Predicting Employee Attrition

Data Science Capstone Project, April 2021

The Problem

- According to the U.S. Bureau of Statistics in 2018, the average turnover rate in the U.S. is about **12-15%** annually.
- LinkedIn shows an average annual worldwide employee turnover rate of **10.9%** in 2018.

In simple terms, for every 10 employees hired, 1 will leave.

Who might impact?

Small Companies

- Limited resources, cannot afford to hire people based on trial and error

Large Companies

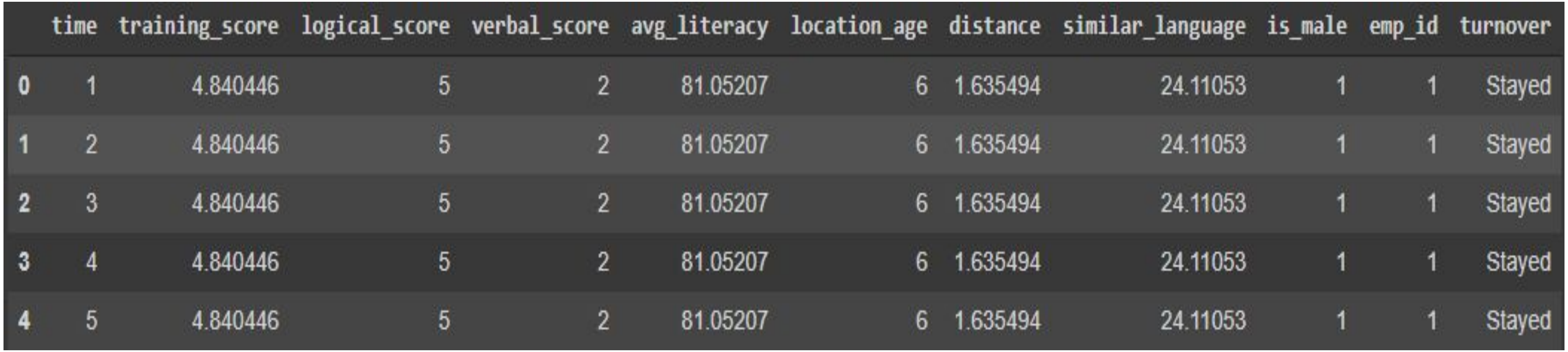
- Resources may not be limited but in the long run, it accumulates to a significant amount

What factors might affect turnover?

- Salary
- Education level
- Satisfaction level
- Time spent in company
- Literacy level
- Age
- Location
- Travel distance to work
- Gender
- Similarity of prevailing language

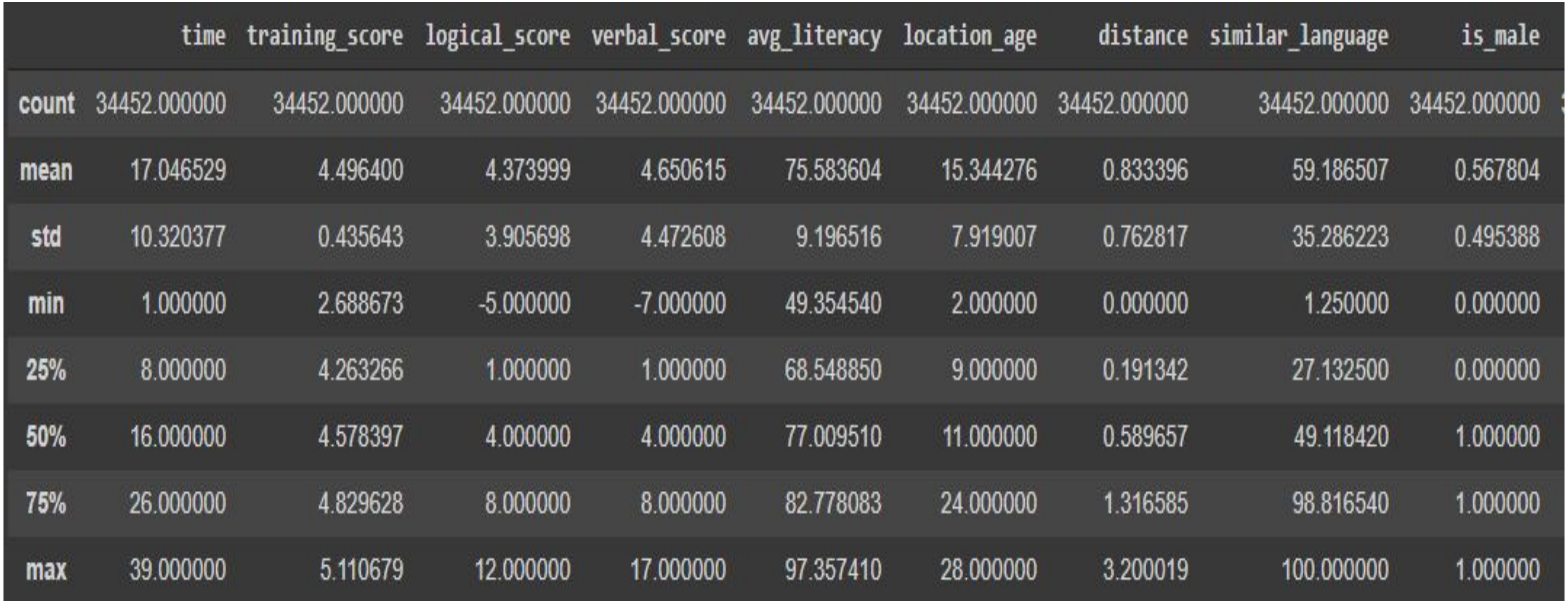
Sample of cleaned data

Mostly numeric data types for comparison purposes



	time	training_score	logical_score	verbal_score	avg_literacy	location_age	distance	similar_language	is_male	emp_id	turnover
0	1	4.840446	5	2	81.05207	6	1.635494	24.11053	1	1	Stayed
1	2	4.840446	5	2	81.05207	6	1.635494	24.11053	1	1	Stayed
2	3	4.840446	5	2	81.05207	6	1.635494	24.11053	1	1	Stayed
3	4	4.840446	5	2	81.05207	6	1.635494	24.11053	1	1	Stayed
4	5	4.840446	5	2	81.05207	6	1.635494	24.11053	1	1	Stayed

Average Stats of features



	time	training_score	logical_score	verbal_score	avg_literacy	location_age	distance	similar_language	is_male
count	34452.000000	34452.000000	34452.000000	34452.000000	34452.000000	34452.000000	34452.000000	34452.000000	34452.000000
mean	17.046529	4.496400	4.373999	4.650615	75.583604	15.344276	0.833396	59.186507	0.567804
std	10.320377	0.435643	3.905698	4.472608	9.196516	7.919007	0.762817	35.286223	0.495388
min	1.000000	2.688673	-5.000000	-7.000000	49.354540	2.000000	0.000000	1.250000	0.000000
25%	8.000000	4.263266	1.000000	1.000000	68.548850	9.000000	0.191342	27.132500	0.000000
50%	16.000000	4.578397	4.000000	4.000000	77.009510	11.000000	0.589657	49.118420	1.000000
75%	26.000000	4.829628	8.000000	8.000000	82.778083	24.000000	1.316585	98.816540	1.000000
max	39.000000	5.110679	12.000000	17.000000	97.357410	28.000000	3.200019	100.000000	1.000000

Average Stats of features

- Logical and verbal score are relatively low, both averaging around 4.5 with a max score of 12 and 17 respectively
- On the other hand, average literacy score is very high at about 75 with max score of 100
- This data set seems to consists of a younger group of working people with max age of 28
- In terms of similarity of languages between people, this group has a 98.8% similarity within 0.75 range
- Slightly higher percentage of male

Scaling imbalance turnover rate

Original Data

- Model Accuracy = 0.9857
- ROC score = 0.6849

Upscale Minority Class

- Model Accuracy = 0.6305
- ROC score = 0.6844

Downscale Majority Class

- Model Accuracy = 0.6447
- ROC score = 0.6964

Observation:

- Original Data may have the highest model accuracy but class is imbalanced
- Upscaling class has an overall lower model accuracy and ROC score due to the huge difference between minority and majority class. Since the data points are reuse and over-used, it can be said to be overfitting
- Downscaling majority class decreased model accuracy as most of the data points are left out and the random selection could have left out important points

Training Models

Logistic Regression

- Accuracy score = 0.9860
- Test CV score = 0.6673
- Train CV score = 0.6742
- STD CV test score = 0.0343
- Cross Validation Score
  - 0.7179652, 0.64713546, 0.62896349, 0.64469899, 0.69765829

KNN - K-Nearest Neighbor

- Accuracy score = 0.9860
- Test CV score = 0.6603
- Train CV score = 0.6644
- STD CV test score = 0.0404
- Cross Validation Score
    - 0.64571037, 0.63843017, 0.70884234, 0.70434165, 0.60400663

SVM - Support Vector Machine

- Accuracy score = 0.9860
- Test CV score = 0.5333
- Train CV score = 0.5501
- STD CV test score = 0.04356
- Cross Validation Score
  - 0.56095787, 0.44707455, 0.54383946, 0.55510812, 0.55974417

Random Forest Classifier

- Accuracy score = 0.9833
- Test CV score = 0.6715
- Train CV score = 0.7032
- STD CV test score = 0.0367
- Cross Validation Score
    - 0.65524006, 0.67603465, 0.70499306, 0.61017394, 0.71123651

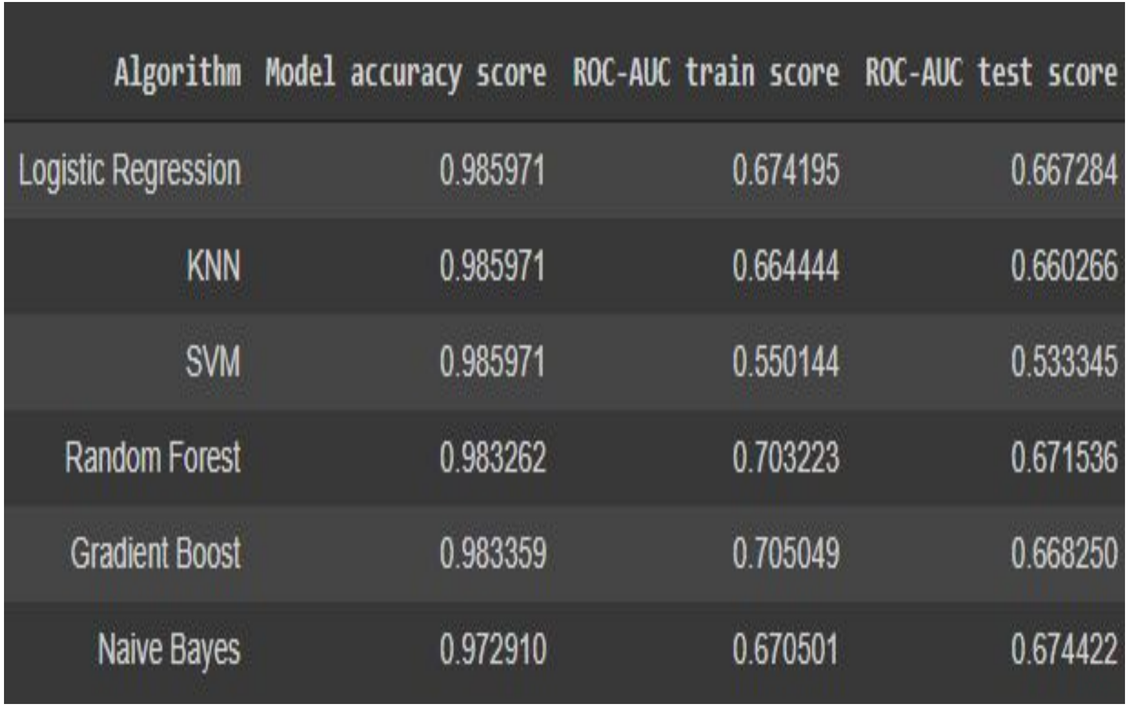
Gradient Boosting Classifier

- Accuracy score   = 0.9834
- Test CV score   = 0.6682
- Train CV score   = 0.7050
- STD CV test score   = 0.0321
- Cross Validation Score
  - 0.64020565, 0.70703191, 0.70803865, 0.64323542, 0.64273629

Naive Bayes (GaussianNB)

- Accuracy score = 0.9729
- Test CV score = 0.6744
- Train CV score = 0.6705
- STD CV test score = 0.0314
- Cross Validation Score
  - 0.67791852, 0.64539271, 0.68249805, 0.63926771, 0.72703123

Algorithm train-test score performance Comparison

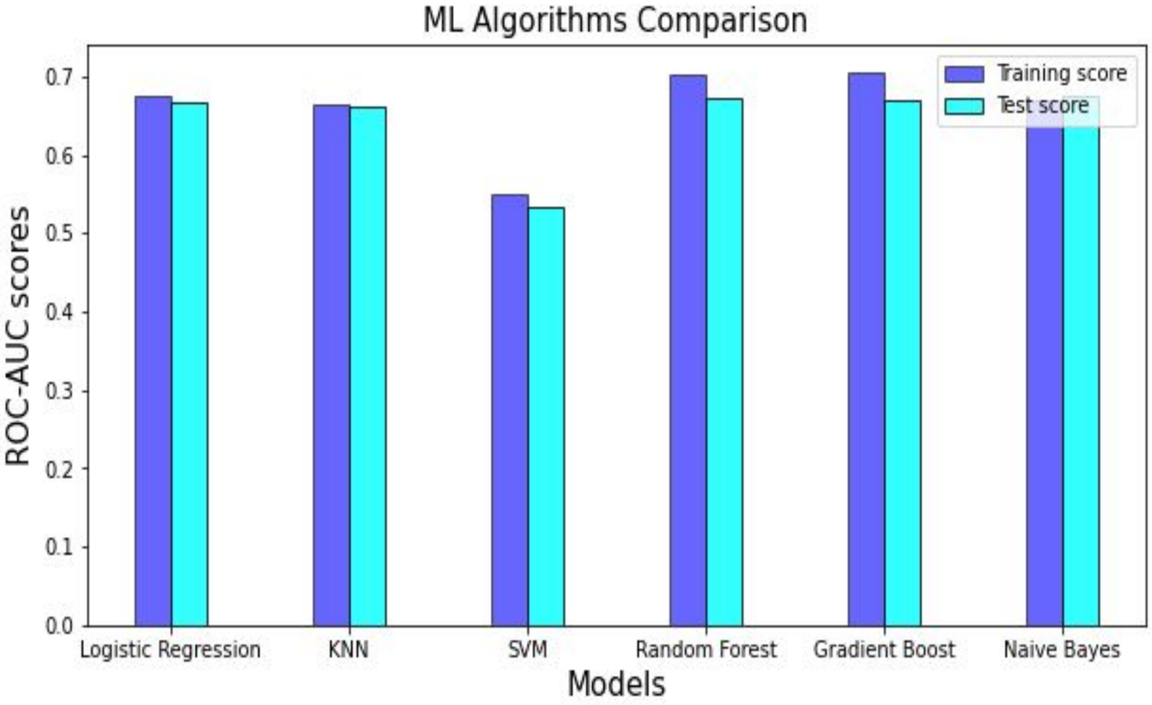


Algorithm	Model accuracy score	ROC-AUC train score	ROC-AUC test score
Logistic Regression	0.985971	0.674195	0.667284
KNN	0.985971	0.664444	0.660266
SVM	0.985971	0.550144	0.533345
Random Forest	0.983262	0.703223	0.671536
Gradient Boost	0.983359	0.705049	0.668250
Naive Bayes	0.972910	0.670501	0.674422

Model Accuracy Comparison:

- For the most part, all algorithms have a relatively high model accuracy score which is the ideal scenario. However, having a high model accuracy score is not everything as the model could have very well be overfitting. Therefore, we will also be looking at the ROC-AUC train/test score for comparison.

Visual Comparison on algorithm performance



Comparison:

- SVM is certainly not a suitable algorithm in this case
- Logistic Regression, KNN, and Naive Bayes have a relatively similar training and test score at about 6.7
- Random Forest and Gradient Boosting appears to top the cheat in both training and testing score

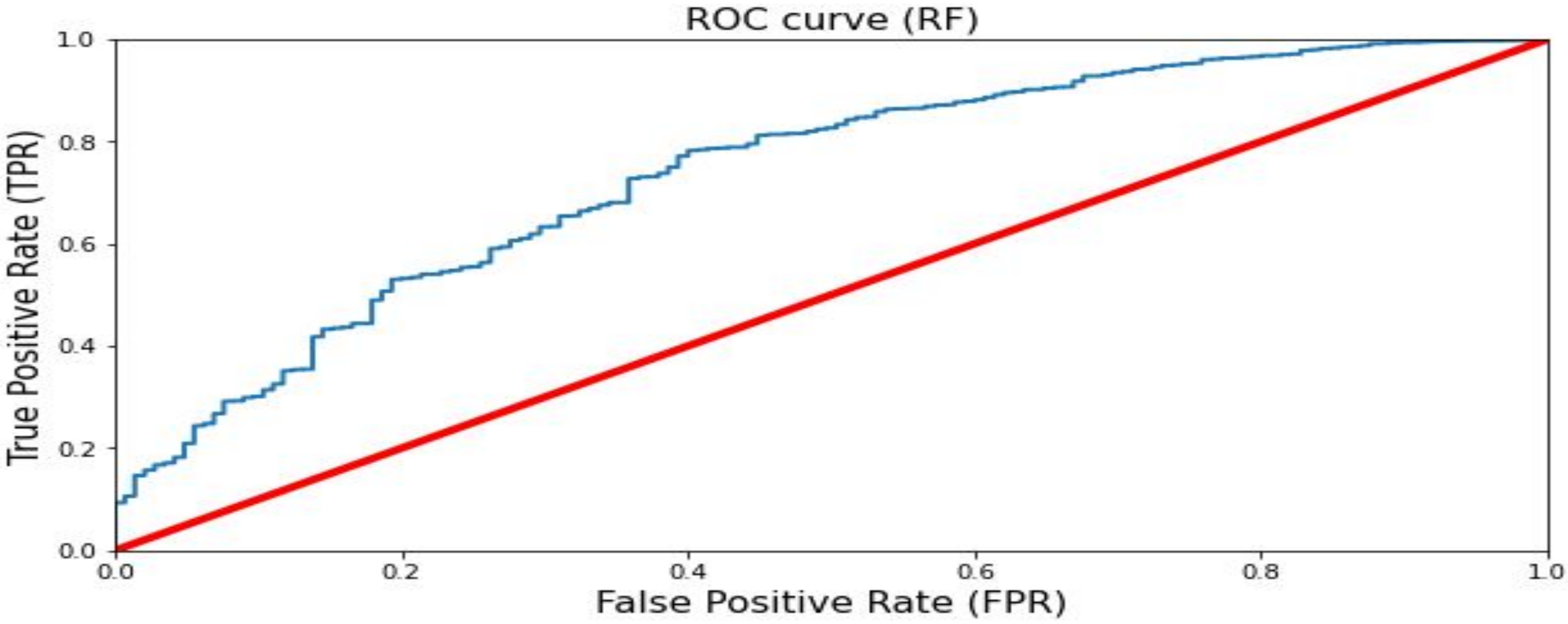
Fine tuning Models

Random Forest

- min_samples_leaf = 5
- min _samples_split = 2
- N_estimators = 200
- N_jobs = -1
- Random_state = 1

```
Best Score = 0.9855
Accuracy Score = 0.9859
CV Score = 0.6826
```

Random forest: Roc-auc score = 0.7385



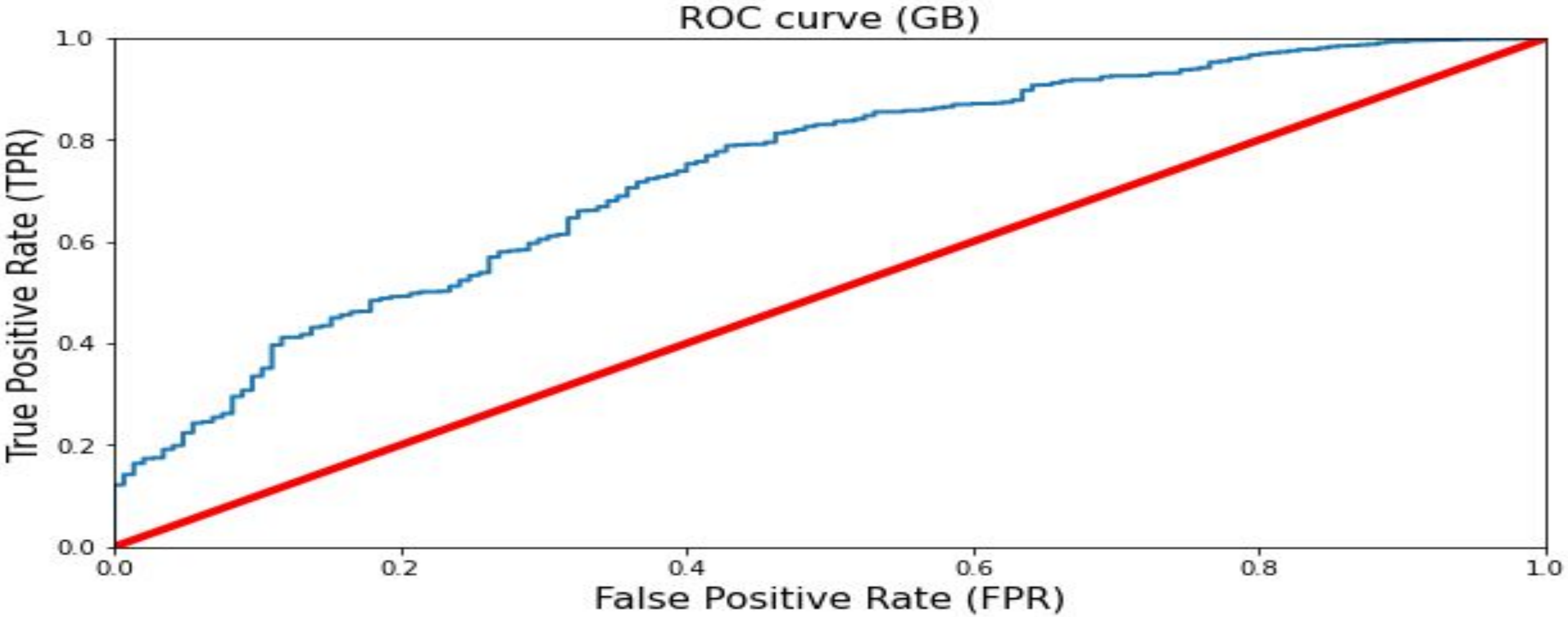
Fine tuning Models

Gradient Boosting

- Criterion = friedman_mse
- Learning_rate = 0.1
- Loss = deviance
- Max_depth = 4
- Max_features = 0.3
- Min_samples_leaf = 150
- Min_samples_split = 2
- N_estimators = 100
- Presort = deprecated
- Subsample = 1.0
- Tol = 0.0001
- validation _fraction = 0.1

```
Best Score = 0.9856
Accuracy Score = 0.9859
CV Score = 0.6787
```

Gradient Boosting: Roc-auc score = 0.7344



Improvements / suggestions

- This is a relatively small dataset, therefore there is room for improvement if more data were to be trained.

- Other improvements that could be made to achieve results would be to test out other ML models such as Extreme Gradient Boosting Classifier Model.

Conclusion

- Out of all the algorithms used, SVM showed the lowest ROC-AUC scores. Although I have selected the RF and GB as my method of modeling due to its high ROC-AUC scores, I know that this may not necessarily by the best solution as the model could be overfitting.
- The used of Cross Validation in this notebook is to prevent overfitting of model on the same dataset. This method splits training set into k-smaller sets where the models are trained using k-number of folds and predictions are validated on remaining testing sets.
- Noticed that the hyperparameters of the GB algorithm was not too fine-tuned. This is because my current setup does not allow me to perform too many computations to achieve optimized hyperparameters.
- The final turnover prediction was saved into a csv file (gbc_turnover_prediction.csv).