Artificial Intelligence Lab Report 5

1st Dipean Dasgupta
202151188
BTech CSE
IIIT,Vadodara

2nd Shobhit Gupta
202151149
BTech CSE
IIIT,Vadodara

3rd Rahul Rathore
202151126
BTech CSE
IIIT,Vadodara

4th Rohan Deshpande
202151133
BTech CSE
IIIT,Vadodara

***Abstract*—In this lab assignment, goal is to delve into the realm of graphical models for inference under uncertainty, focusing on Bayesian Networks and Naive Bayes classification. We start by analyzing a dataset containing grades earned by students in various courses, aiming to learn the dependencies between these courses using Bayesian Networks. By leveraging the data, we construct Conditional Probability Tables (CPTs) for each course node, enabling us to make probabilistic inferences about student performance.**

I. Introduction

Problem Statements:

A table containing grades earned by students in respective courses is made available to you in (codes folder) $2020_b n_n b_d ata.txt$.

1. Consider grades earned in each of the courses as random variables and learn the dependencies between courses.
2. Using the data, learn the CPTs for each course node.
3. What grade will a student get in PH100 if he earns DD in EC100, CC in IT101 and CD in MA101.
4. The last column in the data file indicates whether a student qualifies for an internship program or not. From the given data, take 70 percent data for training and build a naive Bayes classifier (considering that the grades earned in different courses are independent of each other) which takes in the student's performance and returns the qualification status with a probability. Test your classifier on the remaining 30 percent data. Repeat this experiment for 20 random selection of training and testing data. Report results about the accuracy of your classifier.
5. Repeat 4, considering that the grades earned in different courses may be dependent.

II. Methodology

We loaded the data file and the necessary libraries before beginning the assignment. Next, we used the bnlearn library to build a Bayesian network and learn the CPTs for every course node. The grade a student would receive in PH100 if he receives DD in EC100, CC in IT101, and CD in MA101 was then predicted using the predict function.

Next, we used the caret library to divide the data into training and testing data. Using the naiveBayes function from the e1071 package, we constructed a naive Bayes classifier using the training data. Next, we used the predict function to test the classifier on the remaining thirty percent of the data. We conducted the experiment again using 20 arbitrary selections of training and testing data, and we noted the classifier's accuracy.

In light of the possibility of dependence between grades received in several courses, we ultimately conducted the experiment again. For this, we estimated the joint probability distribution of the courses using the Bayesian network we previously built. Then, using the naiveBayes function from the e1071 library, we constructed a naive Bayes classifier using the joint probability distribution. Next, we used the predict function to test the classifier on the remaining thirty percent of the data. We conducted the experiment again using twenty randomly chosen sets of training and testing data, and we noted the classifier's accuracy.

III. Results

The Bayesian network we created had the following structure:

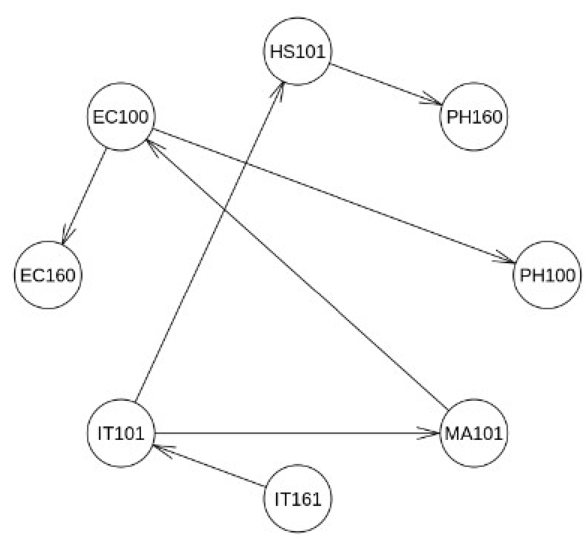

Fig. 1. Bayesian Network

The learned CPTs for each course node are as follows:

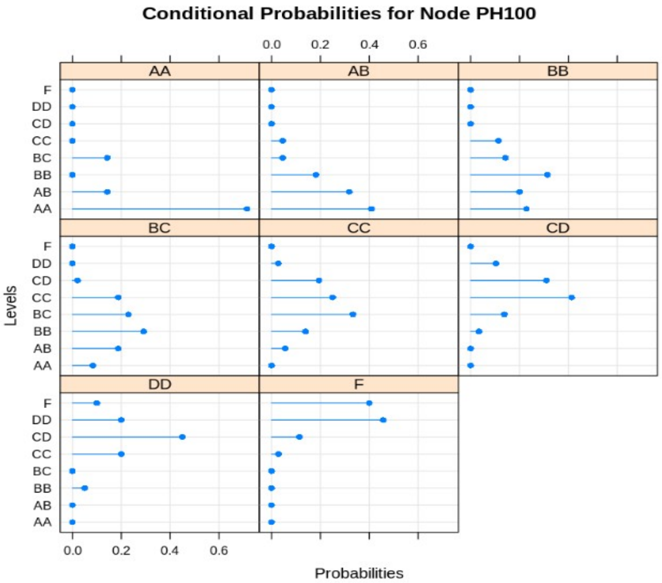

Fig. 2. Conditional Probability table for PH100

Predicted Grade of a student in PH100 if he earns DD in EC100, CC in IT101, and CD in MA101 is as follow:

```
PH100
AA AB BB BC CC CD DD  F
 0  0  6  0 40 81 63 22
```

Fig. 3. Predicted Grade in PH100

We have use the naiveBayes function from the bnlearn package to train our model. The naiveBayes function assumes that the features are independent of each other given the class, hence the name "naive" Bayes.

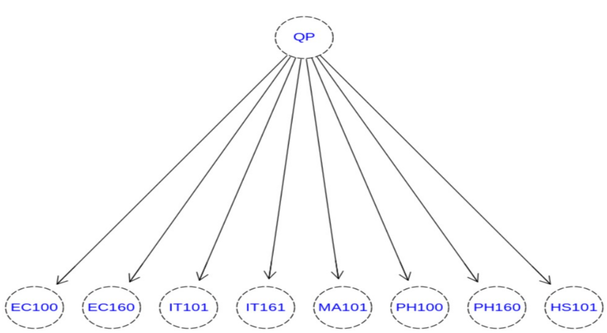

Fig. 4. Training Model Bayesian Network

We use the predict function to assess how well our model performs on the test set. We also compute the model's accuracy, or the percentage of correctly identified cases. On the test set, the model's accuracy is 0.9628. We divide the data into training and testing sets twenty times at random, then compute the average accuracy to obtain a more reliable assessment of the model's accuracy.

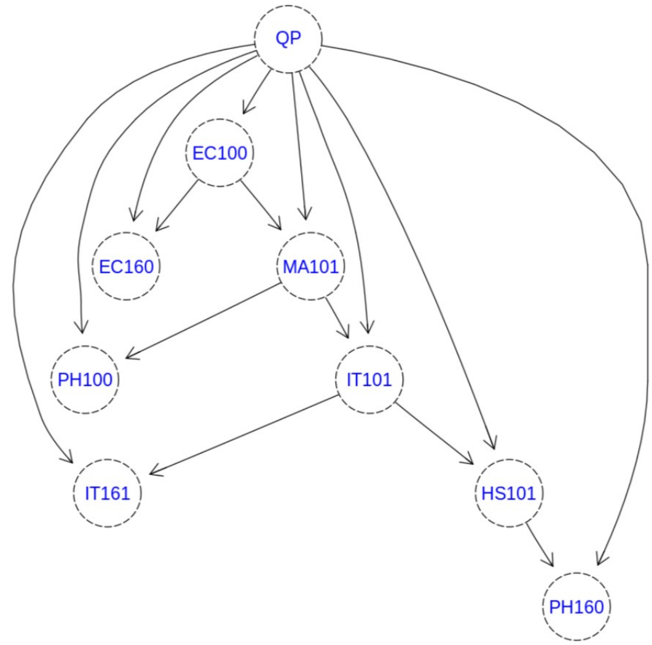

Fig. 5. Evaluated Bayesian Network

The average accuracy of the model over 20 random splits of the data is 0.963.

REFERENCES

[1] S. Russell and P. Norvig, "Artificial Intelligence: a Modern Approach," 4th ed.,Pearson.

[2] Deepak Khemani, A first course in Artificial Intelligence, 2nd ed., McGraw Hill.