Stats 503 Kaggle Competition Report

Team Name: Oliver6866 & LiangqiTang
Lingqi Huang(hlingqi@umich.edu)
Liangqi Tang(tanglq@umich.edu)

April 2024

1 Introduction

In this Kaggle competition, the task at hand involves predicting the variable 'y' based on a total of 54 features, encompassing various aspects such as self-evaluation, teacher evaluation, extracurricular activities, district information, and 50 other potentially normalized variables. To tackle this challenge, our approach began with thorough exploratory data analysis to understand the relationships between these predictors and the target variable 'y'. Subsequently, we employed linear regression to establish initial associations. Building upon this, we leveraged tree-based methods, including Gradient Boosting and XGBoost, as well as Neural Network techniques to refine our predictions. Our efforts yielded promising results, achieving an R-squared value of 0.8445 with tree-based methods and an even higher R-squared value of 0.8695 utilizing Neural Networks.

2 Exploratory Data Analysis

Prior to applying any statistical methods, we identified that the 'district' variable was categorical, while the remaining variables were continuous. To handle this categorical variable, we transformed it into dummy variables, resulting in the creation of seven new categorical variables. Upon scrutinizing the 50 SRP variables, we observed that they exhibited characteristics resembling those generated from a standard normal distribution, with a mean of 0 and a standard deviation of 1. To substantiate this observation, we generated QQ-plots for each SRP variable and noted a close alignment of points with the QQ-line, lending credence to our conjecture.

Additionally, we computed the correlation matrix among all SRP variables, revealing negligible correlations between pairs, with most correlation coefficients near zero. Consequently, we decided against employing dimension reduction methods, as the uncorrelated nature of the variables suggested that such techniques would not enhance our prediction performance. These findings led us to entertain the hypothesis that the dataset comprises synthetic data, with the target variable 'y' generated using concealed functions, thus prompting further exploration into the underlying mechanisms governing the data generation process.

3 Prediction Methodology

3.1 Linear Models

To gain deeper insights into the contribution of each variable, we employed a simple linear model encompassing all variables, including the newly generated categorical variables. Remarkably, we discovered that the t-values associated with the 50 SRP variables were notably large, indicating that the coefficients of these SRP variables were statistically significant. Conversely, the variable 'teacher_eval' exhibited limited explanatory power within our linear model, while the categorical variables proved to be statistically significant.

Upon fitting a simple linear regression solely for the SRP variables, we observed that this model could account for approximately 43.5% of the variance. By incorporating the remaining undropped variables, the explanatory power of the model increased to around 65%.

In preparation for the subsequent implementation of tree-based methods, we introduced second-order terms for 'self_eval' and 'extracurricular'. This decision was informed by our observation that including these second-order terms not only augmented the model's R-squared value but also rendered these terms statistically significant. We opted against employing spline methods due to concerns about potential overfitting issues in the test set.

3.2 Tree Based Methods, Picking Learning Rates and Max Depth

To implement tree-based methods, we retained all SRP variables, as tree-based methods inherently select variables that explain the most variation when constructing decision trees. Our dataset was randomly split into training data (80%) and test data (20%), and models were trained on the training set. Notably, we opted not to utilize the Random Forest method due to the uncorrelated nature of all 50 SRP variables, which could lead to a high bias problem.

Initially, we employed the Gradient Boosting method with a maximum depth of 3. However, we found that higher maximum depths resulted in a high variance problem during validation. We then transitioned to the XGBoost method, which proved to be more efficient and yielded superior results. Maintaining consistency with our previous settings, including the number of tree estimators, learning rate, and maximum depth, we found that XGBoost significantly reduced computation time while delivering impressive performance.

By setting the number of estimators to 25,000 and iteratively adjusting the learning rate and maximum depth, we identified optimal hyperparameters—specifically, a learning rate of 0.0176 and a maximum depth of 2—that minimized the average test error to approximately 0.366. Subsequently, we trained our model using these optimal settings and achieved an R-squared value of 0.845 on the test set in the Kaggle competition using XGBoost. we then further explored the dataset using neural networks.

3.3 Neural Network

Basic Settings

- 6 hidden fully connected layers with hidden units: 64, 128, 256, 256, 128, 64.
- Number of input features: 60(One-hot encoded "district", teacher_eval, extracurricular, and 55 SRP variables).
- Number of output features: 1 ("y")
- Activation Function: RELU
- Loss function: Mean Squared Error(MSE)
- Learning Rate: 0.0001
- Epoch: 2000
- Weight Decay: 0.01

We do not scale the data since we found that training on unscaled data can get better results. Also, we use cross validation to decide on learning rate, number of epochs, weight decay. We finally constructed neural Network by setting small learning rate of 0.001 and large epoch number of 2000 to get a smoother loss curve, and we set weight decay of 0.01 to prevent overfitting problem. Our final submission.csv is predicted by the model trained on the whole dataset (to make full use of training data) that achieved a training MSE of 0.32 and R^2 of 0.869, which performs better than both linear model and tree-based methods.

4 Discussions and Limitations

In both tree-based methods and neural networks, we opted to incorporate all predictors into our models. To potentially enhance model performance, we also considered the utilization of regularization techniques.

However, a challenge emerged due to the presence of all SRP variables in the dataset, possibly standardized beforehand. This led to a high bias issue when employing traditional Random Forest and Bagging methods. Consequently, we refrained from increasing the maximum depth, which could exacerbate overfitting problems and introduce high variance in the test set. The problem of whether standardization could have improved model performance remained unresolved.

Furthermore, our training dataset was limited, raising concerns about potential overfitting of our models. Due to computational constraints, we were unable to explore deeper neural networks to ascertain if they could offer superior performance. Additionally, our utilization of basic MLP models, coupled with minimal preprocessing of the dataset, underscored the inherent limitations in our approach. While we achieved commendable results, indicative of the neural network's efficacy to a certain extent, there exists substantial scope for refinement and enhancement.

5 Group Member Contribution

Lingqi Huang contributed to exploratory data analysis, linear model fitting, and implementing tree-based methods. Liangqi Tang focused on constructing neural networks and fine-tuning hyperparameters to improve prediction performance.

6 Reference

[1] `https://xgboost.readthedocs.io/en/stable/`

[2] *Dive Into Deep Learning*, by Aston Zhang, Zachary C. Lipton, Mu Li, Alexander J. Smola

[3] *An Introduction to Statistical Learning: with Applications in Python (Springer Texts in Statistics) 1st ed. 2023 Edition*, by Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, Jonathan Taylor.