CS 161: Fundamentals of Artificial Intelligence

Prof. Darwiche

Spring 2023 – Assignment 8 – Due 11:59pm, Tuesday, MAY 30

Please submit your solutions on Bruinlearn. Submitted file should be a formatted PDF **hw8.pdf**. In addition to your solutions file, you will need to submit two **.net** files as indicated below. All modeling, learning, sensitivity analysis and inference for this homework should be done using SamIam. You can download SamIam from http://reasoning.cs.ucla.edu/samiam. You can find video tutorials for using Samiam from http://reasoning.cs.ucla.edu/samiam/index.php?h=videos. The principal part of SamIam is pure Java, and therefore will run on any system with a Java Runtime Environment, regardless of architecture.

Here are some tips for installing Java Runtime Environment:

1. Enter https://www.java.com/en/download/manual.jsp and select a version of Java to download.
2. Follow instructions in to setup the local environment https://www.tutorialspoint.com/java/java_environment_setup.htm. Don't forget to set your Environment variable PATH to point to the Java executable (on windows e.g. `C:\ProgramFiles\Java\jdk-20\bin`).

Here are some tips for installing SamIam:

1. Enter http://reasoning.cs.ucla.edu/samiam,
2. Click 'download',
3. Enter your personal information,
4. Select 'SamIam Release',
5. For Windows and Linux, select 'Windows i386' or 'Linux i386' if you use a 32-bit system; select 'Windows amd64' or 'Linux amd64' if you use a 64-bit system. For MacOS, select 'MAC OS X i386'.
6. For Mac and Linux users: you will be able to directly launch the Samiam GUI. For windows users: you need to go to Samiam/samiam.bat and edit the last line by adding a pair of quotes around `%~$PATH:1`. Then you will be able to launch the Samiam GUI from samiam.bat.

   Note: We have a beta-version of Samiam with more advanced functionalities, which you are welcome to download and use from https://github.com/kenmueller/SamIam/releases. For questions related to this release, please contact Scott Mueller scott@cs.ucla.edu or Piazza.

1 Question 1

Consider the following problem which was discussed in class:

> Suppose that we have a patient who was just tested for a particular disease and the test came out positive. We know that one in every thousand people has this disease. We also know that the test is not reliable: it has a false positive rate of 2% and a false negative rate of 5%. Our goal is then to assess our belief in the patient having the disease given that the test came out positive. If we let the propositional variable D stand for the patient has the disease, and the propositional variable T stand for the test came out positive, our goal is then to compute $\Pr(D|T)$.

You may also recall being surprised that $\Pr(D|T) \approx 0.045$. The goal of this question is then to identify conditions under which this probability will be no less than .30. You will need to find the answer to this by constructing a Bayesian Network and using the **sensitivity analysis** engine of SamIam. You need to turn in **test.net** and contain the following Information in hw8.pdf :

- Your complete Bayesian network (Structure and CPTs) in test.net file.
- A constraint on each of the following, which is sufficient to ensure that $\Pr(D|T) \geq 0.3$: The prior probability of having the disease, the false positive for the test, and the false negative for the test. Screenshot the results from SamIam and attach the pictures in the report.

Hint: you need to choose an algorithm like sheony-shafer to run the sensitivity analysis.

2 Question 2

Consider the following scenario:

> When Sambot goes home at night, he wants to know if his family is home before he tries the doors. (Perhaps the most convenient door to enter is double locked when nobody is home). Often when Sambot's wife leaves the house she turns on an outdoor light. However, she sometimes turns on this light if she is expecting a guest. Also, Sambot's family has a dog. When nobody is home, the dog is put in the back yard. The same is true if the dog has bowel trouble. Finally, if the dog is in the backyard, Sambot will probably hear her barking, but sometimes he can be confused by other dogs barking. Sambot is equipped with two sensors: a light sensor for detecting outdoor lights and a sound sensor for detecting the barking of dogs(s). Both of these sensors are not completely reliable and can break. Moreover, they both require Sambot's battery to be in good condition.

Your task is to build a bayesian network that Sambot will use to reason about the above situation using the modeling and inference tool SamIam. Specifically, given sensory input, Sambot needs to compute his beliefs in various events: whether his family is home, whether any of his sensors are broken, whether the dog is in the backyard, and whether it has bowel trouble. You need to proceed as follows:

(a) Decide on the set of variables and their values. These variables and values must match those in the given data file sambot.dat.

(b) Construct the causal structure.

(c) Learn the network CPTs using the EM algorithm and the data file (sambot.dat) provided in the assignment. Your initial network should have uniform parameters.

You need to turn in **sambot.net** and contain the following Information in hw8.pdf:

- The most likely instantiation of all variables given that Sambot has sensed the lights to be on, but has sensed no bark. Explain how you obtained this answer (for partial credit in case you get the wrong answer). Screenshot the results from SamIam and attach the pictures in the report.
- The most likely instantiation of the sensors given that the family is home and no guests are expected. Explain how you obtained this answer (for partial credit in case you get the wrong answer). Screenshot the results from SamIam and attach the pictures in the report.
- The smallest set of variables Z in your network such that the two sensors are independent given Z. Justify your answer based on d-separation.

- The type of network you constructed: tree, polytree (singly-connected network), or multiply-connected network.

Hint: you need to choose an algorithm like sheony-shafer to run the EM algorithm

Submission

- Submit all solution files **on bruinlearn**.
- Submit a pdf **hw8.pdf** containing your solution and two **.net** files.