ECE356 - Database Systems
Lab 4 - Data Mining

March 19, 2019

Objective: Using the dataset provided, apply the data mining classification technique of Decision Trees to predict a specific output with a computing tool such as MATLAB. In addition, practice some critical thinking skills related to justifying the outputs you get from changing certain parts of your classifier.

Dataset: This is the Sean Lahman baseball dataset. This set encompasses players who are nominated in the hall of fame and players who have been elected to be in the hall of fame.

Assume that the classification labels are nominated and elected.

Part 1: MATLAB

The first part of the lab requires you to apply a data classification technique, i.e. Decision Trees using MATLAB[1]. For this task, you need the input datasets (the description for which follows) and by using that data, you should be able to program in MATLAB to get the result.

For any classification algorithm, we need to select the input features in a process called feature selection to make the machine learn and build the model. Since we are dealing with Lahman database, the first part of this lab requires that you extract the necessary dataset to classify players into nominated or elected into the hall of fame. In the second part, you need to create a decision tree classifier using MATLAB to classify the extracted data. To validate your decision tree algorithm, divide the dataset into two separate parts: training set (80%) and the testing set (20%). The training set is the data that is used as input into an algorithm to make the machine learn and build a model. Once the model is developed, another set of elements called the testing set is used to evaluate the behavior of the model in order to predict its accuracy. One way for evaluating the model is by comparing the original classification labels of the data set with the ones that you will predict using the model.

*Please note that other ways such as are confusion matrices, recall, precision, and/or F1 score may be used to validate your decision tree algorithm. Please feel free to give these methods a try and examine their output.

1. **Input:**

   For the purpose of this lab, you should first extract the dataset from the Lahman database using the appropriate MySQL query. This dataset should be saved into a CSV file that contains the following columns: $playerID, feature1, feature2, ..., featureN, classification$. The extracted CSV file should be converted into MATLAB structures so they can be used as inputs to the decision tree algorithm.

   $Hint$: Check Batting, Pitching, and any possible table for necessary features that qualify a player to be nominated or elected.

2. **Task:** Given the 80/20 split, divide the dataset into training and testing samples. You are supposed to convert the given dataset into tabular format for MATLAB, as explained in Input section. Then build a decision tree classifier using MATLAB to predict whether a player has been nominated or elected. You should experiment

[1] If you want to use Python, please contact the teaching team.

with different impurity measures introduced in class: Entropy, and Gini's index. For each impurity measure, report the accuracy of your model for 5 randomly selected 20% testing datasets.

MATLAB functions provided at the end of Part 1 can be used to train and test your classifier.

3. **Output:** Run the algorithm for each impurity measure to calculate the classification accuracy. In addition to the analysis report, your submission must contain a CSV file named: **g_number_DT_<metric>.csv** for accuracy of your model, where, g_number refers to your lab group number, and metric is the corresponding impurity measure. So, for example, if your group number is 14, the chosen metric was gini, your file will be called: g14_DT_gini.csv. This file should have two columns with headings: Dataset number, Accuracy in its first sheet, and three columns with headings: Iteration, Classification, Predictions in its second sheet.

   - **Dataset number** is an additional column that you add to indicate the number of the iteration, starting from 1.
   - **Accuracy** Your testing accuracy for this particular iteration.
   - **Classification** are the actual labels from the classification column of the testing dataset, as described in the Input section.
   - **Prediction** will have the predicted labels calculated from your program.

   Your program must create this CSV file. You will need to submit the CSV file and the MATLAB code as part of your deliverables.

   If there is no code in the program for creating these files, no credit will be given to this submitted file.

   Your program will be evaluated based on your analysis and justifications of your output using different impurity measures. Therefore, please make sure that you provide analytic observations and solid justifications to your selected features.

MATLAB functions helpful for the algorithm are as follows:

Table 1: MATLAB functions that are helpful for Part 1

MATLAB function	Description
MODEL = FITCTREE (TBL, Y)	This function is used to train the Decision Trees model in MATLAB. It needs two arguments: TBL refers to the data with the actual data values, and Y are the classification labels, provided as a part of the training dataset. It returns a Decision Tree model that can be used when testing the algorithm. For further details, you can refer to the MathWorks® website: https://www.mathworks.com/help/stats/fitctree.html.
YP = PREDICT (MODEL, DATA)	This MATLAB function can be used to predict the output of a trained model. The two arguments are the MODEL which is the model you would have achieved while training the data, and DATA refers to the test data on which the model needs to run. The ground truth of the testing sample is thus only used for evaluation purposes when you will be calculating accuracy of the results. For further details, you can refer to the MathWorks® website: https://www.mathworks.com/help/stats/compactclassificationtree.predict.html.

Some other useful MATLAB functions that may help you in coding are as follows:

Table 2: MATLAB functions that may help you in Part 1

MATLAB function	Description
loss	Calculates the classification error of the model generated
confusionmat	Indicates how many labels were correctly classified and how many of them are inaccurate
view	Gives the textual and visual representation of the Decision tree model, including the split attribute, split condition and the path it takes in the tree (for example, left child or right child, etc) as a result of that condition.

Deliverables

1. **Report**

   Write a report on your findings. This report should be a maximum of 4 pages long supported by figures and includes sections on the following:

   - **Analysis and results:** Justify the choice for your selected features. Also, describe the results obtained for your impurity measures, providing adequate measurements for accuracy of the classifiers, as well as a few observations on the results for different dataset numbers. A snapshot for the produced decision tree should be provided.
   - **Comparison:** Compare results of your model using the different impurity measures. Visual comparisons are encouraged.

2. **Part 1**

   MySQL code to produce the necessary dataset for the decision tree classifier. In addition, the output CSV file for the dataset.

3. **Part 2**

   MATLAB Code of your decision tree algorithm to produce the accuracy results.

You should complete all deliverables and submit them to the appropriate box on LEARN within two weeks of your scheduled lab.