CS 676 Machine Learning Homework 1B

March 20, 2015

1 Problem Set 1b (115 Points)

1.1 Inference in a Simple Model [50 points]

We implement this simple bayes network by python. It can complete this three node graphical model's probability calculation perfectly, and some of n nodes bayes network probablity calculation. Please notice the following things:

- The roots (nodes without any parent) are also listed in the cpd-simple.txt,cpd-extended.txt. The form is NameOfRoot None 0.3
  For example, in cpd-simple.txt, FluRate=Elevated None 0.5 means FluRate is a root without any parents, P(FluRate=Elevated)=0.5
- Our code can correctly give any form conditional probablities, marginal probablities, joint probablity of this 3 -node-graphical model. Like P(MaryGetsFlu), P(MaryGetsFlu,RoommateHasFlu), P(MaryGetsFlu,RoommateHasFlu,FluRate),P(MaryGetsFlu,RoommateHasFlu‖ FluRate), P(MaryGetsFlu‖ RoommateHasFlu,FluRate),etc.
- If the inputting data is wrong, like inputting ./bayes-query network-simple.txt cpd-simple.txt MaryGetsFlu=Impossible FluRate=0.9, where Impossible,and 0.9 are not the status of MaryGetsFlu,and FluRate, our code may still give a answer, but this answer must be wrong.
- Our code can run well on my mac, if it can not run in the linux system of gradLab, please let us know.

1.2 Designing an Extended Model [30 points]

In the extended model, we have seven variables and corresponding values they can take in the parentheses respectively as follows:

$$
\begin{aligned}
&FluRate(Mild, Moderate, Severe)\\
&MaryGetsFlu(Yes, No)\\
&RoommateHasFlu(Yes, No)\\
&CoworkerHasFlu(Yes, No)\\
&MaryIsVaccinated(Yes, No)\\
&IsFluSeason(Yes, No)\\
&PreviousFluRate(Mild, Moderate, Severe)
\end{aligned}
$$

The directed edges between variables in the network are as follows:

$$
\begin{aligned}
IsFluSeason &\rightarrow PreviousFluRate \\
IsFluSeason &\rightarrow FluRate \\
PreviousFluRate &\rightarrow Flurate \\
FluRate &\rightarrow RoommateHasFlu \\
FluRate &\rightarrow CoworkerHasFlu \\
RoommateHasFlu &\rightarrow MaryGetsFlu \\
CoworkerHasFlu &\rightarrow MaryGetsFlu \\
MaryIsVaccinated &\rightarrow MaryGetsFLu
\end{aligned}
$$

There are multiple sets of qualified conditional probability tables satisfying the eight intuitions. Here, we added another assumpting that if mary is vaccinated, the probability of her getting flu will reduce dramatically regardless of the status of her roommate and coworker. Hence, we picked up the following set.

IsFluSeason=Y	IsFluSeason=N
0.5	0.5

MaryIsVaccinated=Y	MaryIsVaccinated=N
0.5	0.5

	PreviousFluRate=Mild	PreviousFluRate=Moderate	PreviousFluRate=Severe
IsFluSeason=Yes	0.2	0.5	0.3
IsFluSeason=No	0.7	0.2	0.1

	FluRate=Mild	FluRate=Moderate	FluRate=Severe
IsFluSeason=Yes,PreviousFluRate=Mild	0.48	0.36	0.16
IsFluSeason=Yes,PreviousFluRate=Moderate	0.1	0.7	0.2
IsFluSeason=Yes,PreviousFluRate=Severe	0.07	0.38	0.55
IsFluSeason=No,PreviousFluRate=Mild	0.8	0.15	0.05
IsFluSeason=No,PreviousFluRate=Moderate	0.32	0.6	0.08
IsFluSeason=No,PreviousFluRate=Severe	0.3	0.25	0.45

	MaryGetsFlu=Yes	MaryGetsFlu=No
CoworkerHasFlu=Yes,RoommateHasFlu=Yes,MaryIsVaccinated=Yes	0.45	0.55
CoworkerHasFlu=No,RoommateHasFlu=Yes,MaryIsVaccinated=Yes	0.35	0.65
CoworkerHasFlu=Yes,RoommateHasFlu=No,MaryIsVaccinated=Yes	0.3	0.7
CoworkerHasFlu=Yes,RoommateHasFlu=Yes,MaryIsVaccinated=No	0.9	0.1
CoworkerHasFlu=No,RoommateHasFlu=No,MaryIsVaccinated=Yes	0.05	0.95
CoworkerHasFlu=No,RoommateHasFlu=Yes,MaryIsVaccinated=No	0.6	0.4
CoworkerHasFlu=Yes,RoommateHasFlu=No,MaryIsVaccinated=No	0.7	0.3
CoworkerHasFlu=No,RoommateHasFlu=No,MaryIsVaccinated=No	0.15	0.85

1.3 Parameter Estimation [35 points]

a. 5 points: What are the parameters of this network?

Suppose this is a Bayes Network. There exists 5 nodes totally. For node A, since A has no parents,and it has two states, therefore, we can use 2 parameters:

$$P(A=0) = \theta_{A0} P(A=1) = \theta_{A1} = 1 - \theta_{A0}$$

For node B, since B has node A as its unique parent, and B has two states, then we can use 4 parameters:

$$P(B=0|A=0)=\theta_{B0|A0}, P(B=1|A=0)=\theta_{B1|A0}=1-\theta_{B0|A0}$$

$$P(B=0|A=1)=\theta_{B0|A1}, P(B=1|A=1)=\theta_{B1|A1}=1-\theta_{B0|A1}$$

Similarly for node C, it has A,B as its parents, and C has two states, then we can use 8 parameters:

$$P(C=0|B=0,A=0)=\theta_{C0|B0,A0}, P(C=1|B=0,A=0)=\theta_{C1|B0,A0}=1-\theta_{C0|B0,A0}$$

$$P(C=0|B=1,A=0)=\theta_{C0|B1,A0}, P(C=1|B=1,A=0)=\theta_{C1|B1,A0}=1-\theta_{C0|B1,A0}$$

$$P(C=0|B=0,A=1)=\theta_{C0|B0,A1}, P(C=1|B=0,A=1)=\theta_{C1|B0,A1}=1-\theta_{C0|B0,A1}$$

$$P(C=0|B=1,A=1)=\theta_{C0|B1,A1}, P(C=1|B=1,A=1)=\theta_{C1|B1,A1}=1-\theta_{C0|B1,A1}$$

Then we can repeat such methods to get the parameters of D,E. In general, we need to 2+4+8+8+4=26 parameters to present this graphical model. But since the states of each node are binary, therefore, we can use 1+2+4+4+2=13 parameters to represent the 26 parameters.

> **b. 15 points:** Using pseudocode, design an algorithm for learning the network parameters. You may abstract away any procedures that you would like, but it must be very clear to us that you know and fully understand what that procedure does. Along with your pseudocode, provide a list of descriptions for each abstracted procedure you use (this is where you should demonstrate your understanding). You cannot, for example, use one line of code that calls a function learn() . If you feel like you are using a function that is too high-level, email us to confirm.

```
# Nodes are A, B, C, D and E

# Data size M =20

#Initialize \theta_0 for some number \in (0,1)

    parameter.initialize()

#Setup a threshold for convergence

     error=c  #c is a constant number, like 10^-6

#Stop the iteration till |\theta_k+1-\theta_k| <= \epsilon

     def isConverge(error, parameter):
            if (Theta[t+1]-Theta[t])*(Theta[t+1]-Theta[t])<error:
                  return True
            return False
#find the missing data

    missingData=method.miss(Dataset)

    While(isConverge(error, parameter)==False):

     #E-step
            #calculate probability by using  the observed data, and current parameter
            calculateProb(parameter,dataset)
```

```
        #count the various condition of various nodes

        for node in Node:
              #if node doesn't have any parent
              if len(node.parent)==0:
                     if node.condition==x1:
                           count[node][node.condition=x1]+=1
              #if node has a parent
              elif len(node.parent)==1:
                    if node.condition==x1 and parent.condition=x2:
                           count[node][node.condition=x1,parent.condition=x]+=1
               #if node has more than one parent
               elif len(node.parent)>1:
                     if node.condition==x1:
                            if condition(parent)==x5:
                                  count[node][node.condition=x1,parent.condition]+=1

        #calculate the Likelihood function by all the data,and current parameter

  #M-step
         #if each node has 0,1 states, it is binary
           for theta in Parameter:
                 #if node has no parent
                 theta[i][condition=0]=count[i][condition=0]/count[i]
                 theta[i][condition=1]=1-theta[i][condition=0]
                 #if nodes has parents
                 theta[i][condition=0,parent.condition=x]=count[i][cond=0,parent.cond=x]
                                                                   /count[i][parent.cond=x]
                 theta[i][condition=1,parent.condition=y]=1-theta[i][cond=0,parent.cond=y]

 #Go to While Function to do the next loop

#if we have obtained optimal parameters by such EM step,then we need to output them
output(result)
```

Deveriable. Model construction and Analysis

Based on Figure 2, we have five random variables which construct as our graphical model whose node set is $N = \{A, B, C, D, E\}$ by the restriction from this graphic model, we have $A,\ B|A,\ C|AB,\ D|B,\ E|BC$ are conditional independent random variables. And let $O_N = \{O_A, O_B, O_C, O_D, O_E\}$ as their observability variables. Since each node is a binary random variable, then we have the likelihood is

$$L(\theta, \psi : \mathcal{D}) = \theta^{M[1]}(1-\theta)^{M[0]}\psi^{M[1]+M[0]}(1-\psi)^{m[?]} \tag{1}$$

As we expect, the likelihood function in this example is a product of two functions: a function of θ, and a function of ψ. We can easily see that the maximum likelihood estimate of θ is $\frac{M[1]}{M[1]+M[0]}$, while the maximum likelihood of ψ is $\frac{M[1]+M[0]}{M[1]+M[0]+M[?]}$.
Then clearly the posterior also factorizes:

$$P(\theta|\mathcal{D}) \propto P(\mathcal{D}|\theta)P(\theta) = L(\theta, \psi : \mathcal{D})P(\theta) \tag{2}$$

the θ here could be $\theta_{X|\mathrm{pa}(X)}$ in which X could be any node of graphical model Figure 2. Formally, we can write the tth CPT as $x_t|X_{\mathrm{pa}(t)} = c \sim \mathrm{Cat}(\theta_{tc})$, where $\theta_{tck} := P(x_i = k|X_{\mathrm{pa}(t)} = c)$, for $k = 1 : K_t, c = 1 : C_t$

and $t = 1 : T$. Here K_t is the number of states for node $t, C_t := \prod_{s \in \mathrm{pa}(t)} K_s$ is the number of parent combinations, and T is the number of nodes. Obviously $\sum_k \theta_{tck} = 1$ for each row of each CPT.
Let us put a separate Dirichlet prior on each row of each CPT, i.e., $\theta_{tc} \sim \mathrm{Dir}(\alpha_{tc})$. Then we can compute the posterior by simply adding the pseudo counts to the empirical counts to get $\theta_{tc}|\mathcal{D} \sim \mathrm{Dir}(N_{tc} + \alpha_{tc})$, where N_{tck} is the number of times that node t is in state k while its parents are in state c:

$$N_{tck} :=\sim_{i=1}^{N} \mathbb{I}(x_{i,t} = k, x_{i,\mathrm{pa}(t)} = c) \tag{3}$$

the mean of this distribution is given by the following:

$$\bar{\theta_{tck}} = \frac{N_{tck} + \alpha_{tck}}{\sum_{k'}(N_{tck'} + \alpha_{tck'})} \tag{4}$$

Then here is the table of probabilities of five independent conditional random variable, in which note that $N_{tck=0}$ means the number of times that node t is in state 0 while its parents are in state c, similar as $N_{tck=1}$:
For independent random variable A, we have:

$N_{tck=0}$	$N_{tck=1}$	$\theta_{A=0}$	$\theta_{A=1}$
6	8	7/16	9/16

For independent random variable $B|A$, we have:

	$N_{tck=0}$	$N_{tck=1}$	$\theta_{B=0}$	$\theta_{B=1}$
A = 0	1	5	1/4	3/4
A = 1	5	1	3/4	1/4

For independent random variable $C|AB$, we have:

	$N_{tck=0}$	$N_{tck=1}$	$\theta_{C=0}$	$\theta_{C=1}$
(A,B) = (0,0)	0	1	1/3	2/3
(A,B) = (0,1)	2	1	3/5	2/5
(A,B) = (1,0)	3	2	4/7	3/7
(A,B) = (1,1)	1	0	2/3	1/3

For independent random variable $D|B$, we have:

	$N_{tck=0}$	$N_{tck=1}$	$\theta_{D=0}$	$\theta_{D=1}$
B = 0	2	3	3/7	4/7
B = 1	6	0	7/8	1/8

For independent random variable $E|BC$, we have:

	$N_{tck=0}$	$N_{tck=1}$	$\theta_{E=0}$	$\theta_{E=1}$
(B,C) = (0,0)	0	3	1/5	4/5
(B,C) = (0,1)	2	1	3/5	2/5
(B,C) = (1,0)	0	4	1/6	5/6
(B,C) = (1,1)	0	1	1/3	2/3