**IS 670 Machine Learning for Business Analytics**

**Group Project 2**

**Team Members: Priyanka Malla, Jwalant Maheta, Maharshi Patel, Alaba Adewunmi**

**Data Representation and preprocessing:**

Given are the two datasets

* Reading the dataset file: StudentClass.csv

Table

Description automatically generated

* Reading the dataset file: StudentGraduate.csv

Table

Description automatically generated

* Printing columns of both the files:

Graphical user interface, text, application

Description automatically generated

**Research question:**

**1. Do students perform better in Morning class (before 12:00pm) or afternoon class?**

* First, we split the studentclass.csv dataset into two data frames where one has the records of students attending the morning class (class before 12:00:00PM) and other has the records of student will classes in the afternoon (class after 12:00:00PM).

Text

Description automatically generated with medium confidence

* Calculating the average of grades of students with classes in the morning which is **2.8128**

Graphical user interface, text, application, email

Description automatically generated

* Calculating the average of grades of students with classes in the afternoon which is **2.7753**

Graphical user interface, text, application, email

Description automatically generated

* Calculating the percentage of the students grade average during morning and afternoon classes

**Hence the students attending the morning classes has better performance than students in the afternoon class (70.34>69.38)**

Graphical user interface, text

Description automatically generated

**2. Math, Physics and Statistics are considered quantitative classes. If a student take 2 or more quantitative classes in one semester, will he/she have lower average score?**

* First, we have assigned the values 4 to 0 for the grades A to F.
* Then we created a data frame with students having 2 or more quantitative classes (math, statistics, physics)
* Then we calculate the average GPA of quantitative classes in that academic year and total graded average of each student in that year to compare the quantitative grades to overall average.

Graphical user interface, text, application

Description automatically generated

* Displaying the average of quantitative classes and total average of that student in that year.
* For example for student with student ID 1 in 2004, quantitative GPA is 3.25 and total GPA is 2.7Text

  Description automatically generated
* Displaying the number of students having higher and lower quantitative average.

**As we can see there are more students have higher quantitative GPA (427>424).**

Graphical user interface, text, application

Description automatically generated

Number of students having higher quantitative GPA is 427

Number of students having lower quantitative GPA is 424

**3. Which Major has higher quantitative class GPA?**

* The below code sample shows the average of grades of all the student scores in the respective quantitative classes in every major.

Code sample:

for g in math:

math\_sum = math\_sum + points[g]

math\_avg = math\_sum/len(math)

for g in phy:

phy\_sum = phy\_sum + points[g]

phy\_avg = phy\_sum/len(phy)

for g in stat:

stat\_sum = stat\_sum + points[g]

stat\_avg = stat\_sum/len(stat)

maj\_qclass[major] = {'Math':math\_avg, 'Physics':phy\_avg, 'Statistics':stat\_avg}

for maj in maj\_qclass:

print(str(maj)+"---->"+str(maj\_qclass[maj]))

Text

Description automatically generated with medium confidence

* Displaying the output of the math, statistics, physics average grades in each major.

Graphical user interface, text, application, email

Description automatically generated

* Calculating the average of all three quantitative classes to compare.

**Hence displaying the average of the quantitative score, says that Marketing has the highest average quantitative grade average than the other majors (MKTG = 8.48).**

Graphical user interface, text, application, email

Description automatically generated

**4. Can you combine two dataset and build a classification model to predict if a student can Graduate ON Time (Note: We cannot use the last 2 semester’s performance to predict the graduation, since it is too late to predict the graduation)**

* Reading the dataset files
* Dropping the Null values
* Converting the StudentID into integer Datatype.

Table

Description automatically generated

* Merging the two datasets into a single data frame to build a classification model.

Table

Description automatically generated

* Making a list of all the dates and studentID’s that have taken subjects in last 2 semester of their major. Because we cannot predict the graduation based on last 2 semester's performance.

Graphical user interface, text, application

Description automatically generated

* Making another list of all the row numbers of dataset which needs to be deleted from data frame. Because it contains last 2 semester's performance.

Graphical user interface, text, application

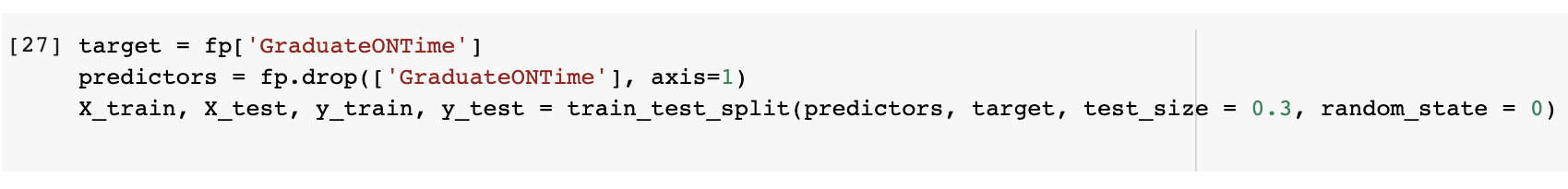
Description automatically generated

* Dropping all the rows from the data frame. Got the final dataset on which we have data except the of last 2 semester data of individual students.

Table

Description automatically generated

* Splitting the data into training and testing data sets, the target variable is ‘GraduateONTime’.



*To determine which classification model predicts the student graduation on time,*

*We are using three classification models*

***1. Using Logistic Regression:***

* Performing logistic regression model on the dataset.
* Also given the confusion matrix for the logistic regression model.

Graphical user interface, text, application, email

Description automatically generated

* ***Performance of this Model is given below, as we see the Accuracy of Logistic regression model is 0.73, the F1-scores is 0.84.***

Table

Description automatically generated

***2. Random Forest classification model:***

* Training the Random Forest Classification model on the Training set.

Graphical user interface, text, application

Description automatically generated

* ***Below is the performance of the Random Forest classification model, the accuracy of this model is very high which is 0.99 and the F1-score value is 0.99.***

Table

Description automatically generated

***3. Using K-NN Model:***

* Training the K-NN model on the Training set.

Text

Description automatically generated

***Below is the performance of the K-NN model, the accuracy of this model is 0.95 and the F1-score value is 0.97***

Table

Description automatically generated

* From all the three models and prediction we have the below accuracy and F1 scores:

|  |  |  |
| --- | --- | --- |
| Model | Accuracy | F1-score |
| Logistic Regression model | 0.73 | 0.84 |
| Random Forest classification model | 0.99 | 0.99 |
| K-NN classification model | 0.95 | 0.97 |

**So according to these prediction Random Forest classification model has the best performance on the give dataset.**