* Team Member:  
  Jiayue Chen, USC ID: 1010470489  
  Zhengan Pan, USC ID: 3134818448  
  Jiahe Zhang, USC ID: 4321317535
* Kaggle Project Name: Don't Overfit!II
* Kaggle Score: 0.903

Don’t Overfit! II

The objective of this Kaggle project is to predict the target values according to 300 features. The challenge of this Kaggle project is that, there are only 250 rows of training set but 19.8k rows for the test set. As a result, it has a high possibility that our model will overfit the training set but poorly predict the test set. Therefore, we need to find a way to avoid overfitting and improve the rate of accuracy for the test set.

We first split the existing training set into new training set and test set to build the models and check the test accuracy. We have built XGBoost Model, Random Forest Model, Support Vector Machine, Stochastic Gradient Descent, and Logistic Regression Model. However, when we built the model fitting in with the training data, most of them always overfit and have an extremely low test accuracy. Thus, we tried to select top 20 most important features to build the models rather than using all 300 features. We import two package called ELI5 and SHAP, which helped us to choose the top 20 features effectively. After choosing top 20 features, we use GridSearchCV to find out the best parameters for each model and rebuild the new models. After that, I used KFold validation and Neural Network to check if the overfit always occur. Here are the result of train score and test score for each model:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | XBGoost | Random Forest | SVC | SGD | Logistic |
| Train Score | 100.0 | 100.0 | 84.5 | 89.5 | 64.5 |
| Test Score | 74.0 | 78.0 | 72.0 | 74.0 | 62.0 |

After trying these five models and feature selection, we still did not get a decent test score for this Kaggle Project. Therefore, we went to Kaggle Notebook and tried to get some help from other Kaggle users. Fortunately, we find a method called “LB Probing”. By LB Probing, we can get the coefficients for all 300 features and thus improve the accuracy of predicting the target values for the real test set. We have discussed how to utilize these coefficients and how to get them in our python file. After using LB Probing, we got a decent public score on Kaggle, which is 90.3. Therefore, although this is an unusual method to predict the target values, it works pretty well and provides us a new insight of how to solve these kinds of problems.