

CS 476/676 Machine Learning: Data to Models, Spring 2017

Assignment 2a

Due by 11:59 pm on Monday, March 6

1 Instructions

This is the first part of the two part assignment. This is a fairly short assignment with no programming. As usual, read the document carefully.

1.1 What to Hand In

All of your submission files should be handed in as a single archive named `hw2-username1-username2.zip`, where `username` is replaced with your JHED IDs(not the Hopkins ID in SIS) of the team-mates. See the `Deliverables` section in each section for specific details.

Hand in the zip archive by creating a private note to the instructors on Piazza with the title *Submission from <names of team members>*, and attaching your zip file to that note. The note should be submitted to the `hw2_submission` folder.

All your written solutions should go in a single `pdf` file named `writeup.pdf`. Put the names of the team members in the writeup file. **Latex is required for the writeup.**

Any questions/discussion related to the assignment should be posted in `hw2_qa` folder.

1.2 Submission Policies

Please note the following:

- **Collaboration:** Please work in groups of 2 or 3. The homeworks are a way for you to work through the material you're learning in this class on your own. But, by working in a group, and debugging each other's solutions, you'll have a chance to learn the material in more depth. The recommended format for tackling these problem sets is the following. Write a high level sketch of the solution for all of the problems on your own. Meet as a group to brainstorm your solutions and converge on a solution as a group. It is important that you have a good understanding of how you'd have approached the problem independently before discussing your solution with the other group members. Developing this intuition will serve you well in the final exam where you will be required to work on your own. Pursuant to your group meeting, write up the solutions on your own. Thereafter, meet as a group to clean up and submit a final write up as a group. By now, each of you should have a solid understand of the concepts involved, and by meeting as a group, you've had a chance to see common ways in which one can make mistakes. Submit your final solution as a final writeup for the group. Your submission should include the names of every team member.
- **Late Submissions:** We allow each student to use up to 3 late days over the semester. You have late days, not late hours. This means that if your submission is late by any amount of time past the deadline, then this will use up a late day. If it is late by any amount beyond 24 hours past the deadline, then this will use a second late day. **You will not receive any credit, if your late days are up and you submit after the deadline. When you submit an assignment as a team, every team member will lose late days if the assignment is submitted late.**

2 Network Manipulation (15 points)

One operation on Bayesian networks that arises in many settings is the marginalization of some node in the network. Consider the network below, which models the flu prevalence in a population. (This is a different network than what you designed in Assignment 1.)

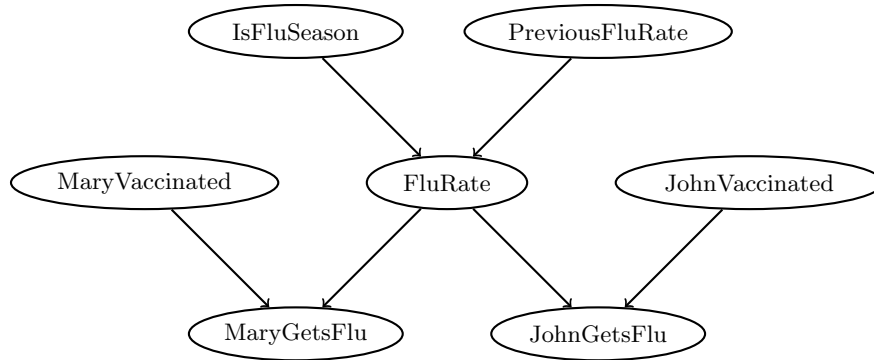


Figure 1: Network \mathcal{F} .

- [10 points]** Draw a new network that is a minimal I-map over the marginal distributions $P_{\mathcal{F}}(\text{IsFluSeason}, \text{PreviousFluRate}, \text{MaryVaccinated}, \text{JohnVaccinated}, \text{MaryGetsFlu}, \text{JohnGetsFlu})$. Be sure to maintain all dependencies in the original graph.
- [5 points]** Generalize the above procedure to come up with a node-elimination algorithm for Bayesian networks. Given a graph \mathcal{G} and a node to eliminate E , the algorithm should produce a new graph \mathcal{G}' that is a minimal I-map over the marginal distributions $P_{\mathcal{G}}(\mathcal{X} - \{E\})$. Keep it brief.

3 Network Queries (20 points)

3.1 Requisite Nodes [8 points]

Let's consider the sensitivity of a particular query $P(X|\mathbf{Y})$ to the CPD of a particular node Z . Let X and Z be nodes (which are not directly connected) and \mathbf{Y} be a set of nodes. We say that Z has a *requisite CPD* for answering the query $P(X|\mathbf{Y})$ if there are two networks \mathcal{B}_1 and \mathcal{B}_2 that have identical graph structure \mathcal{G} and identical CPDs everywhere except at the node Z , and where $P_{\mathcal{B}_1}(X|\mathbf{Y}) \neq P_{\mathcal{B}_2}(X|\mathbf{Y})$; in other words, the CPD of Z affects the answer to this query.

This type of analysis is useful in various settings, including determining which CPDs we need to acquire for a certain query.

Suppose we modify \mathcal{G} into a graph \mathcal{G}' which is identical to \mathcal{G} except contains a new “dummy” node \hat{Z} which is a parent of Z (thereby altering the CPD of Z). One way to test whether Z is a requisite probability node for $P(X|\mathbf{Y})$ is to test whether \hat{Z} has an active trail to X given \mathbf{Y} in \mathcal{G}' —if so, you can conclude that altering the CPD of Z would affect the result of $P(X|\mathbf{Y})$.

- a. [4 points] Prove that the above strategy is a sound criterion for determining whether Z is a requisite probability node for $P(X|\mathbf{Y})$ in \mathcal{G} . Specifically, show that if this criterion fails to identify Z as a requisite node, then for all pairs of networks $\mathcal{B}_1, \mathcal{B}_2$, $P_{\mathcal{B}_1}(X|\mathbf{Y}) = P_{\mathcal{B}_2}(X|\mathbf{Y})$.
- b. [4 points] Show that this criterion is weakly complete (like d-separation) in the sense that, if it identifies Z as a requisite in \mathcal{G} , there exists some pair of networks $\mathcal{B}_1, \mathcal{B}_2$ such that $P_{\mathcal{B}_1}(X|\mathbf{Y}) \neq P_{\mathcal{B}_2}(X|\mathbf{Y})$.

Note: Proofs are not more than a few lines.

3.2 Comparing Network Types: Disease severity over time [12 points]

In this section, we will consider modeling the severity of a disease over time using *linear chain* graphical models, which are commonly used to model discrete time series data.

The random variable Y_i denotes the disease severity (None, Mild, Severe) on day i . The random variable $S_{i,j}$ indicates a symptom j on day i , such as the patient's temperature or whether or not she has a cough (for example, if Mary had a case of the flu, we could model it day-by-day using such a model). The symptoms \mathbf{S} are observed. The severity \mathbf{Y} is not observed, but it can be inferred from the observed symptoms.

Figure 2 shows three different types of networks to model this. The first is directed graph which encodes $P(\mathbf{Y}, \mathbf{S})$, called a hidden Markov model (HMM). The second is a directed graph which encodes $P(\mathbf{Y}|\mathbf{S})$ called a maximum-entropy Markov model (MEMM). The third is a type of conditional random field (CRF), which also models $P(\mathbf{Y}|\mathbf{S})$. CRFs can be partially directed or undirected; a partially directed version is shown here.

The differences between these models are subtle, and even in practice it is not always clear which model to use. As you'll learn later in the semester, the choice of model has implications and tradeoffs regarding learning and inference. In this problem, we want you to think about the subtle differences in assumptions made by these models.

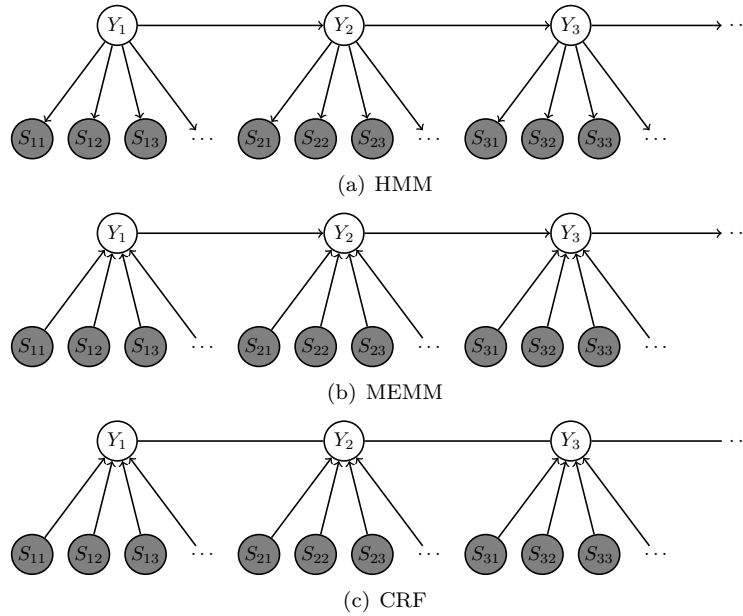


Figure 2: Three types of linear chain models, described in Section 3.2. Gray nodes are observed variables; white nodes are unobserved.

Suppose we are modeling Mary’s flu over a series of days and suppose that on day 1 and 2, Mary’s symptoms are consistent with the flu. However, on day 3, Mary begins vomiting. Vomiting is not typically associated with the flu, so we suspect that she actually has some other illness, and therefore it is unlikely that she has had the flu on any of the days. More precisely, we want to say that if we observe that the vomiting symptom on day 3 is active (denoted $S_{3,1}$) then the probability of a mild or severe flu case on day 1 (Y_1) should decrease.

- a. **[3 points]** Describe whether each of the three networks can or cannot model this influence, and why. Specifically, is it possible to define the CPDs such that $P(Y_1 = \text{Severe} | S_{3,1} = \text{Yes}) < P(Y_1 = \text{Severe} | S_{3,1} = \text{No})$? To use the terminology of 3.1, this asks whether $S_{3,1}$ is a requisite probability node for $P(Y_1)$.
- b. **[3 points]** Repeat the previous question assuming that Y_3 is observed. That is, for each of the three models, determine if $S_{3,1}$ is a requisite node for $P(Y_1 | Y_3)$.
- c. **[3 points]** Repeat the first question assuming both Y_3 and Y_2 are observed. That is, for each of the three models, determine if $S_{3,1}$ is a requisite node for $P(Y_1 | Y_3, Y_2)$.
- d. **[3 points]** Sometimes we might want to use features that are strongly correlated. For example, suppose one symptom variable represents a “dry cough” and another variable represents any type of cough: if a person exhibits a dry cough, then by definition they are also exhibiting a cough, so if the value of the former is **yes**, the value of the latter will also be **yes**. For each of the three models, describe whether having correlated symptom variables is a violation of the model’s assumptions of independence or conditional independence.

4 Variable Elimination(15 points)

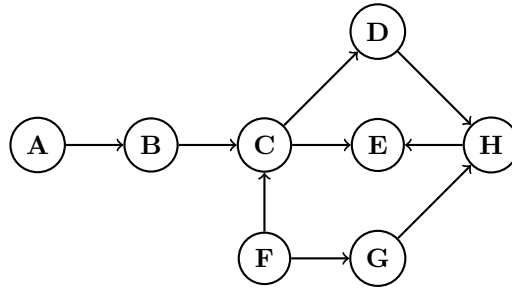


Figure 3: Graphical model for the variable elimination problem.

- a. **[10 points]** For the two variable elimination orderings given below, draw a valid clique tree. Reason that your resulting clique trees are valid. Assume $\{G,H\}$ are your query nodes.
- A-B-F-E-C-D
 - C-D-A-B-F-E
- b. **[5 points]** Now, consider the following (valid) clique tree for inference in the network shown above. Here $\{C,E,H\}$ is the root node and messages are sent upstream to $\{C,E,H\}$. Write down the distribution over the query variables in the sum-product form as defined by the message passing operations shown in this clique tree.

