BLG 454E Learning From Data (2019)
Term Project Report

Idil Ugurnal, Cagla Kaya

***Abstract*— In this report, the details of a classification competition and its steps are elaborated. The method for classifying instances to have Autism Spectrum Disorder (ASD) or not is explained in detail. To do this, feature selection and ensemble methods are used and explained. In an accuracy between 0.55 0.675 depending on the data and selected features is received. This report explains the methods used and the results for this project.**

I. INTRODUCTION

This project is done in order to use what we have learned in the Learning From Data Class in practice. Our target is to classify each instance in the given test data to determine if they have Autism Spectrum Disorder (ASD) or not. To do this, feature selection methods and ensemble classifiers are used. Ensemble learning is a method to help improve learning via machine learning. Different methods are used to achieve prediction accuracy [1]. AdaBoost classifier is one of these ensemble methods, our project uses this method to classify instances with ASD. To explain how AdaBoost classifies our data, it combines multiple poor-accuracy classifiers and gets one high accuracy strong classifier [2]. More information about our experiment and methods will be given in the report.

- Kaggle Name: Idil Ugurnal and Cagla Kaya, Team Name: 150150017_150150117, Public Score: 0.67500, Current Public Rank:18

II. DATA SET USED

The training data for this project consists a sample brain graph which represented as distance between two brain regions and each brain graph labeled according to if autism appeared Given that the training data set had 120 instances with 595 features, too many features for low number of instances, so it was necessary to reduce dimension for better learning. Feature selection is the dimension reduction method we used for extracting most relevant features. For feature selection, firstly we use Chi-Square test to select k features to see which instances are more related with the output. Then, we calculated the correlation between the selected features. Some of the features selected was highly correlated as seen in the Fig. 1. (Highly correlated feature pairs represented as yellow and low correlation is represented with blue.) Since the correlation measures how the features are related to each other, we eliminate the feature which has less relation with output between redundant features. After doing so, we had a feature set that is small enough in dimension to train and get accurate results with an ensemble learner.

III. METHODS USED

The experiment conducted has three parts; prepossessing part where we select features that are useful and reduce the dimension of the data set, training part where we train the data set, and finally, the prediction part where the trained model is used to classify the instances in the given test data. Python was used as the prigramming language when conducting the experiment.

- Feature Selection: The number of instances in the training data are 120, and the number of features are 595. This situation is explained as the curse of dimensions. SelectKBest function under sklearn.feature_selection library is used to select best 50 features out of the 595 features. You can see from Fig.1 the correlation of the features.

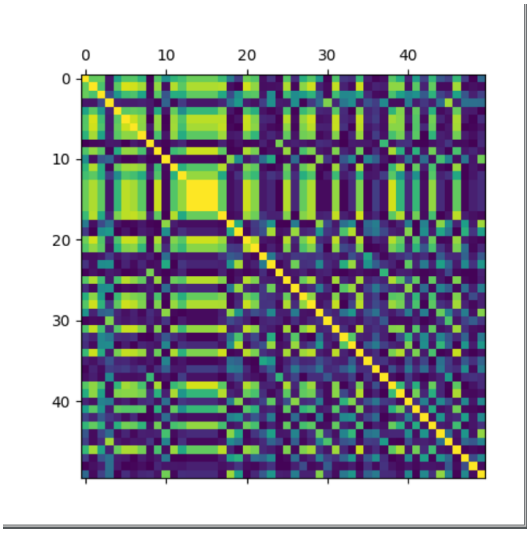

Fig. 1. Correlation of the Selected Features

- Training Model: AdaBoostClassifier from the python library sklearn.ensemble is used to train the data and fit the model. This classifier has two important parameters named as n_estimators and learning_rate. N_estimators parameter is the maximum number of estimators at which the boosting operation is finished. If there is a perfect fit before reaching this value, the procedure is terminated [3]. The other parameter learning_rate has a trade-off relationship with n_estimators. As n_estimators grow it shrinks or if it shrinks, the other grows. For n_estimators, we gave the value of 10 since it got the best accuracy at this result and learning_rate was left at

its default value. More detailed representation can be seen on Fig.2.

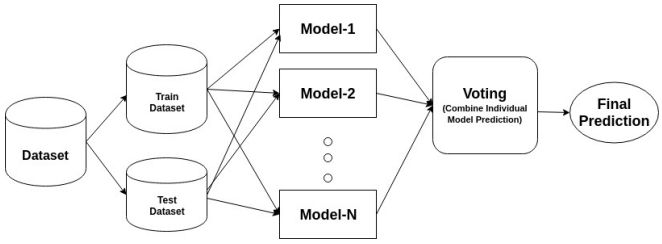

Fig. 2. Ada Boost Classifier

- Prediction: After training our model with the given training data, we predict values for the test data.

IV. RESULTS

At the end of project, according to the results at Kaggle, accuracy of our model was %67,5 for public data with ranking 18 and %52,5 for private data with ranking 18. At the beginning, for dimension reduction, we used feature extraction technique PCA, but accuracy of our model was lower. So we change our method for dimension reduction. And we took best result for fifty best features according to chi square test minus five correlated features. Other than that, for Adaboost, we tried different number of estimators and ten was the best.

V. CONCLUSIONS

In conclusion, feature selection was used in order to get rid of the curse of dimension. After altering our feature vector, we used AdaBoost Classifier to train our data set. Finally, the trained model was used to predict classes (has ASD -¿ 1, does not have ASD -¿ 0) in the test data. The first accuracy result at the start of our trials for finding ensemble methods and feature selection methods was around 0.40, after feature selection and using the correct ensemble method, we managed to increase this score to 0.675.

REFERENCES

[1] V. Smolyakov, “Ensemble learning to improve machine learning results.”
[2] A. Navlani, “Adaboost classifier in python.”
[3] scikit learn, “Adaboost classifier.”