FinalReport

b327zhan

8/17/2021

(Note that all code written in R has been knit to a pdf. So it's ok not running them if you didn't use R before.)

I used two models in this competition, the first one is based on decision tree using library `lightgbm`, and this is the one generated final submitted result.

Another one is based on neural network; it is a self-implemented voter neural network. They have different data processing method. And this report will mainly focus on the `lightgbm`

(A) Decision Tree Model (lightgbm)

(1) Data cleaning and data processing

Included in `Data-Process.pdf`, which is generated from `Data Process.Rmd`. It contains all the code and document.

It cleans some missing type and introduces some new features.

(2) Model fitting and prediction

This part is included in `./Tree/tree.ipynb`

It includes showing feature importance; hyperparameter tuning; prediction; confidence interval between public and private prediction.

(3) Result diagonose

This part is included in `./Tree/Visualization/visulization.pdf`

(B) Neural network voter (optional to read, different approach)

This approach is too time consuming for such competition for a course, so I implemented it to see if NN has a great boosting. It end up having similar performance on public score. So, lightgbm method is the main method. But this one gives a different view of processing the data, `panda` was used here, not `R`.

(1) and (2) has been included in `./NN based/NNvoters.ipynb`. The following is for short.

(1) Data cleaning and data processing

Cleaning the data by filling in missing types.

Categorical data encoded as one-hot vector.

Log transform the salary to decrease the effect of outliers.

Decrease the dimension by LinearSVC.

(2) Model fitting and prediction

The idea of this network is that to train n neural network, each with a small amount of data, say 2000 data with balanced label. When fitting the data, they will vote to give a final answer. But if the number of 0 voting is very similar to that of 1 voting, it collects that data point to set `S`. And another n neural network will then train based on a subset of combination of original data and `S`.

So when it comes to prediction, the network will vote on the data, and pass those data having similar votes to the second group of voters.

(3) Result diagonose

Same procedure.

(C) Appendix

Note that `Comparision` contains 4 results

1. Lightgbm with all features (0.71797/0.71006)
2. lightgbm with selected features by dropping features by mutual infomation score (0.71409/0.70995)
3. voters result (0.71573/0.71059)
4. public best score (0.71901/0.71463)

After some differencing, I chose to submit lightgbm with all features,

it scored 0.71797/0.71006 (private/public)

If I didn't train with balanced data, for 1, it will give 0.72043/0.71314.

But this is a matter of test set. In general cases, we should balance the data, But at least for this data set, we can't tell which model is acatully the best.