

## ASSIGNMENT 4: MEDICAL DIAGNOSIS

**Goal:** The goal of this assignment is to get experience with Bayesian Networks and understand their value in the real world.

**Scenario:** Medical diagnosis. Some medical researchers have created a Bayesian network that models the inter-relationship between (some) diseases and observed symptoms. Our job as computer scientists is to write a diagnosis software for them. Based on the symptoms of a new patient our software will advise the doctors about the likely diseases and the tests to conduct.

**Problem Statement:** We are given the Bayesian Network created by the researchers. The network is here:

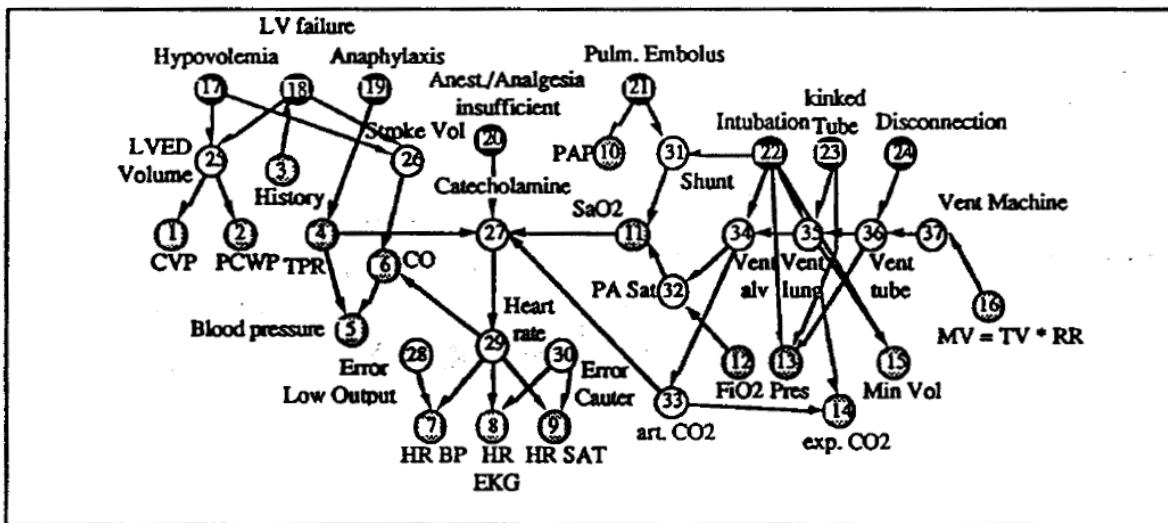


Fig. 1 The ALARM network representing causal relationships is shown with diagnostic (●), intermediate (○) and measurement (⊙) nodes. CO: cardiac output, CVP: central venous pressure, LVED volume: left ventricular end-diastolic volume, LV failure: left ventricular failure, MV: minute ventilation, PA Sat: pulmonary artery oxygen saturation, PAP: pulmonary artery pressure, PCWP: pulmonary capillary wedge pressure, Pres: breathing pressure, RR: respiratory rate, TPR: total peripheral resistance, TV: tidal volume

Notice that eight diagnoses are modeled here: hypovolemia, left ventricular failure, Anaphylaxis, insufficient analgesia, pulmonary embolus, intubation, kinked tube, and disconnection. The observable nodes are CVP, PCWP, History, TPR, Blood Pressure, CO, HR BP, HR EKG, HR SAT, SaO2, PAP, MV, Min Vol, Exp CO2, FiO2 and Pres.

Such networks can be represented in many formats. We will use the .bif format. BIF stands for Bayesian Interchange Format. The details about the format are [here](#). We are also providing a .bif parser so that you can start directly from a parsed Bayesian network represented as a graph.

Your goal is to compute the probability of each diagnosis given the observations. You will implement any one Bayes net inference algorithm of your choice. Our recommendation is to use either variable elimination or Gibbs sampling. After that you will input a sample observation file and output the probability of each of the eight diagnoses.

### **Input format:**

We will work with [alarm.bif](#) network. Please have a look at this file to get a basic understanding of how this information relates to the Bayes net image above.

A sample input file for a patient is as follows: in each line it has a variable in quotes followed by its value in quotes. The file ends with a full stop. For example, the following represents a patient with high HREKG, zero EXPCO2 and normal CVP. We don't know anything else about this patient.

"HREKG" = "High"

"ExpCO2" = "Zero"

"CVP" = "Normal"

.

### **Output format:**

Your output format will be diagnosis followed by probability distribution in the same order as in input file. It will have eight lines one for each diagnosis. Five decimal places are sufficient.

"Hypovolemia" = { 0.44311 0.55689 }

"Intubation" = { 0.33 0.01 0.66 }

....

For example, up above it says that Intubation equals normal, esophageal and onesided have probabilities 0.33, 0.01 and 0.66 respectively. This order is because this is the order in which the variable values are prescribed in the original .bif network.

### **Code**

We are providing the sample code of inputting a Bayes net in .bif format and parsing it into a data structure representing the Bayesian Network. The zipped code folder contains 5 files: alarm.bif (the ALARM network) , hw5.bif (the network from the last problem of hw5) , DJ.java, InferenceGraph.java, InferenceGraphNode.java, README.txt and BayesNets.jar. BayesNets.jar contains a java library that can read a bif file into a nice data structure.

The file DJ.java provides basic code that reads the alarm.bif file into a data structure and prints it to standard output. To understand the data structure, look at README.txt and DJ.java first. Once you understand those, you can look at the complete API in InferenceGraph.java, InferenceGraphNode.java— Do not use these 2 files in your code, they are already part of the jar file.

You can link your java code with BayesNets.jar as follows on a linux command line:

```
javac -cp BayesNets.jar DJ.java (Compiling)
```

```
java -cp .:BayesNets.jar DJ (Running)
```

If you use eclipse, you can simply add BayesNets.jar to your project.

You can use this basic data structure to write the inference method. And, you would write the top level I/O routine that takes a new patient and outputs the diagnoses probabilities. A sample input.txt file is in the code.

We will likely not run any of your code for this assignment. After the submission deadline, we will create a form where you will submit diagnosis probabilities given sample patient files (which we will provide) and we will check your diagnosis probabilities against those computed by our inference code.

## **Writeup**

Your writeup (max 2 pages) should discuss your choice of algorithm and its reason. You should discuss any observations regarding correct implementation, what worked and what didn't.

You should also discuss the answer to the following question: if the doctor could prescribe a test to measure one of the observable variables, which test should he prescribe? You need not write code to answer this question. But you need to describe your method (and relevant equations). Assume that all tests are equally expensive and measure exactly one observable variable and give 100% accurate measurements.

## **Testing your code**

Since it is nearing the quarter you will help us test your code. A day after the submission date we will release a set of sample patient files, and you will have to give us diagnosis probabilities.

### **What to submit?**

1. Submit a zipped folder with your writeup named "your UW net id".pdf and another zipped folder (Code.zip) with all your code. You do not need to provide a compilation or running script, we will only look at it to make sure the code is sensible and consistent with your results in (2) below. You will be penalized for any submissions that do not conform to this requirement.
2. A day after the deadline, run your code on the sample patient files we release and submit your results through a catalyst form we will create.

### **Evaluation Criteria**

1. [5 points] Your writeup.
2. [6 points] Soundness of your codes (based on computing correct diagnosis probabilities for the sample patient files we will release and the code you submit).
3. [1.5 points] Discretionary points based on the writeup and code.

### **What is allowed? What is not?**

1. You have to do this assignment by yourself. No teams are allowed.
2. We prefer you build on top of the given code and hence use Java programming language. If you want to use a different programming language you will have to write your own parser.
3. You are not allowed to read other codes or test against other softwares for Bayesian networks. You are welcome to do Google search to understand general Bayes net algorithms.
4. You must not discuss this assignment with anyone outside the class. Make sure you mention the names in case you discuss with anyone in the class outside your team.
5. Your code will be automatically evaluated. There will be negative penalty if your output is not automatically parsable and the TA has to go back and forth with your team to get your code/output working. Make sure it satisfies the format checker before you submit.