GM for the problem shown in Figure 1 as your model. Use `dataset.dat` to estimate the parameters of this graphical model, i.e., estimate the marginals $P(X_i)$ for nodes that do not have any parents (in our case node 0 only) and the conditionals $P(X_i|\pi(X_i))$ for all other nodes $X_i$ ($\pi(X_i)$ represents the set of parents of $X_i$). You can do this by simply counting and normalizing, i.e. enumerate all the assignments in the dataset, and for each variable v, count the number of times a variable is true for each assignment to its parents, and then normalize the counts using the total number of times the parents had that assignment.

- **d) Model Accuracy:** Measure the similarity of the model to the true joint probability distribution (i.e., `joint.dat`. That is, for each assignment, how similar are the probabilities returned by your model to the true probability distribution. To keep things simple, you can compare the distributions based on their L1-distance. That is, for each assignment $a_i$ to all the variables, obtain p($a_i$) from the true joint distribution ($(i+1)^{th}$ row in `joint.dat`) and p($\hat{a}_i$) using your model. The L1 distance is defined as $|p(a_0) - \hat{p}(a_0)| + |p(a_1) - \hat{p}(a_1)| + \cdots + |p(a_{4095}) - \hat{p}(a_{4095})|$.

  Note: An alternative distance measure more appropriate to probability distributions is KL-divergence. You could evaluate the distance using the Jensen–Shannon divergence (which is essentially just symmetrised KL divergence). [1]. However, this is **not required**.

- **e) Implement Variable Elimination or Message Passing** Now, implement one of the above algorithms: you can directly implement the variable elimination procedure or you could implement this using the message passing protocol. If you use message passing, you can use the clique tree view or the factor graph view. Your implementation should be general i.e. your implementation should be able to handle any valid query. You can validate your implementation using the three queries described above. Report the final answers for all the three queries and run times. Compare the result and runtime to the case when you simply used the samples to answer the queries.

**Report** You should provide a short (roughly 1 page) summary describing your explorations and results. Discuss the accuracy (how close were you to the true joint probability distribution?) of your model. What was the likelihood of the data according to the true distribution.

Submit a hard copy of the report with the assignment. Please also upload the complete source code of your implementation to canvas. Remember to include a small Readme file and a script that would help us execute your code.

[1] `http://en.wikipedia.org/wiki/Jensen-Shannon_divergence`