1. Introduction

Access to higher education is essential for promoting economic mobility and reducing income inequality, as it provides individuals with opportunities for social mobility, financial stability, and an improved quality of life. However, the cost of attending college can be a significant barrier for many students, particularly those from low-income families. To address this challenge, federal and state governments, as well as colleges and universities, have implemented a variety of student aid programs to provide financial support to students. This report will focus on different types of student aid programs impact on college attendance and persistence rates.

2. Problem Statement

Based on initial analysis on the College Scorecard Dataset, which includes information about the amount of financial aid that each institution's students receive, and the numerical data represents college attendance and persistence rates. Group 20 decided to examine **the impact of different types of student aid programs on college attendance and persistence rates**. Specifically, we will focus on federal grants, student loans, and work-study programs, and use regression analysis to estimate the effects of these programs on college attendance and persistence rates.

2.1 Problems to be analyzed

1) Study the impact of **Pell Grants student loan**, **PLUS loan** and **federal loan** on the 4-year college student graduation rate, and determine which fund has a greater effect.
2) Compare the impact of **other factors** on the 4-year college student graduation rate.
3) Analyze why certain specific factors have a **greater impact**.

2.2 About the data

This report focuses on a specific period of time **2018-2019**, and because the original dataset is huge, we have decided to focus only on the cases where students completed their studies in four years. Initially, we have selected several related variables as the table below shown, however, we find that some variables have some bias on specific features. For example, after we build the model to find the effect of the selected independent variables on **PELL_COMP_ORIG_YR4_RT**, we find that the features related to "federal Pell Grant" will always be allocated a much higher weight, because both of them are linked to "federal Pell Grant". After trying different dependent variables, we finally find that COMP_ORIG_YR4_RT has less bias and can be treated as a representative result variable.

Dependent variables (Persistence and Graduation Rates)	**COMP_ORIG_YR4_RT**: Completion rate for first-time, full-time students after 4 years (chosen)
	LO_INC_COMP_ORIG_YR4_RT: Completion rate for first-time, full-time students with low family income after 4 years
	PELL_COMP_ORIG_YR4_RT: Completion rate for first-time, full-time students receiving Pell Grants after 4 years
	LOAN_COMP_ORIG_YR4_RT: Completion rate for first-time, full-time students with federal student loans after 4 years
	OVERALL_YR4_N: Number of students completing within 4 years
	LO_INC_YR4_N: Number of students with low family income completing within 4 years
	PELL_YR4_N: Number of students receiving Pell Grants completing within 4 years
	LOAN_YR4_N: Number of students with federal student loans completing within 4 years
Independent variables	PCTPELL: The percentage of undergraduates receiving a federal Pell Grant.
	PCTFLOAN: The percentage of undergraduates receiving federal student loans.
	DEP_STAT_PCT_IND: The percentage distribution of students by dependency status and income.
	GRAD_DEBT_MDN: The median debt for students who have completed their education.
	WDRAW_DEBT_MDN: The median debt for students who have withdrawn from the institution.
Control Variables (Institutional and Student Characteristics)	CONTROL: The control of the institution (public, private nonprofit, or private for-profit).
	UGDS: The total undergraduate enrollment.
	ADM_RATE: The admission rate of the institution. (2237/7869)
	SAT_AVG: The average SAT score of students admitted to the institution. (1413 / 7869)

3. Analysis Procedures

3.1 Data selection & cleaning

Step1. Load the dataset: Import necessary libraries and load the dataset into a pandas DataFrame.

Step2. Select relevant columns: Choose the columns that are relevant to the analysis, as previously mentioned in the list of variables related to financial aid, college attendance, and persistence rates.

Step3. Handle missing values: Depending on the nature of the missing data, every time, when we choose a set of variables, we need to remove the rows with missing values, due to numbers of data may be missed in some cases, to make our result accurate, we do not fill those blank with a default value or an average.

Step4. Standardize numerical variables: Some variables might have different scales, such as the average SAT score, which can lead to biased results. To avoid this issue, we standardize the numerical variables using Min-Max scaling.

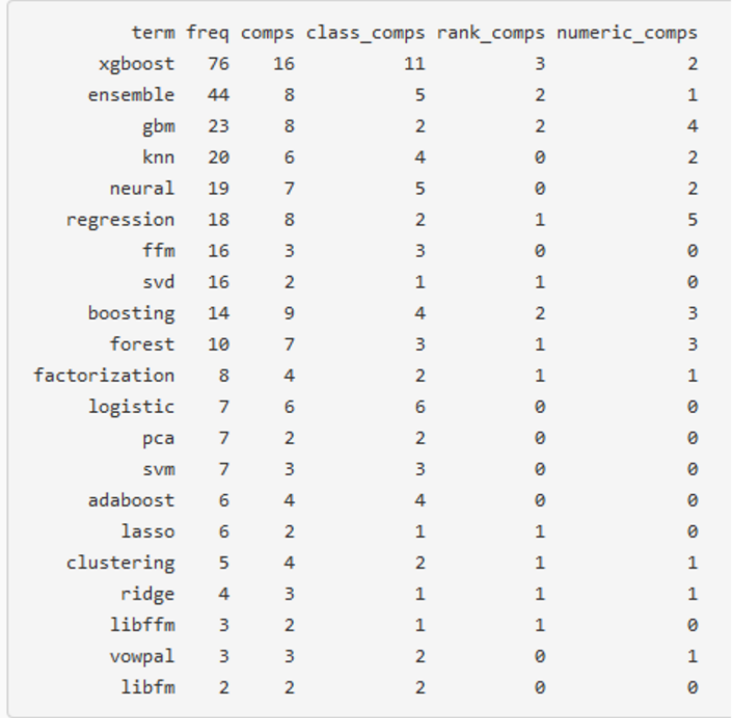
Step5. Encode categorical variables: Our dataset contains categorical variables, "CONTROL", to process it, we encode them into numerical values.

Step6. Split the data: After cleaning and preprocessing the data, we split it into training and testing sets with a ratio of 8:2, to build and evaluate machine learning models.

After completing these steps, we get a clean and preprocessed dataset, ready for further analysis and to build regression machine learning models

3.2 Model selection

After studying the performance of the models in the previous competition, we mainly applied three models in this study, XGBoost, LightGBM, and Linear Regression.



```
         term freq comps class_comps rank_comps numeric_comps
      xgboost   76    16          11          3             2
     ensemble   44     8           5          2             1
          gbm   23     8           2          2             4
          knn   20     6           4          0             2
       neural   19     7           5          0             2
   regression   18     8           2          1             5
          ffm   16     3           3          0             0
          svd   16     2           1          1             0
     boosting   14     9           4          2             3
       forest   10     7           3          1             3
factorization    8     4           2          1             1
     logistic    7     6           6          0             0
          pca    7     2           2          0             0
          svm    7     3           3          0             0
     adaboost    6     4           4          0             0
        lasso    6     2           1          1             0
   clustering    5     4           2          1             1
        ridge    4     3           1          1             1
       libffm    3     2           1          1             0
       vowpal    3     3           2          0             1
        libfm    2     2           2          0             0
```

Figure 1. Different Model Performances in Competitions

Due to the fact that we have learned Random Forest and Support Vector Machine in class as well, we added the two models to see whether in some cases they can have a reasonable performance

Model	Parameters	Description
XGBoost	n_estimators, max_depth, learning_rate	Boosted tree ensemble model that uses gradient boosting and decision trees.
LightGBM	n_estimators, max_depth, learning_rate	Gradient boosting framework that uses tree-based learning algorithms.
Linear Regression	None	Linear approach to modeling the relationship between a dependent variable and one or more independent variables.
Random Forest	n_estimators	Ensemble learning method for classification, regression, and other tasks that operates by constructing a multitude of decision trees at training time.
SVM	kernel function, penalty C, gamma, epsilon, max_iter	Machine learning algorithm that can be used for regression and classification tasks. The algorithm maps input data to a high-dimensional feature space and then finds a hyperplane that separates the data into different classes.

3.3 Modeling process

In the modeling process, firstly we applied grid search for XGBoost, lightGBM, Random Forest and SVM, using 5-folder validation, to determine the rough range of parameters that perform better in this case.

Besides using independent variables in the modeling, we also tried to add different variables as control variables, to see what features will have a significant effect on the completion of those financial-aided students.

All those models are processed in a similar way, firstly we apply a training dataset to train the model, whose hyperparameter has been tuned in advance. Besides using independent variables in the modeling, we also tried to add different variables as control variables, to see what features will have a significant effect on the completion of those financial-aided students. Every time when we train a set of X dataset, we record the information of the best model. The judgment criterion is the root mean square error (RMSE) between the predicted feature values (y_pred) and the test values (y_test).

3.4 Model output

After we have completed the model training and selected the model, then we will output the weights of every input feature as below:

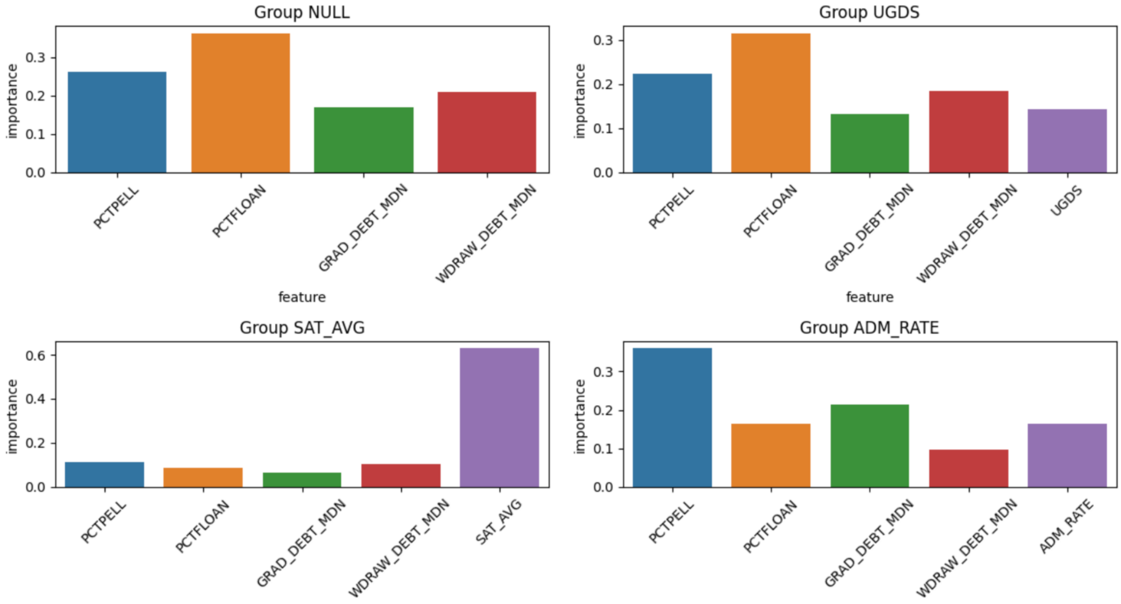


Figure2: Weight of Different Features

Besides, we also chose the best performed input data set(dependent variables plus SAT_AVG), as well as the model it applied:

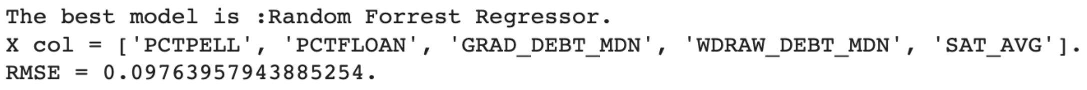

```
The best model is :Random Forrest Regressor.
X col = ['PCTPELL', 'PCTFLOAN', 'GRAD_DEBT_MDN', 'WDRAW_DEBT_MDN', 'SAT_AVG'].
RMSE = 0.09763957943885254.
```

Figure3: Key Information of the Best Result

Although the overall best result came from the random forest model, in most of the cases, like using other input features, XGBoost was always the best model among those five models.

4. Result Analysis

4.1 Prediction remarks

We visualized the RMSE of y_test and y_pred, to have a intuitive view of the predicting performance:

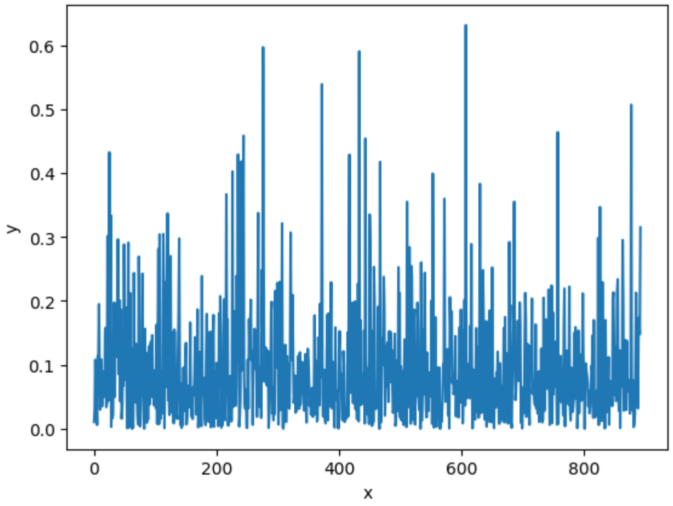


Figure4: RMSE of Test Data and Predicted Data

From the above figure, we can see that the standard deviation error of most of the predicted results is concentrated around 0.1. Considering the limited amount of data in one year and the inherent errors in fitting data, we believe that such a result, although not outstanding, is acceptable.

4.2 Pell Grants vs. Loan

The result of weights of independent values is shown in Figure5, we can see that loan completion has a stronger positive correlation with student graduation rates than Pell Grants. We believe that students who have the courage to take out loans to complete their studies generally are more determined. They may emphasize more on the importance of learning so that they choose to take on debt to attend university. At the same time, the loan also puts them under a certain amount of pressure, so they have to study hard in order to get a good job to repay the loan. Compared with this, students who receive financial aid to complete their studies are somewhat less motivated.

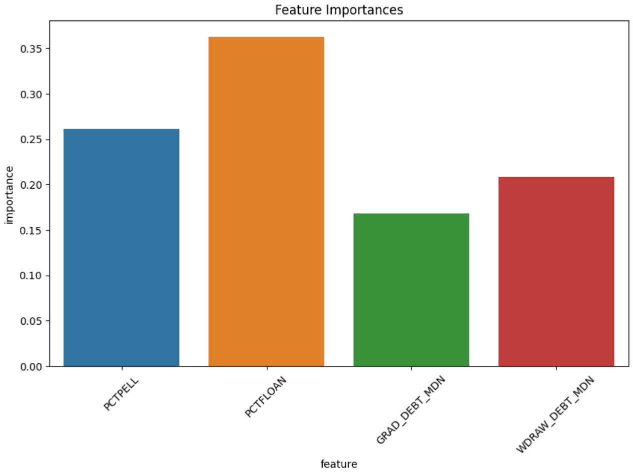


Figure5: Weights of Independent Variables

4.3 Why does the SAT score have such a huge effect?

By analyzing the results, SAT scores have a strongest correlation with student graduation rates. This is in line with the logic of the fact. SAT scores indicate the academic ability of the students before entering this university to some extent. Therefore, students with high SAT scores can complete their studies more easily, so they will have a significant completion rate.

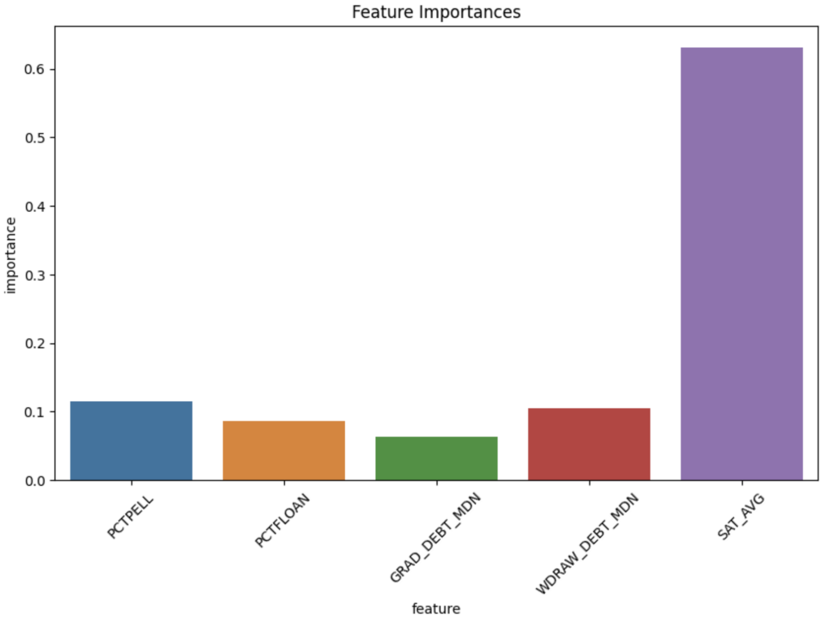


Figure6: Weights of Independent Variables plus SAT Score

5. Kaggle Solution

We first begin by defining a grid of hyperparameters for the XGBoost model, which includes the number of estimators (n_estimators) and the maximum depth of the trees (max_depth). We then use GridSearchCV from scikit-learn to perform a search over the hyperparameters in the grid, using the XGBoost model as the estimator.

Next,we define a TransformedTargetRegressor from scikit-learn, which is a wrapper that applies a transformation to the target variable before fitting a model, and then undo the transformation to make predictions. In this case, the target variable is transformed using a QuantileTransformer, which maps the data to a normal distribution. The GridSearchCV object is used as the regressor within the TransformedTargetRegressor.

The code fits the TransformedTargetRegressor to the training data, which performs the grid search and identifies the best hyperparameters. The best hyperparameters are then extracted from the GridSearchCV object and used to create a new XGBoost model with the optimal hyperparameters. This new model is then fit to the training data using the TransformedTargetRegressor.

Overall, the model performs hyperparameter tuning using grid search and trains a machine learning model with the optimal hyperparameters. The machine learning model is an XGBoost model, and the target variable is transformed using a QuantileTransformer.