Exam Score Regression for ‘MathScore’

Byung Joe Kim, MSA 2023 Data Science

This raw dataset contains 30,641 instances and 14 variables, comprised of 4 numerical variables 10 non-numerical variables The feature-to-sample ratio learns heavily towards the sample size. If there are not enough features, there will be less variance in the data, risking a loss in accuracy or the model being overfit. However, including uncorrelated variables in models can also affect accuracy. Therefore, feature selection needs to be conducted carefully in respect to these factors.

The target value I have chosen is ‘MathScore’, which is a numerical variable with values that follow a normal distribution.

Exploratory data analysis

The dataframe.describe() function provided the Mean, Range, Std for the numerical variables.

	NrSiblings	MathScore	ReadingScore	WritingScore
count	29069.000000	30641.000000	30641.000000	30641.000000
mean	2.145894	66.558402	69.377533	68.418622
std	1.458242	15.361616	14.758952	15.443525
min	0.000000	0.000000	10.000000	4.000000
25%	1.000000	56.000000	59.000000	58.000000
50%	2.000000	67.000000	70.000000	69.000000
75%	3.000000	78.000000	80.000000	79.000000
max	7.000000	100.000000	100.000000	100.000000

I also provided code that outputs the unique groups for the non-numerical categorical variables.

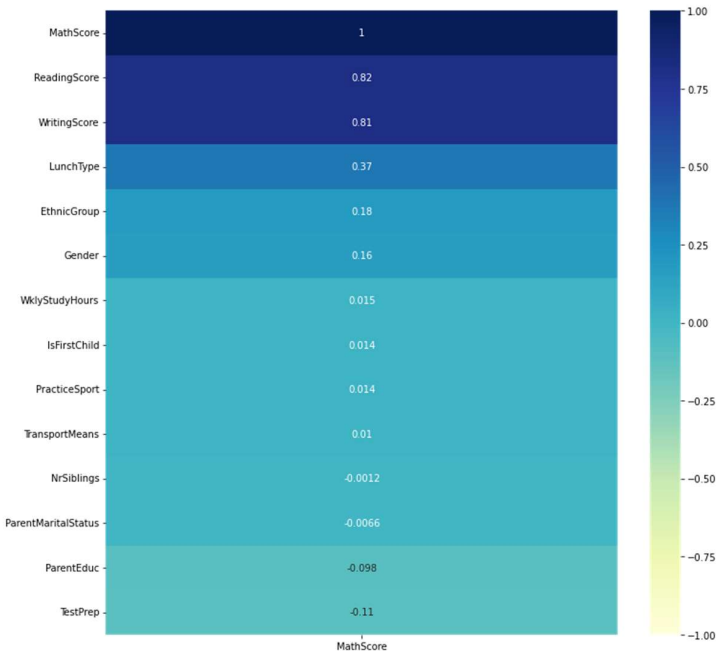
Data Visualisation

For the categorical variables, there were no features that had significant imbalanced results The “unknown” value imputed for the missing inputs often did not go beyond 10% of the total dataset. I was satisfied that no further action would be needed as this imputation would not significantly affect model accuracy.

I was able to represent the numerical variables through graphs in the notebook. Other than the ‘NrSiblings’ variable, all numerical variables follow a normal distribution.

Correlation

When examining correlation between the variables, I chose using heatmaps that implement the default Pearson’s coefficient correlation. This is a suitable option because the dataset size is so large such that the variables are assumed to follow a normal distribution under the central limit theorem. However, the outliers and unknown values of some variables may likely affect these[correlation scores.



As indicated by the 'MathScore' heatmap above, the 'ReadingScore' and 'WritingScore' variables have a strong correlation to the target variable.

Ideally, I want to remove the features that have a low correlation score. For this dataset, I would only wish to retain the variables which score over 0.3. However, this would defeat the purposes of the ML dataset: Predicting the exam scores based on personal and socio-economical factors. Therefore, I initially choose to retain all features at the cost of optimising my model accuracy. However, considering the performance of the models, I eventually decided to remove all variables with a correlation below 0.1 .

Feature selection

Eventually that I removed features such that there were only 5 varaibles left: [MathScore, ReadingScore, WritingScore, Gender, EthnicGroup]. This would be represented through 13 columns due to OneHotEncoding. This is all shown in the "changed_exam_scores.csv". The "preprocessed_exam_scores.csv" no longer has any use.

Part 2

Model Selection

Initially, I chose the LinearRegression as my main model and DecisionTreeRegression as my secondary model due to their speed and simplicity because the current dataset has more than 30,000 samples.

The Linear Regression model works by using least squares calculations to find a linear pattern of correlation between the target variable and other features. This model was performing comparatively better due to the existence of variables with a strong correlation to the target variable. A problem with the Linear Regression is that that there is not much room for tuning. Therefore, I introduced variants of LinearRegression to improve performance. I was choosing between Ridge and Lasso but ended up with Lasso because there is not much collinearity with the target variable in the datatset. The target variable only has strong correlation with 'ReadingScore' and 'MathScore' meaning that Lasso is more suitable.

On the other hand, the Decision Tree model was heavily underperforming. Therefore, I decided to improve the decision tree model through ensemble methods. I decided to upgrade the Decision Tree model into a RandomForest regression model. Combined with this and cost-complexity pruning, I was expecting better performance.

For these reasons, I proceeded training and testing with DecisionTree Regression, RandomForest Regression, LinearRegression and Lasso Regression. I conducted hyperparameter tuning using cross validation of the training dataset.

Results and Evaluation

Model	R squared	RMSE	MAE
LinearRegression	0.6715162523016156	8.793954358513451	7.216979611568991
DecisionTreeRegression	0.6661216662712768	8.865870480318819	7.2611636783324665
LassoRegression	0.6715382934708978	8.793659317602582	7.216638118489102
RandomForest Regression	0.6691946382890933	8.823261049926279	7.2269065748282495

The R Squared metric shows the percentage of variance in the target variable that the can be explained by the other features. The RMSE(Root of mean squared error) metric is the square root value of the MSE metric, which is the sum of the (difference between predicted value and actual value) squared divided by the size of the dataset. Similarly, the MAE(Mean absolute error) metric is the average absolute difference between the predicted values and the actual values.

As noticeable, Lasso Regression model performed the best in every metric, followed closely by the

Linear Regression Model. This was then followed by the Random Forest model and then the Decision Tree Model. Although the decision tree had improved significantly after pruning, both tree-based models were unable to overtake the other two in performance. This is likely happening because the dataset overfits to the variables that did not have a high correlation value.

Overall, I think that such performance of the models was expected, given the low correlation of the variables with the target variable. Although this low correlation within the raw datatset cannot be solved, I could further preprocess the dataset by removing or standardizing outliers to increase model performance. Furthermore, I could also conduct more cross validation or use stratified-k-fold cross validation to ensure there is no overfitting while training the model.