Assignment 2: Bayesian Networks

CS486/686 – Spring 2017

Out: May 23, 2017
Due: June 2 (11:59 pm), 2017.

Submit an electronic copy of your assignment via LEARN. Late submissions incur a 2% penalty for every rounded up hour past the deadline. For example, an assignment submitted 5 hours and 15 min late will receive a penalty of ceiling(5.25) * 2% = 12%.

Be sure to include your name and student number with your assignment.

Every year, credit card companies lose millions of dollars due to frauds resulting from lost or stolen cards. Recently, the financial industry has turned to AI for solutions to the fraud detection problem. Intuitively, credit card holders tend to make purchases following a certain pattern. A fraud is likely to happen when this pattern is broken. In this assignment you will implement a simple fraud detection system in 3 steps. First, you will implement the variable elimination algorithm. Second, you will define your fraud detection system as a Bayesian network and compute the likelihood of a fraud in different situations. Computations in your Bayesian network should be done with your implementation of the variable elimination algorithm. However, if you are not able to complete your implementation of variable elimination, you can also do the computations by hand since they are relatively simple.

1. **[0 pts]** Implement the variable elimination algorithm in by coding the following 4 functions in Python. Factors are essentially multi-dimensional arrays. Hence use numpy multidimensional arrays as your main data structure. If you are not familiar with numpy, go through the following tutorial: https://docs.scipy.org/doc/numpy-dev/user/quickstart.html.

   (a) **restrictedFactor = restrict(factor, variable, value):** function that restricts a variable to some value in a given factor. [Tip: use slicing operations to implement this function.]

   (b) **productFactor = multiply(factor1, factor2):** function that multiplies two factors. [Tip: take advantage of numpy broadcasting rules to multiply factors with different variables. See https://docs.scipy.org/doc/numpy-dev/user/basics.broadcasting.html]

   (c) **resultFactor = sumout(factor, variable):** function that sums out a variable in a given factor. [Tip: use the sum operation to implement this function.]

   (d) **normalizedFactor = normalize(factor):** function that normalizes a factor by dividing each entry by the sum of all the entries. This is useful when the factor is a distribution (i.e. sum of the probabilities must be 1).

   (e) **resultFactor = inference(factorList, queryVariables, orderedListOfHiddenVariables, evidenceList):** function that computes $Pr(queryVariables|evidenceList)$ by variable elimination. This function should restrict the factors in factorList according to the evidence in evidenceList. Next, it should sumout the hidden variables from the product of the factors in factorList. The variables should be summed out in the order given in orderedListOfHiddenVariables. Finally, the answer should be normalized when a probability distribution that sums up to 1 is desired.

   Tip: test each function individually using simple examples from the lecture slides. If you wait till the end to test your entire program it will be much harder to debug.

What to hand in: hand in a printout of your code. Note that there are no marks given for Question 1. However, in Questions 2, part of the marks will be given for using your variable elimination algorithm to do the computations.

2. **[75 pts]** Suppose you are working for a financial institution and you are asked to implement a fraud detection system. You plan to use the following information:

   - When the card holder is traveling abroad, fraudulent transactions are more likely since tourists are prime targets for thieves. More precisely, 1% of transactions are fraudulent when the card holder is traveling, where as only 0.4% of the transactions are fraudulent when she is not traveling. On average, 5% of all transactions happen while the card holder is traveling. If a transaction is fraudulent, then the likelihood of a foreign purchase increases, unless the card holder happens to be traveling. More precisely, when the card holder is not traveling, 10% of the fraudulent transactions are foreign purchases where as only 1% of the legitimate transactions are foreign purchases. On the other hand, when the card holder is traveling, then 90% of the transactions are foreign purchases regardless of the legitimacy of the transactions.
   - Internet purchases are more likely to be fraudulent. This is especially true for card holders who don't own any computer, tablet or smart phone. Currently, 80% of the population owns a computer, tablet or smart phone and for those card holders, 10% of their legitimate transactions are done over the internet, however this percentage increases to 15% for fraudulent transactions. For those who don't own any computer, tablet or smart phone, a mere 0.1% of their legitimate transactions is done over the internet, but that number increases to 5.1% for fraudulent transactions. Unfortunately, the credit card company doesn't know whether a card holder owns a computer, tablet or smart phone, however it can usually guess by verifying whether any of the recent transactions involve the purchase of computer related accessories. In any given week, 10% of those who own a computer, tablet or smart phone purchase (with their credit card) at least one computer related item as opposed to just 1% of those who don't own any computer, tablet or smart phone.

   (a) **[30 pts]** Construct a Bayes Network to identify fraudulent transactions.

   Tip: The probabilities given in the text below can be inserted directly into corresponding conditional probability tables (i.e., no computation needed to obtain the conditional probability tables).

   What to hand in: Show the graph defining the network and the conditional probability tables associated with each node in the graph. This network should encode the information stated above. Your network should contain exactly six nodes, corresponding to the following binary random variables:

   - OC – card holder owns a computer, tablet or smart phone.
   - $Fraud$ – current transaction is fraudulent.
   - $Trav$ – card holder is currently traveling.
   - FP – current transaction is a foreign purchase.
   - IP – current purchase is an internet purchase.
   - CRP – a computer related purchase was made in the past week.

   The arcs defining your Bayes Network should accurately capture the probabilistic dependencies between these variables.

   (b) **[15 pts]** What is the prior probability (i.e., before we search for previous computer related purchases and before we verify whether it is a foreign and/or an internet purchase) that the current transaction is a fraud? What is the probability that the current transaction is a fraud once we have verified that it is a foreign transaction, but not an internet purchase and that the card holder purchased computer related accessories in the past week?

   What to hand in: Indicate what queries (i.e., $Pr(variables|evidence)$) you used to compute those probabilities. Whether you answer the queries by hand or using the code you wrote for Question 1, provide a printout of the factors computed at each step of variable elimination (as done in the lecture slides). Use the following variable ordering when summing out variables in variable elimination: $Trav$, FP, $Fraud$,

IP, OC, CRP. Note that a maximum of two thirds of the marks are earned if you answer correctly the question by doing the computations by hand instead of using your program.

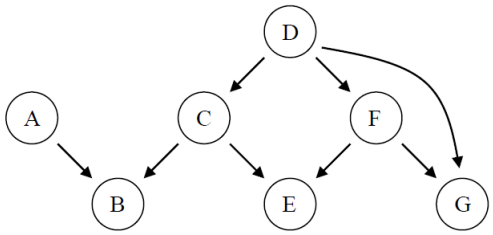
(c) **[15 pts]** After computing those probabilities, the fraud detection system raises a flag and recommends that the card holder be called to confirm the transaction. An agent calls at the domicile of the card holder but she is not home. Her spouse confirms that she is currently out of town on a business trip. How does the probability of a fraud changes based on this new piece of information?

What to hand in: Same as for Question 2b.

(d) **[15 pts]** Suppose you are not a very honest employee and you just stole a credit card. You know that the fraud detection system uses the Bayes net designed earlier, but you still want to make an important purchase over the internet. What can you do prior to your internet purchase to reduce the risk that the transaction will be rejected as a possible fraud?

What to hand in: Tell me the action taken and indicate by how much the probability of a fraud gets reduced. Follow the same instructions as for Question 2b.

3. **[25 pts]** You have just been hired as a consultant for a car manufacturer. The company would like to improve their maintenance service at its dealerships by assisting the mechanics with an automated fault diagnosis tool. After talking with several experts, youve built the following Bayesian network. Each node is a Boolean variable that corresponds to the status (e.g., working or not) of a car component and each edge indicates a probabilistic dependency for failure.



(a) **[14 pts]** Suppose a mechanics would like to know what components may influence the functioning of other components. Answer the following questions and give a brief justification based on the D-separation rules.

i. Are D and G independent? n
ii. Are D and G independent given F? n
iii. Are A and G independent? y
iv. Are A and G independent given B? n
v. Are A and G independent given B and C? y
vi. Are A and G independent given B and D? y
vii. Are A and G independent given B, D and E? n

(b) **[11 pts]** Suppose a mechanics would like to know $\Pr(C|A = true, E = false)$. What is the subset of relevant variables that is sufficient to answer this query? Give a brief justification based on the rules to identify relevant variables. C D E F