|  |  |  |
| --- | --- | --- |
| **AIN 380** | **Homework #9**  **Decision Trees Classification** | **NAME:** |

The owner of a diner chain named EATOS has been fielding complaints from the wait staff about their level of tips. Tip levels are considered as shown here and **must** be **recoded** in your Python program as shown here: **<3 pts>**

|  |  |
| --- | --- |
| **Tip Percent** | **Tip Level** |
| < 0.162 | 0 |
| otherwise | 1 |

**Note**: when you recode the Tip percent to a “Tip Level”, you may **NOT** use “Dummies”. **<-3 pts>**

The staff is complaining that they are being assigned an unequal balance of table sizes and day/time shifts. With respect to day/time shifts, the staff is complaining about fairness regarding weekends. So, the owner has collected data for a week of customer tipping activity. The first few lines of data look like this:

A screenshot of a table

Description automatically generated

The ***day*** needs to be recoded, so recode as follows: **<4 pts>**

|  |  |
| --- | --- |
| **Old day value** | **New day value** |
| Thur | 0 |
| Fri | 1 |
| Sat | 2 |
| Sun | 0 |

And the ***time*** needs to be recoded so recode “Lunch” as 0 and “Dinner” as 1. **<2 pts>**

Drop **both** the ‘total\_bill’ and the “Tip Percent” column from the dataframe. The **target** variable is “Tip Level”. Split the data into training and test using a 60%-40% split**. <4 pts>**

After you recode these values, create a **sklearn** *Decision Tree Classifier* ML model using the default “gini” ASM method and the training data. **Then, using the test data, keep running your Python code to create a model until you see an accuracy score over 52%.** Then print the accuracy score, confusion matrix and resulting decision tree. Make sure all output is well labeled. **<7 pts>**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Then, create a second **sklearn** *Decision Tree Classifier* ML model using the “entropy” criterion as the ASM method and use the same training data as was used in the first model. Then, using the same test data, print the accuracy score, confusion matrix and resulting decision tree. Make sure all output is well labeled. **<5 pts>**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

What recommendation can you make to the owner of EATOS **from** the “**gini**” ASM learning model? With what, if any, staff scheduling should he be concerned? Be **VERY** specific and thorough using the learning model results in your explanation! **< 10 pts>**

**\_\_\_Due to the size of the monitor that I am working on, I couldn’t see the full scope of the value matrices, so my specific recommendations may be wrong for that reason. Since the learning model would shows is that on Fridays and Saturdays, we have a large amount of customers, mostly with larger table sizes, so we will need more servers during that time frame. I am also noticing that we don’t have many table sizes larger than 4, so we won’t have to worry about the servers getting overwhelmed when taking that many orders. The final thing that I am noticing is that this restaurant is getting a large amount of middle size tables in the afternoon on Fridays and Saturdays, so we need to make sure that specifically in the afternoon we need more servers for those days. The biggest concern with the staffing schedule is that we won’t have enough people on nights for Friday and Saturday, and too many for Thursdays and Sundays.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Include your recommendation with all of your output and program listing for credit. Also upload a zip file containing your Python code and the dataset file to Canvas.**