Attrition Analysis of IBM

Authors:
Xiao Li (xl998)
Yi Nian (yn778)
Xuan Wang (xw1452)
Jingrou Wei (jw5238)

Abstract:

In this project, we construct correlations among factors, analyze important features, and implement and evaluate several models to predict the attrition rate in IBM. We will try different classifiers to predict the employees' probability of attrition and predicate performances of models by receiver operating characteristic (ROC). Moreover, based on the best model, the project will find certain ways to minimize the attrition rate.

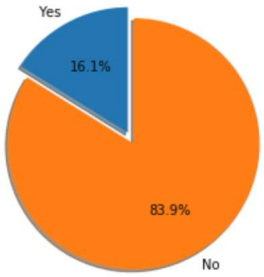
I. Introduction and Motivation

In human resources context, attrition refers to the gradual reduction of the size of employee by not replacing personnel lost through retirement or resignation. Although bringing in new talented employees will benefit for renovating technology and notion of the company, it takes risk by losing high-valued and high-performing employee. Therefore, proper strategies are required to control the growing employee attrition rate. However, factors contributed to the attrition are not simplex, this project will explore the dataset, analyze attrition distribution of these factors; moreover, it will implement and evaluate models to predict the attrition rate. Finally, we hope to find the best plan to minimize employee attrition.

II. Methodology

1. Data Visualization

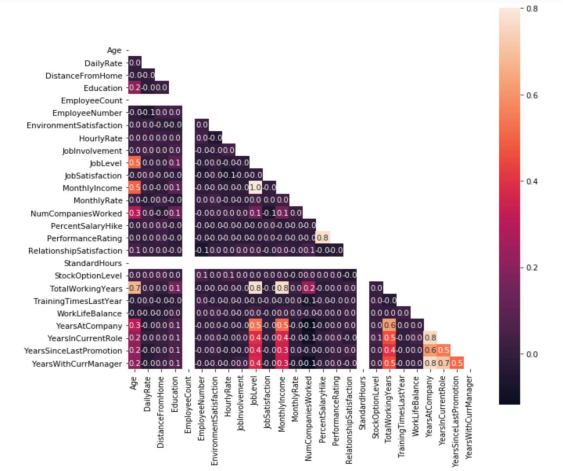
The dataset we used is IBM HR Analytics Employee Attrition & Performance data[1] provided by kaggle. The data involves the personal information, job conditions, as well as attrition status.



Pie Chart of Attrition Rate

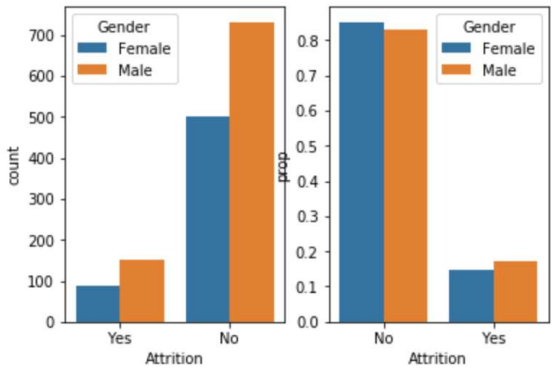
After affirming that there is no missing data, we first check the attrition rate and attrition distribution of factors to get a general understanding of the attrition rate of the variables. Since our target variable, Attrition, is a categorical feature we map it to numerical values. With a number of 1470 groups of data, there are 237 employees left the company, take the 16.1% of the total as shown in the pie chart. This ratio is higher than 10%, the average turn off rate suggested by the labor expert, and also higher than 13.2%, the turn off rate of technology companies calculated by LinkedIn.[2]

To get better understanding of the relationship among features, we construct a heatmap to show the correlation between each two variables. From the correlation matrix, we



find most of the features are uncorrelated. However, there is still a high correlation between some features, such as Monthly Income and Job Level(1.0). It is absolutely reasonable because if an employee has higher job level, then he or she well worth the higher income.

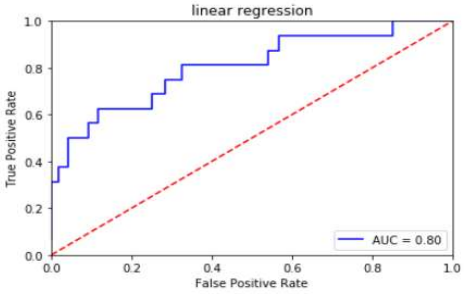
Since most features are uncorrelated, we would like to study the relationship between attrition rate and each feature by construct attrition distributions with each factors. For example, if we only consider Gender as the factor. We use seaborn to plot bar graph of attrition count and rate of female and male employees. The graph indicates that higher proportion of males are more likely to leave compared to females. Similar method is applied to other features so that we can see how attrition is distributed across each feature.



2. Statistical Method

We first tried to fit our model based on a linear model. By deleting the factors that are not statistically significant, the R square increased but not much. Since each factor did not have significant impact of each other according to the correlation graph. We observed the attrition distribution with each significant factors and found that some features have a non-linear relation with attrition, so we tried to add more nonlinear terms into the model.

The final model includes all the significant terms which leads to a high AUC value. However, since we keep changing this model manually based on its performance solely on this certain dataset, it may not be general enough.



3. Machine Learning Method

We first need to do some data cleaning by removing variables that are the same all the time and not meaningful, such as ‘over 18’ and ‘Standard Hour’. Then, we assign an unique indicator for each categorical variables in the dataset, for example, we will replace the feature ‘Gender’ by ‘Gender_Female’ and ‘Gender_Male’. Finally, we make all the features in our dataset have mean 0 and std 1 because features that are in different scales could bring troubles for some classifiers like Support Vector Machine.

Another important thing is that we need to find a suitable evaluation metric for our models. Since our dataset is unbalanced, metrics like accuracy are not suitable here. After some discussion, we decided to use receiver operating characteristic curve as known as ROC curve as our evaluation metric. ROC curve illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied.[3] The ROC curve is created by plotting the correlation of the true positive rate (TPR) against the false positive rate (FPR). The larger the area under the ROC curve (AUC) is, the better prediction the model will make.

To optimize the performance of models, we combined random search and grid search technique to tune the parameters of the models.

The first model we used is Logistic Regression, a direct and powerful method for classification problem. After doing grid search,

the best hyperparameters are *L2 penalty*, *liblinear solver*, and regularization constant of 1.

The second model is Decision Tree, a non-parametric supervised learning method used for classification and regression. It predicts the value of a target variable by learning simple decision rules inferred from the data features[4]. We started with the baseline decision tree classifier and use grid search to find the best hyperparameters. The resulting hyperparameters are *'entropy' criterion, 'auto' max_features, 35 min_samples_leaf, and 80 min_samples_split.*

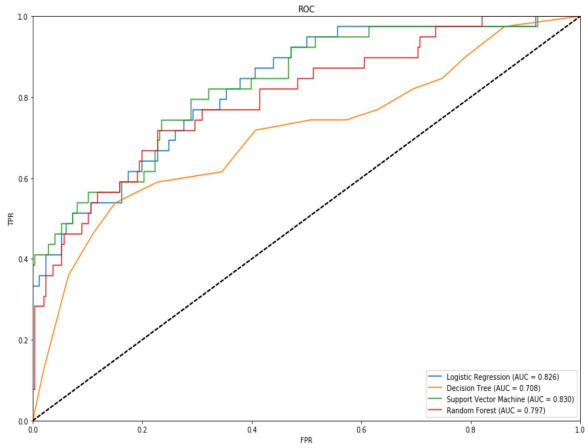
Based on Decision Tree, we also tried Random Forest, which is an ensemble of Decision Trees that could increases the overall result. We did grid search on *n_estimators, criterion, max_depth, min_samples_split, and min_samples_leaf.* And the set of hyperparameters is *n_estimators = 400, max_depth = 50, min_samples_split = 10, and min_samples_leaf = 1* with *'entropy' criterion.*

The last but not the least, we used Support Vector Machine (SVM) which could be used for both classification and regression analysis, a discriminative classifier formally defined by a separating hyperplane, that is given labeled training data, SVM will assign new examples to one category[5]. We tried different types of kernels and penalizing constants. As a result, we get the linear kernel with penalizing constants equals to 1.0.

The ROC curve of all the models after parameter tuning is shown in the figure below. From the plot, we could see that SVM has the highest AUC score at about 0.830.

4. Problems

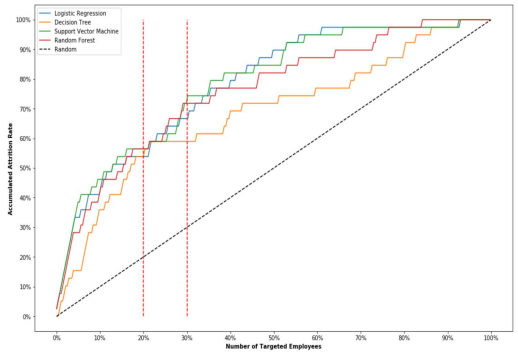
From the plot below, we could also see that Random Forest and Decision Tree didn't perform well. Since we know that these two classifiers tend to perform bad when there are too many features and too few samples, we decide to do a feature reduction and see if a smaller feature set could improve the results of the two classifiers.



We did feature reduction based on the feature importance provided by Mutual Information, which could provide a measure of the mutual dependence between each feature and the target variable. But after we did parameter tuning on the smaller dataset and plot the ROC curve for each model again, we notice that although the AUC score for Random Forest and Decision Tree improved a bit, they couldn't beat the performance of SVM on the full feature set. So we decided to still choose SVM as our best model.

5. Reducing Attrition Rate

Until now we could predict attrition fairly well, so it comes to our mind that we need to find some ways to help the company reduce attrition rate. So firstly we plot a cumulative gain curve.



From the graph we could see that if we target the top 30% of employees that are most likely to leave, we are actually targeting over 70% of the attrition population. This could give us a sense that how many employees that we should focus on to deal with the problem.

We thought of two approaches to deal with the attrition problem, the first is a **naive approach**: from the samples we choose some of the employees with highest attrition probability and change some features of the employees and then put the employees back to the model again to see if the estimated probability of attrition actually drops. And the second is an **interactive approach**: we could design a questionnaire which using the information from feature importance by Mutual Information and list some features that could be potentially leaving reasons for the employees to fill out, and change the employees' condition accordingly.

III. Results and Discussion

For the prediction part, we have chosen SVM as our best model since it has the best AUC score. We are happy with this result and understand that the reason SVM performs well is that SVM could handle large feature sets and tends to perform better on smaller and cleaner datasets which is exactly the case of our dataset. But we also know that SVM could be inefficient with large datasets and less effective on noisy datasets, so although for now this is not a problem since our dataset is fairly small with only 1470 instances, it could be an issue in the future if we could get more data and when the number of outliers starts to increase.

For the reducing attrition rate part, although we have a function that could change some features of an employee to lower the attrition probability, the difference is not very convincing as we expected. We think the reason is that firstly, we change features in an order based on feature importance given by Mutual Information which may not be compatible with SVM. Secondly, there are certain features that have high correlation with the target variable but we can't change them in real life settings, for example, Age and Total Working Years. Therefore we think that the interactive approach could be more appropriate but since we're not in the company, we didn't have the opportunity to give questionnaires to people and test this approach. Hence we can't conclude whether the approach is good or not.

IV. Conclusion

Attrition is unavoidable, from one aspect, it helps the turnover rate of the company; on the other hand, it may cause the loss of high-performance employees and thus influence efficiency of the operation and atmosphere of the workplace. Therefore, it is necessary to predict how an employee is likely to job-hopping and control the attrition rate and we think our model could do the prediction well but we still need some real company experiences to improve our approaches for reducing attrition rate.

V. References:

[1] "IBM HR Analytics Employee Attrition & Performance -Predict attrition of your valuable employees", Dataset, Kaggle.
[2] M.T. Wroblewski, "Negative Effects of Turnover", chron, Nov. 27, 2018.
[3] Receiver Operating Characteristic. *Wikipedia.*
[4] '1.10. Decision Trees', *Scikit Learn.*
[5] Savan Patel, "Chapter 2 : SVM (Support Vector Machine)—Theory", medium.com, May 3, 2017.