Prediction of IBM HR Analytics Employee Attrition using Machine Learning

Shu-Ya Chiang

Abstract- *Human Resource department is very important in an organization. Organizations should be aware of the work attrition rates of their organizations. It's impossible to have a zero attrition rate because there are people who quit their job or retirement. The most common reasons for employees to leave their jobs are lack of opportunities to grow and poor training regarding leaving their jobs willingly or unwillingly. Every organization will face the challenge of work attrition, and it will invest money and time to recruit and train new employees, so utilizing the results from the classification dataset by conducting machine learning models can give an organization a better idea of the factors of influence employees to leave their jobs. Different approaches will be discussed in this paper by resampling the imbalanced class label, feature selection, and machine learning methods, such as Gradient boosting, Random Forest, Ada Boosting, Decision Trees, and Support Vector Machine. The results can be utilized in a further study of evaluating the current management style and improving the promotion system of organizations in order to keep the high-performance worker staying in the organization.*

1. Introduction

Human Resource department plays an important role in an organization. It is in charge of recruiting, training, communicating new workers, managing employee-benefit programs. IBM HR Analytics Employee Attrition & Performance dataset is found on Kaggle and it is a classification problem. The target is Attrition. There are 34 features in the original dataset. After encoding the categorical features and cleaning the dataset, the dataset is 1470 rows and 51 features. The target is Attrition, and the goal is to gather the important features in employees who are likely to leave their job. 237 out of 1470 employees leaves their job, which is around 16% of all cases. It is a challenge to make predictions without demonstrating resample mothed such as oversampling, SMOTE and SMOTEENN, and undersampling on the training set. Conducting feature selections by embedded techniques, for example, Gradient boosting and Random Forest can reduce the model's complexity.

This research is focused on comparing different approaches in resampling methods, feature selections, the prediction of attrition by performing Gradient boosting, Random Forest, Ada Boosting, Decision Trees, and Support Vector Machine machine learning algorisms on this dataset. The model performance will be evaluated by the classification report which is included accuracy, AUC, precision, recall, F1-score, and support. Since this dataset has a serious imbalanced class issue, the AUC score and F1-score will take more accountability in the evaluation. Besides training and test set split, the cross-validation will be utilized in this research as well. The goal is to explain common features from the models that have high AUC, high F1-score, and low run-time. Additionally, visualizing the top 15 importance features from the models and compare the ranks of feature importance in different machine learning methods.

2. Literature review

Utilizing machine learning algorithms predicts employee turnover has some research supporters, so there are few published papers discusses this approach to improve the management style of an organization. In the paper, Prediction of Employee Turnover in Organizations using Machine Learning Algorithms [1], the author applied XGBoost, Logistic Regression, Naïve Bayesian, Random Forest, SVM, LDA, and KNN on HR Information Systems (HRIS) data to build the models, and the results were evaluated by AUC in Training and AUC in Holdout, Run-time in a training model, and Maximum Memory Utilization. As the result of this paper, the XBGoost has the best performance by achieved 0.86 in AUC and costing the smallest memory utilization.

In the paper, A Comparative Test of Two Employee Turnover Prediction Models [4], the authors applied logistic regression (logit model), probability regression (probit model) to predict employee turnover that is voluntary turnover. The features were Age, Gender, Race, Organization committee, and Job performance. In the training set, two-class labels are relatively even, and both models reached 95 % of accuracy to predict turnover. The researchers used the Quadratic Probablily Score to suggest the logit model has better performance because logit's QPS is slightly lower probit model's QPS.

In this paper, Prediction of Employee Turnover in Organizations using Machine Learning Algorithms [3], the author stated that a hybrid model suggested in this study was the combination with Taguchi Methods and the nearest neighborhood rule achieved the 87.85% of accuracy and it could successfully predict employees who have a tendency to quit their jobs. There were twenty features that were selected by conducting Taguchi Methods. The top 10 were gender, age under 22 years old, education college level, marriage, resident, part-time or full time, salary under 25000, department, position, and pre-experience.

In this paper, Evaluation of Employee Attrition by Effective Feature Selection using Hybrid Model of Ensemble Methods [5], the author stated Adaptive boosting had the best result in achieved 85.93% of sensitivity and 90.68% of specificity and boosting methods had better results compared with stacking methods, such as SVM, GLM, KNN, and decision trees. Adaptive boosting and Gradient Boosting achieved over 80% of sensitivity.

In this paper, Analyzing employee attrition using decision tree algorithm [2], the authors used C4.5 (J48), REPTree, and CART (SimpleCart) decision tree algorithms. They ranked the importance of the features and evaluated model performance by confusion matrix, precision, recall, F-measure, and AUC. The AUC was around 0.77 and the results explained that salary and length of service were important in attrition. Employees who worked in a long time and without increasing their salary were likely to leave their job. Employees had lower salaries for a while and realized that they weren't likely to get a raise, so they would leave the companies and looking for better-paid jobs.

3. Methodology

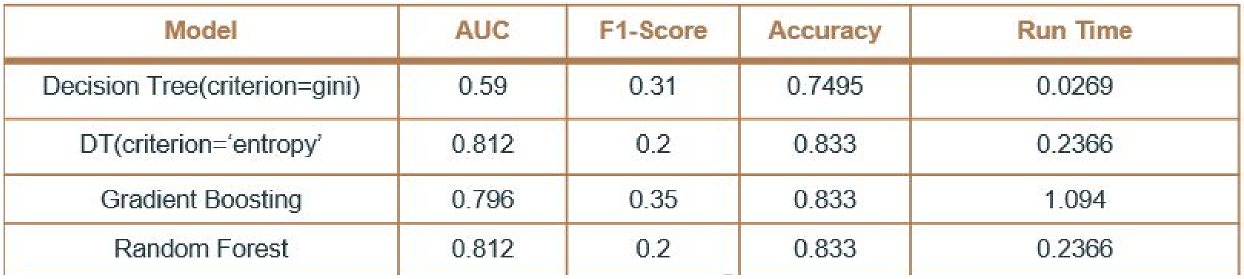
3.1 Data preprocessing

IBM HR Analytics Employee Attrition & Performance dataset on Kaggle has 34 features and the target variable, attrition. There are 1470 records and no missing value, so the main data cleaning is to remove redundant features and encode categorical features to 0 and 1. After encoding six categorical variables, 'BusinessTravel', 'Department', 'EducationField', 'Gender', 'JobRole', and 'marital status', the dataset becomes 51 features. There are 237 out of 1470 employees leave their job, which represents around 16% of the data. It has a seriously imbalanced issue in target- attrition.

3.2 Exploratory data

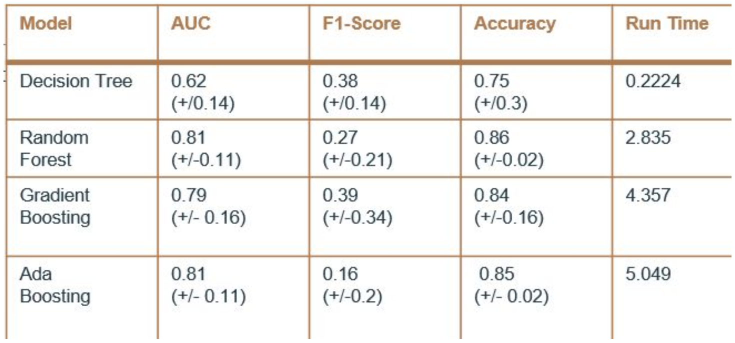
Using data visualization to compare the distribution and relationship between attrition and features, it can get a better understanding of data distribution and be further utilized in explaining models. From the distribution of monthly income and attrition, people whose monthly income are less than 5000 are likely to leave their job. From the distribution of age and attrition, people who are less than 35 years old are likely to leave their job. The correlation plot shows years at company, years in current role, years since last promotion, years with current manager are highly correlated.

Next, split 35% into a test set and 65% in a training set. 14.97% of the training data that Attrition =Yes, and 18.25% of the test data that Attrition = Yes. Decision Tree, Gradient boosting, Random Forest trained the on the training set, evaluated the result from applying the models on the test set. The F1-score and AUC were mainly focused on determining a model has good performance or not. The results showed that Gradient Boosting has the best F1- score and great AUC. The F1-score was 0.35 and AUC was 0.796. Monthly income is the most important feature in these four models. Testing feature importance by XGBClassifier got the most important feature is monthly income as well.



Model	AUC	F1-Score	Accuracy	Run Time
Decision Tree(criterion=gini)	0.59	0.31	0.7495	0.0269
DT(criterion='entropy'	0.812	0.2	0.833	0.2366
Gradient Boosting	0.796	0.35	0.833	1.094
Random Forest	0.812	0.2	0.833	0.2366

Comparing the train test split results with cross-validation results are relatively similar. Gradient Boosting had the best F1 score.



Model	AUC	F1-Score	Accuracy	Run Time
Decision Tree	0.62 (+/0.14)	0.38 (+/0.14)	0.75 (+/0.3)	0.2224
Random Forest	0.81 (+/-0.11)	0.27 (+/-0.21)	0.86 (+/-0.02)	2.835
Gradient Boosting	0.79 (+/- 0.16)	0.39 (+/-0.34)	0.84 (+/-0.16)	4.357
Ada Boosting	0.81 (+/- 0.11)	0.16 (+/-0.2)	0.85 (+/- 0.02)	5.049

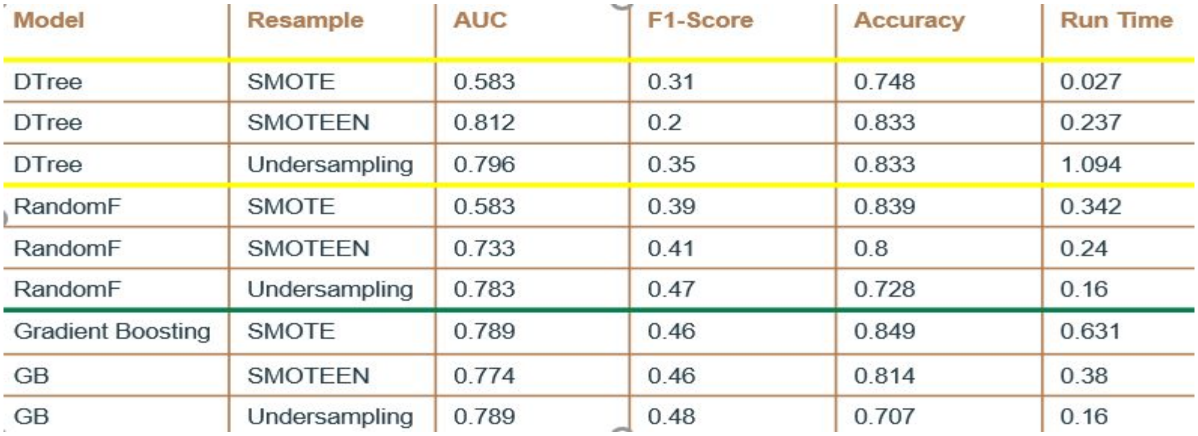
3.3 Resampling target in the training set

Smote method:
Target in the training set became {0: 812, 1: 812}. This method increased the case of attrition = Yes. It can be very optimal because it is less likely 50 % of the employees in an organization have a tendency of leaving their jobs.

SMOTEENN method:
Target in the training set became: {1: 629, 0: 435} This method increased the case of attrition = Yes but used the near neighbor cases to drop the case of attrition = No. It is combined with oversample and undersample ideas.

Undersample method:
Target in the training set became:{0: 143, 1: 143} This method can eliminate some noise in the dataset. It is a recommended approach to deal with an imbalanced dataset.

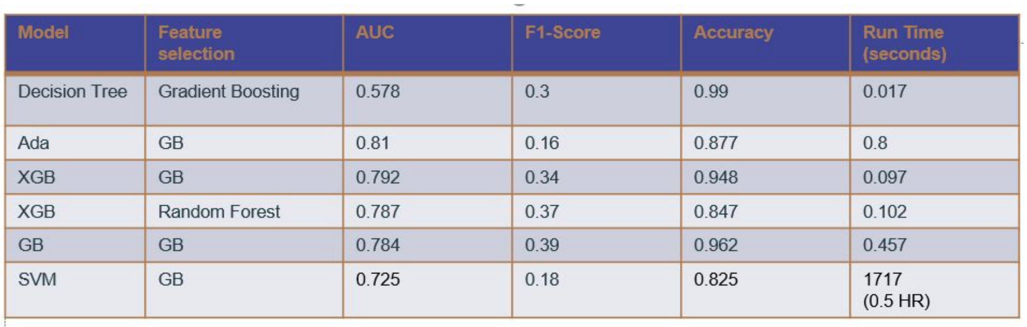
The results show that applied Gradient Boosting on the undersampled data set had the best result because its F1-score on test set achieved 0.48. It still obtained a great AUC, 0.789.



Model	Resample	AUC	F1-Score	Accuracy	Run Time
DTree	SMOTE	0.583	0.31	0.748	0.027
DTree	SMOTEEN	0.812	0.2	0.833	0.237
DTree	Undersampling	0.796	0.35	0.833	1.094
RandomF	SMOTE	0.583	0.39	0.839	0.342
RandomF	SMOTEEN	0.733	0.41	0.8	0.24
RandomF	Undersampling	0.783	0.47	0.728	0.16
Gradient Boosting	SMOTE	0.789	0.46	0.849	0.631
GB	SMOTEEN	0.774	0.46	0.814	0.38
GB	Undersampling	0.789	0.48	0.707	0.16

Monthly income frequently appeared on the top 3 important features in these models.

3.4 Feature selection and modeling:
Two feature selections were used Gradient boosting and Random forests on implement on the resampled and resampled dataset.
The following result showed the feature selection before resampling the target and build the model by a decision tree, Ada, XGB, Gradient boosting, and SVM.



Model	Feature selection	AUC	F1-Score	Accuracy	Run Time (seconds)
Decision Tree	Gradient Boosting	0.578	0.3	0.99	0.017
Ada	GB	0.81	0.16	0.877	0.8
XGB	GB	0.792	0.34	0.948	0.097
XGB	Random Forest	0.787	0.37	0.847	0.102
GB	GB	0.784	0.39	0.962	0.457
SVM	GB	0.725	0.18	0.825	1717 (0.5 HR)

The SVM has the worst result in runtime and F1-score. Boosting methods had better performance overall because their AUC reached 0.78. XGB and Gradient boosting had above 0.34 F1-score.

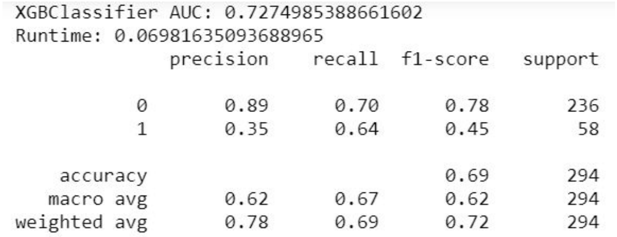
The following feature selection results were from utilizing gradient boosting and random forests on undersampled training data.
1. Gradient boosting selected featues are 'Age', 'DailyRate', 'DistanceFromHome', 'EnvironmentSatisfaction', 'HourlyRate', 'JobInvolvement', 'MonthlyIncome', 'MonthlyRate', 'OverTime', 'PercentSalaryHike', 'RelationshipSatisfaction', 'StockOptionLevel', 'TotalWorkingYears', 'WorkLifeBalance', 'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion', 'Department_Sales', and 'JobRole_Sales Representative'.
Features (total/selected): 50 19

2. Random forests selected features are 'Age', 'DailyRate', 'DistanceFromHome', 'Education', 'EnvironmentSatisfaction', 'HourlyRate', 'JobInvolvement', 'JobLevel', 'MonthlyIncome', 'MonthlyRate', 'NumCompaniesWorked', 'OverTime', 'PercentSalaryHike', 'RelationshipSatisfaction', 'StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear', 'WorkLifeBalance', 'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion', and 'YearsWithCurrManager'.
Features (total/selected): 50 22

These results showed that Gradient boosting selected 19 features and random forests selected 22 features, so Gradient boosting selected features could be used in modeling since it oculd reduce model complexity. In modeling process, gradient boosting had the best model performance in general. XGB was the running up. The undersampling approach was better than the oversampling method in this dataset. Monthly Income, Year with current manager, Totally working years, Environment satisfaction, Distance from Home, and Age occurred frequently in top 15 importance features in many models’ visualization of ranked importance features.

4. Results

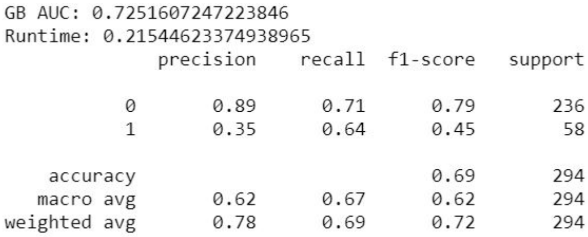
The boosting algorithms had the best performance in modeling. Their F1-scores were above 0.3, and 19% of data in test set was Attrition = Yes. The performance was better than random guessing, and the AUC achieved an average of 0.79. Resampling imbalanced could let the model learning less noise. 19 features were selected by gradient boosting on the undersampled data. SVM had the worst model performance in F1-score and run-time especially when the model has many features. The classification report of the test set from XGBoost which contacted gradient boosting selected features from an under-sampled dataset gave the insight of model performance, the f1-score achieved 0.45, and the AUC was 0.727. It contained 19 selected features and was short in run-time. Furthermore, 19% of the test set were labeled as Attention =Yes. Its F1- score was above 0.19, so it meant there are some useful information can be gathered from this model. Top 11 important features from XGBoost are Overtime, Department_sales, Total Work Years, Worklife balance, JobInvolvement, Environment satisfaction, Years at company, Monthly Income, Stock Option Level, and Distance from Home. The classification report of the test set from the gradient boosting model which contacted gradient boosting selected features from an under-sampled dataset showed the f1-score was 0.45, and the AUC was 0.725. It contained 19 selected features and run-time was 0.215. Its F1-score was also above 0.19, so it meant some useful information can be collected from this model. Top 11 important features from gradient boosting are Monthly Income, Overtime, Age, Daily rate, JobInvolvement, Stock Option Level, Years at company, Distance from Home, Monthly rate, Department_sales, and Environment satisfaction.



```
XGBClassifier AUC: 0.7274985388661602
Runtime: 0.06981635093688965
              precision    recall  f1-score   support

           0       0.89      0.70      0.78       236
           1       0.35      0.64      0.45        58

    accuracy                           0.69       294
   macro avg       0.62      0.67      0.62       294
weighted avg       0.78      0.69      0.72       294
```



```
GB AUC: 0.7251607247223846
Runtime: 0.21544623374938965
              precision    recall  f1-score   support

           0       0.89      0.71      0.79       236
           1       0.35      0.64      0.45        58

    accuracy                           0.69       294
   macro avg       0.62      0.67      0.62       294
weighted avg       0.78      0.69      0.72       294
```

After visualizing a tree from the output of the XGBoost model, it shows that employees whose monthly income is less than $ 3018 are likely leaving their current job. The next feature is 'Overtime < 1' means employees don't work overtime have the tendency to leave the organization. Moreover, employees who stay at the company for less than five years are likely to leave.

Visualized a tree from the output of the Gradient boosting model, it shows that employees who don't work overtime, are not sales representatives, but have low job involvement have the tendency to leave the organization.

5. Discussion

This type of human resource dataset involved people's concern about ethics. It is hard to collect data. Some researchers used another similar dataset to predict employees' turnover and work performance. The study of machine learning in human resource analysis can use in a beneficial way to organizations, so they could aware of improving promotion strategy to retain great employees. It might be utilized in predicting employees' work performance. It could potentially help organizations maintain a healthy turnover rate.

6. Conclusions and future work

It is very challenging to make a prediction on employees who are likely to leave the jobs because this dataset has a huge imbalanced issue and the features are numeric and categorical. It is difficult to handle data cleaning, encode categorical features, and resample the target. The accuracy should not be emphasized in the evaluation of model performance because it can miss leading the audience. The undersampling method improved model performance and boosting methods had great model performances. Most important of all, the results lined up the relationship between feature and target (attrition= Yes or No) in the exploratory stage. Employees who have lower monthly income and job involvement are likely to leave an organization, so organization can rethink promotion strategy.

Future work can use unsupervised learning algorithms such as k-nearest neighbors to see the feature relationships without setting bias of predict attrition since any different approach in this paper could not get an F1-score higher than 0.5 and AUC could not reach 0.85. Maybe this dataset can use predict other variables than attrition. Also, it can try the different feature selection methods, for example, the filter method. It can also build a deep learning neural network models.

Reference

[1] Ajit, P. (2016). Prediction of employee turnover in organizations using machine learning algorithms, *algorithms* 4(5): C5.

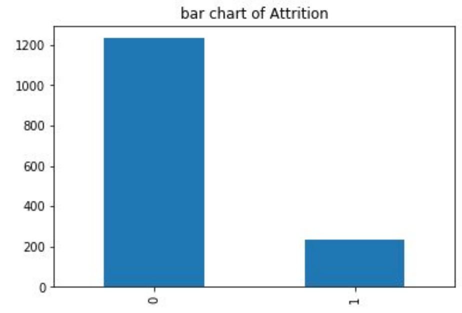
[2] Alao, D., & Adeyemo, A.B. (2013). Analyzing employee attrition using decision tree algorithm, *Computing, Information Systems, Development Informatics and Allied Research Journal*.

[3] Chang, H. (2009). Employee turnover: a novel prediction solution with effective feature selection*, Proceedings of the 3rd WSEAS international conference on Computer engineering and applications.*

[4] Hong, W., Wei, S., & Chen, Y. (2007). A comparative test of two employee turnover prediction models. *International Journal of Management, 24*(4), 808-821.

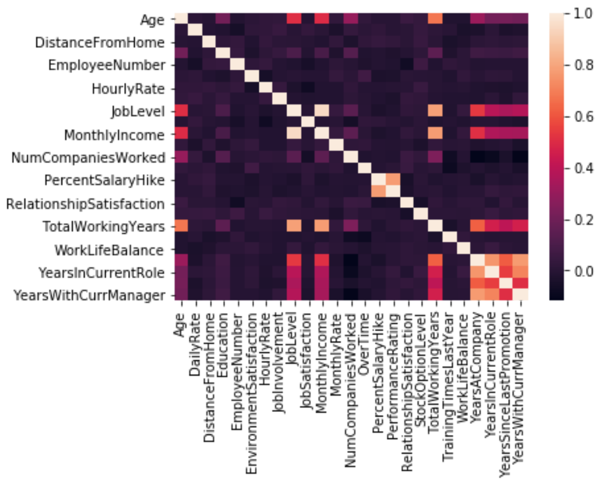
[5] Jain, D. (2017). Evaluation of Employee Attrition by Effective Feature Selection using Hybrid Model of Ensemble Methods.

Appendix



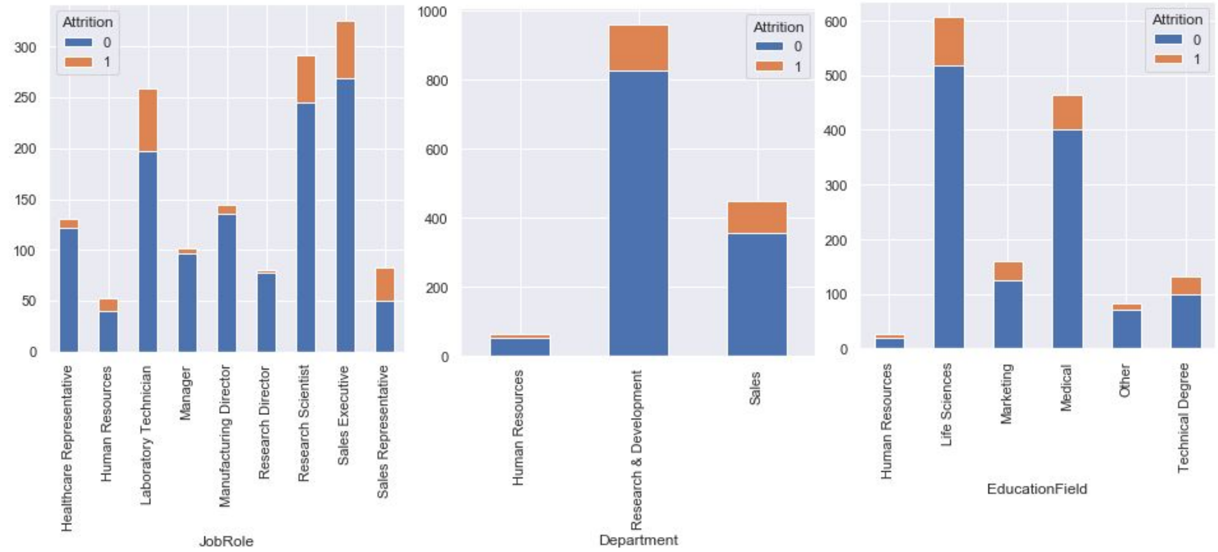
```
0    1233
1     237
Name: Attrition, dtype: int64
```

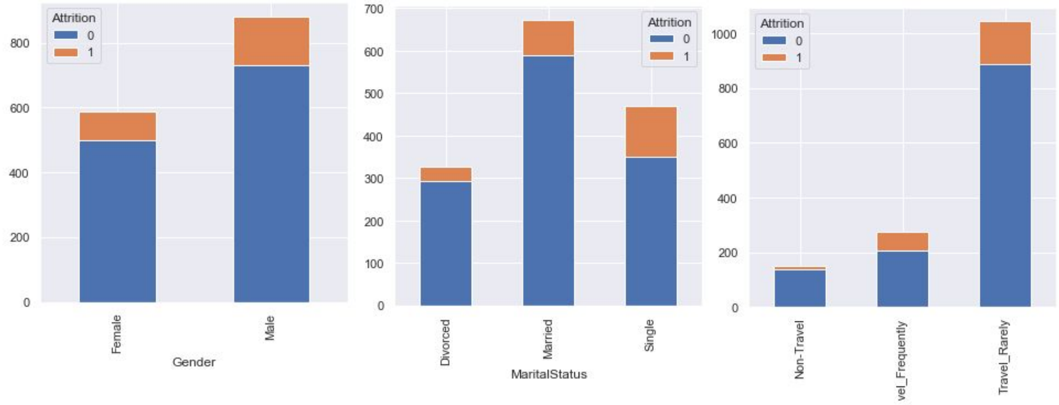
[1]



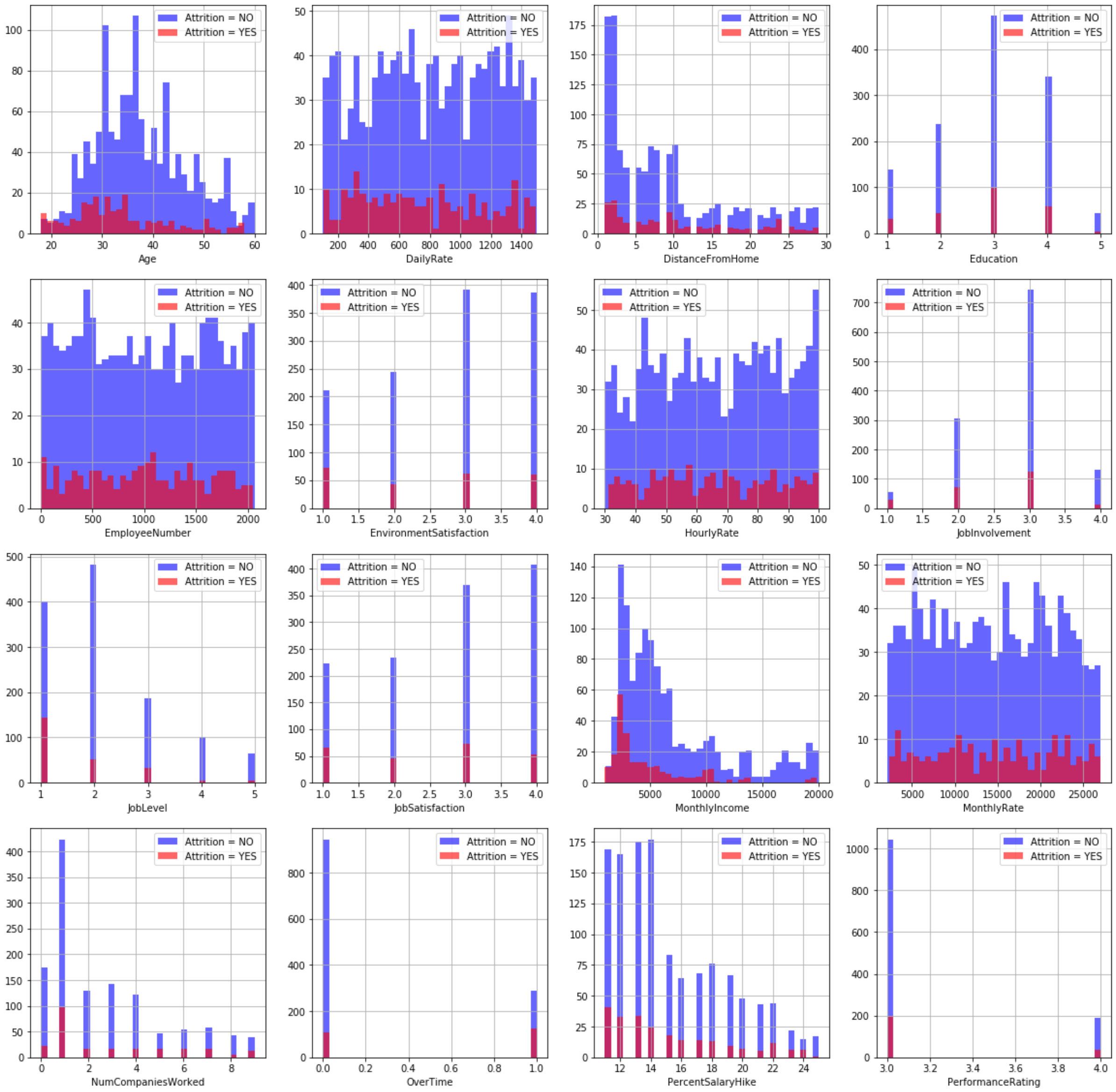
[2]

[3]

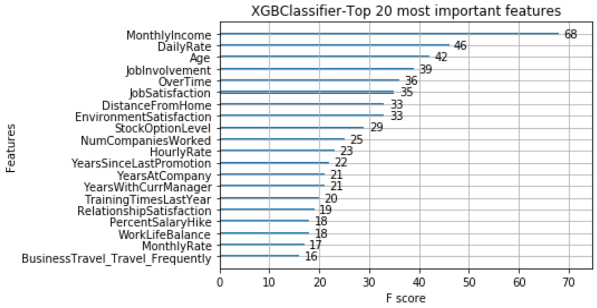




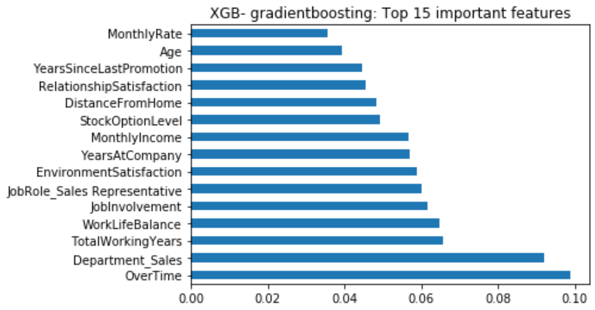
[4]

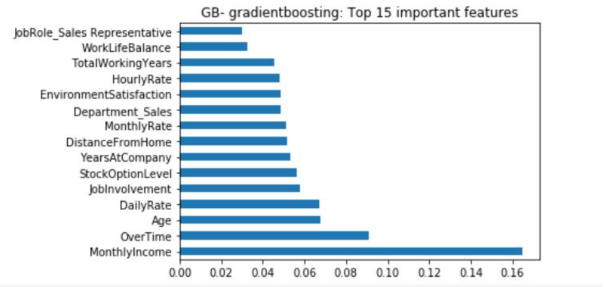

Attrition = NO
Attrition = YES
Age
DailyRate
DistanceFromHome
Education
EmployeeNumber
EnvironmentSatisfaction
HourlyRate
JobInvolvement
JobLevel
JobSatisfaction
MonthlyIncome
MonthlyRate
NumCompaniesWorked
OverTime
PercentSalaryHike
PerformanceRating

[5]

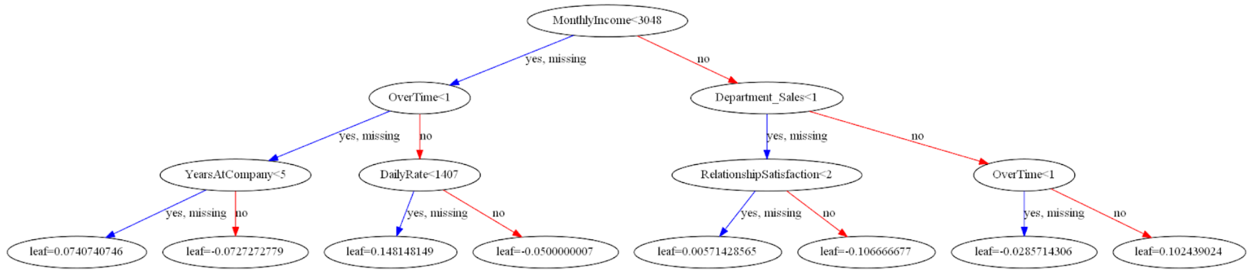

XGBClassifier-Top 20 most important features
Features
F score
MonthlyIncome 68
DailyRate 46
Age 42
JobInvolvement 39
OverTime 36
JobSatisfaction 35
DistanceFromHome 33
EnvironmentSatisfaction 33
StockOptionLevel 29
NumCompaniesWorked 25
HourlyRate 23
YearsSinceLastPromotion 22
YearsAtCompany 21
YearsWithCurrManager 21
TrainingTimesLastYear 20
RelationshipSatisfaction 19
PercentSalaryHike 18
WorkLifeBalance 18
MonthlyRate 17
BusinessTravel_Travel_Frequently 16
0
10
20
30
40
50
60
70

[6]

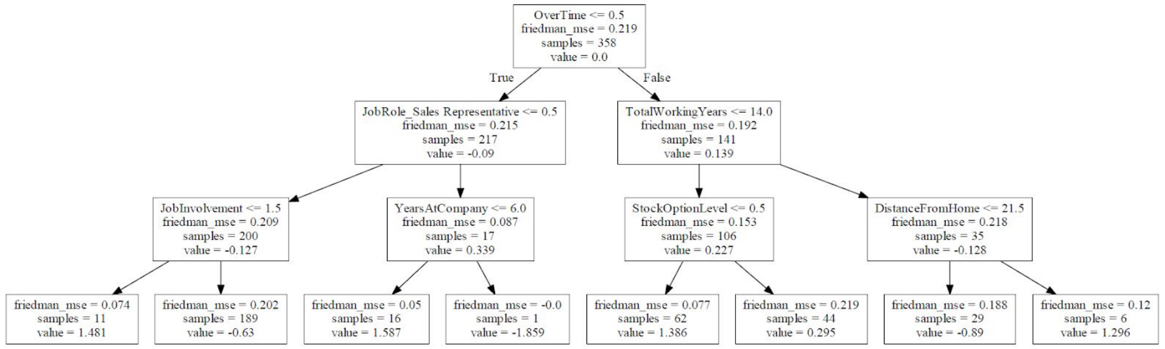




#feature selection using gradient boosting classifier +UNDERSAMPLING   Modeling by XGBClassifier()
[7]



#gradientboosting using gradient boosting selected features from the undersampling dataset -Modeling by gradient boosting
[8]


OverTime <= 0.5
friedman_mse = 0.219
samples = 358
value = 0.0
True
False
JobRole_Sales Representative <= 0.5
friedman_mse = 0.215
samples = 217
value = -0.09
TotalWorkingYears <= 14.0
friedman_mse = 0.192
samples = 141
value = 0.139
JobInvolvement <= 1.5
friedman_mse = 0.209
samples = 200
value = -0.127
YearsAtCompany <= 6.0
friedman_mse = 0.087
samples = 17
value = 0.339
StockOptionLevel <= 0.5
friedman_mse = 0.153
samples = 106
value = 0.227
DistanceFromHome <= 21.5
friedman_mse = 0.218
samples = 35
value = -0.128
friedman_mse = 0.074
samples = 11
value = 1.481
friedman_mse = 0.202
samples = 189
value = -0.63
friedman_mse = 0.05
samples = 16
value = 1.587
friedman_mse = -0.0
samples = 1
value = -1.859
friedman_mse = 0.077
samples = 62
value = 1.386
friedman_mse = 0.219
samples = 44
value = 0.295
friedman_mse = 0.188
samples = 29
value = -0.89
friedman_mse = 0.12
samples = 6
value = 1.296