MTH 522: Advanced Mathematical Statistics

Successful Student Attributes for Predicting Passing a Preliminary Year

03/19/2023

Issues

The purpose of this report is to discuss the implementation of a logistic regression model using student attribute data. Logistic regression is similar to linear regression models in that it predicts something about the target value, but instead of predicting a continuous value, it predicts the probability of the target being a certain value. In this case, the target variable is whether a student will pass or fail their preliminary year. The dataset includes 19 different variables that can be used to predict student success, making it difficult to create a logistic model with a good fit for the target variable. The main challenge is to identify the most important variables in determining whether a student passes or fails their preliminary year.

Findings

The dataset provided contains 108 rows and 33 columns, including the column that needs to be predicted, with a total of 176 missing values which have been replaced with their mean values. Out of the 33 columns, five columns contain word data type. Two of these columns were removed from the dataset as they are deemed unimportant, while the other two columns were converted into a numerical data type.

I created a model using a portion of the data from the dataset to train it, and then tested its accuracy. The model was found to be 81.48% accurate. Next, I calculated the coefficient value for each column in the dataset through statistical processes. Based on this analysis, it was determined that 3-4 variables/factors from the student attribute data are useful in predicting the target variable. These variables include the number of F17 GPA, the number of peer mentor meetings attended, and the number of workshops attended.

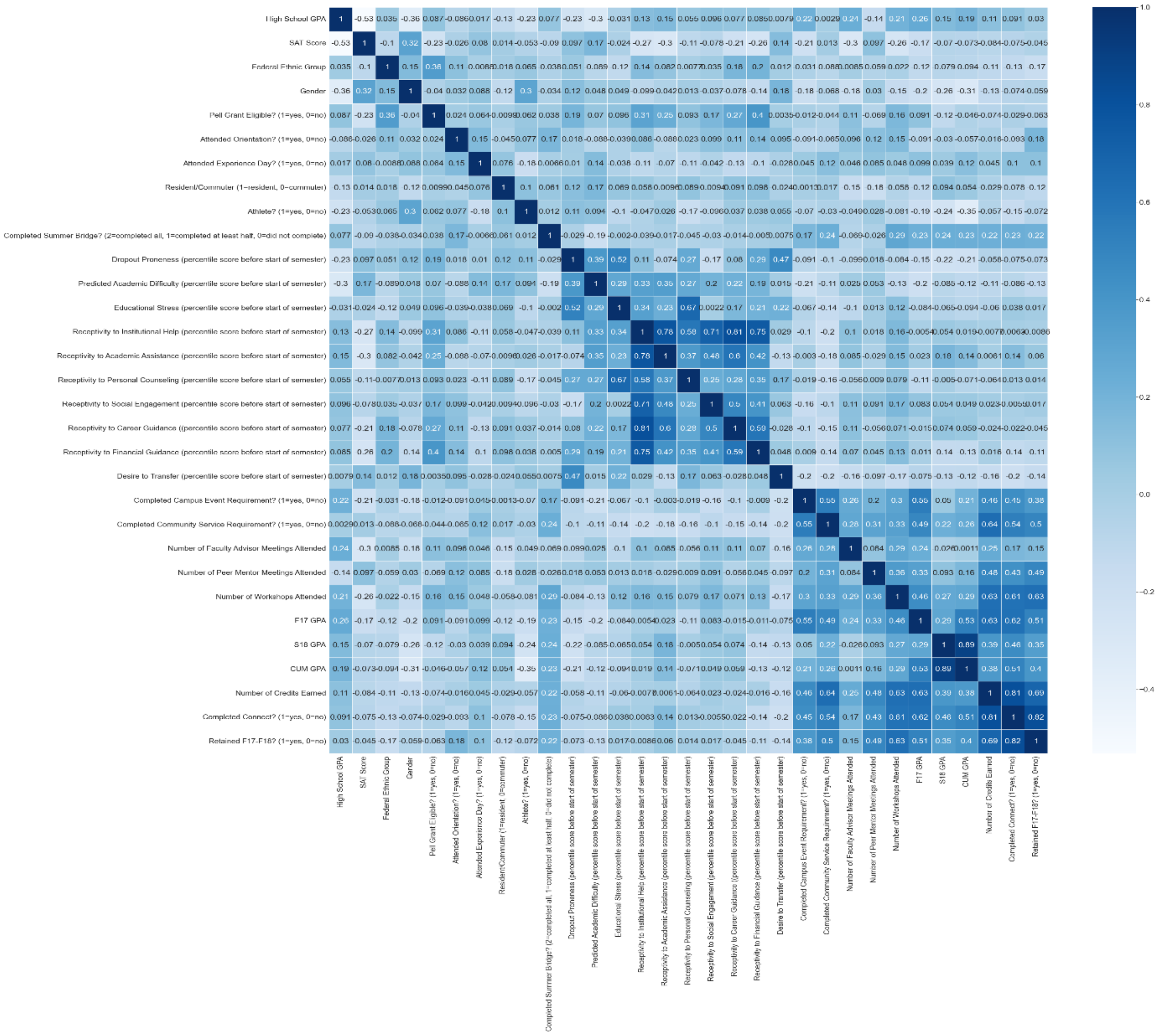
Appendix A: Methods

Firstly, I used the describe method (df.describe()) on the dataset to calculate and display the summary statistics. Next, I used the correlation method (df.corr()) to find the correlation between each column in the dataset. The dataset contains five categorical columns, and two of these variables were converted into numerical data type. To fill in missing values in the columns, I calculated the mean values of the columns using the df.mean() method and then replaced the missing values using the df.fill(df.mean()) method. With the cleaned dataset, I created a model that randomly trains parts of the dataset and tests the model on new, unseen data. I used 75% of the dataset to train the model and the remaining 25% for testing purposes.

Next, I defined a new variable called 'logistic regression' that stores the logistic regression method. I then used the 'logistic regression.fit()' function to fit the testing portion of the predictor variables (x_test) and predicted variables (y_test) using logistic regression. Following this, I made predictions for the predicted variables (y_pred) based on X_test and calculated the accuracy of the predicted variables using the accuracy score method. Finally, I generated a classification report for y_test and y_pred to obtain precision, recall, and f1 scores for both the 'failure' (0) and 'success' (1) outcomes, as well as accuracy, macro average, and weighted average.

Appendix B: Results

We generate heatmap to show correlation matrix between each factor, which is used to identify pairs of variables that are correlated with target variable in order to building a predictive model.



Appendix C: Code

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, confusion_matrix,
classification_report

pd.set_option('display.max_columns', None)
data = pd.read_excel('Preliminary college year.xlsx')

#Dropping last two rows since they are empty
data = data.drop(labels=[106,107], axis=0)
data.info()
data.describe()
data.corr()

ethnic_dict = {
    'White': 0,
    'Black/African American': 1,
    'Hispanic/Latino': 2,
    'Asian': 3,
    'Two or more races': 4,
    'Not Specified': 5
}
gender_dict = {
    'F': 0,
    'M': 1
}
data['Federal Ethnic Group'] = data['Federal Ethnic Group'].map(ethnic_dict)
data['Gender'] = data['Gender'].map(gender_dict)

#Finding Missing Value
miss_values = data.isna().sum()
miss_values
print(data.mean())

#Fill Missing values with Mean values
data_new = data.fillna(data.mean())
```

```
data_new.isna().sum()

#Drop redundant variable
data_new = data_new.drop(['Reason not Retained', 'Reason for not Completing Connect'], axis=1)
print(data_new)

#Plot Heatmap to show correlation
plt.figure(figsize=(50,50))
sns.set(font_scale=2.2)
sns.heatmap(data_new.corr(), cmap='Blues', linewidths=1, annot=True)
plt.show()

#Splitting the data for training and testing
X = data_new.drop(['Retained F17-F18? (1=yes, 0=no)'], axis=1)
y = data_new['Retained F17-F18? (1=yes, 0=no)']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25, random_state=42)

# Create an instance of the logistic regression model and fit it to the training data
logistic_regression = LogisticRegression()
logistic_regression.fit(X_train, y_train)

# Use the model to make predictions on the testing data
y_pred = logistic_regression.predict(X_test)

# Evaluate the performance of the model using accuracy score and confusion matrix
print("Accuracy score:", accuracy_score(y_test, y_pred))

# Identify the most useful variables by examining the coefficients of the logistic regression model
coefficients = pd.DataFrame(logistic_regression.coef_, columns=X.columns)
coefficients = coefficients.transpose()
coefficients.columns = ['Coefficient']
coefficients = coefficients.sort_values('Coefficient', ascending=False)
print(coefficients)

#Classification Report
print(classification_report(y_test, y_pred))
```