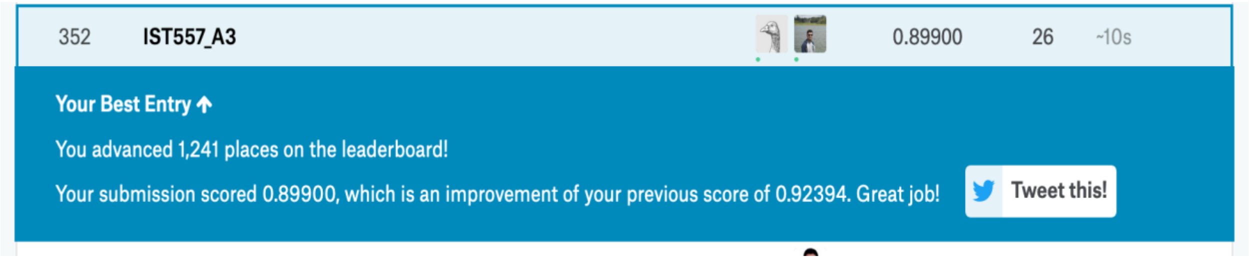
IST 557 Project Progress-3

Mihir Mehta (mzm6664) & Mudit Garg (mxg5783)

Best performance

The best performance- 0.89900

The corresponding snapshot is:



352 IST557_A3 0.89900 26 ~10s

Your Best Entry ↑

You advanced 1,241 places on the leaderboard!

Your submission scored 0.89900, which is an improvement of your previous score of 0.92394. Great job!

Best Method

XGBoost with feature engineering.

After doing an extensive feature engineering, we did parameter tuning and found the following parameters to be optimal.

- max_depth=10,
- n_estimators=1000,
- min_child_weight=0.5,
- colsample_bytree=0.9,
- subsample=0.8,
- eta=0.1,
- seed=1

Other Methods

- After the presentation of top team at the end of round 2, we felt the necessity of fine tuning hyper parameters of our model. There were two options that we could have tried.
    - The first is normal grid search approach
    - Second is bayesian optimization based search approach.

  We decided to explore the second approach. We used "hyperopt" package to fine tune parameters of random forest and xgboost models.
- We tried to fine tune XGboost model and we got RMSE value of 0.97 on Kaggle. This is due to two reasons. i) We fine tuned this XGboost model on old feature engineering. ii) Hyperopt requires atleast 200 iterations to show its effect. Given time constraints, we could not achieve the same.
- Similarly, random forest showed RMSE of 1.23 on similar reasons.
- We then tried adding the features like lags and TFIDF of shop name, item name etc. But the system crashed because of low memory.
- We added features like city name (derived from shop name)and holidays of Russia.
- Seeing the month over month trend of items sales, the sales shoot up in Dec as compared to november probably due to year end. Hence, we decided to add holidays as a feature. Source: https://www.timeanddate.com/holidays/russia/#!hol=25231673

- We then fine tuned the parameters using GridSearchCV and we got RMSE as 0.89900

Going forward, we would like to understand underlying mechanism in hyperopt so that we can get better results.

Summary

Methods	**RMSE**
XGBoost using hyperopt	0.97
RandomForest using hyperopt	1.23
XGBoost using feature engineering	0.905
Tuning XGBoost along with feature engineering	**0.89900**

Contribution
Mudit Garg- 50%
Mihir Mehta – 50%

References

- https://www.kaggle.com/jatinmittal0001/predict-future-sales-part-1
- https://www.kaggle.com/dlarionov/feature-engineering-xgboost
- https://www.kaggle.com/eikedehling/tune-and-compare-xgb-lightgbm-rf-with-hyperopt
- https://towardsdatascience.com/an-example-of-hyperparameter-optimization-on-xgboost-lightgbm-and-catboost-using-hyperopt-12bc41a271e
- https://github.com/jaberg/hyperopt
- https://towardsdatascience.com/hyperparameter-optimization-in-python-part-2-hyperopt-5f661db91324