CS5691: Pattern Recognition and Machine Learning: Data Contest

Sourav Sahoo, EE17B040 (Team 27)

December 27, 2020

1 Introduction

The data contest is about predicting whether an employee is going to leave the company or not in coming few months. The data collected is about the ratings and remarks given by the employees of different companies on a single platform, where the employees can participate according to their choice. We deploy various machine learning models to predict the chance of a employee leaving the company.

2 The Proposed Method

In this section, we briefly describe the raw data received, the preprocessing performed to clean the data and the classifiers used for the final predictions.

2.1 The Data

We receive four `csv` files containing the data and a fifth `csv` file which contains the test data. The train files are:

- `train.csv` – This file contains the data if an employee left the company or not.
- `ratings.csv` – Contains the ratings provided by the employees with time stamp.
- `remarks.csv` – Contains the remarks provided by the employees with time stamp.
- `remarks_supp_opp.csv` – This file contains the data about how well a remark is supported or rejected other employees.

There are 3526 train data points and 882 test data points.

2.2 Data Preprocessing

As the remarks are all redacted, we only take into account the length of any remark to represent the remark itself. For the ratings, we form a list containing the total number of times an employee has rated and the frequency of each type of rating (1-4) for each employee. It is observed that the fraction of employees, prior to their departure, often rated the company lower (1-2) as compared to the employees who did not leave.

Finally, from `remarks_supp_opp.csv` file, we count the frequency of number of supporting vote received by any employee regarding their remark. This attribute roughly serves as a measure of the reliability of the redacted remarks.

All the NaN values and duplicates are dropped from all the data. We split the dates across all the data files into three individual quantities: day, month and year. All the other attributes from `train.csv` is retained. The name of companies are converted into one-hot encoding type vectors. So, finally, we get a 51-dimensional input vector for each data point.

2.3 Implementation

All the experiments are carried out using `Python` and `scikit-learn` library. We tried out various methods starting from simple classifiers like `LogisticRegression` till more ensemble classifiers like `RandomForestClassifier`, `AdaBoostClassifier`, etc[1]. We obtained our best score using **XGBoostClassifier** algorithm. For further model ensembling, a certain fraction of train data (say 80%) was randomly selected to train the classifier and the combined prediction of N such different runs was utilized to make the final prediction.

3 Conclusion

Our best performing model achieves 89.3% weighted categorization accuracy on the public leaderboard on Kaggle contest page. All the codes pertaining this contest are made available here.

[1] A detailed list of methods utilized is available in the submitted code