IBM MACHINE LEARNING

**Fatima, Sayeda**

**8/17/2022**

A picture containing automaton

Description automatically generated­­­-­­

SUPERVISED MACHINE LEARNING: CLASSIFICATION MODELS FOR EMPLOYEE ATTRITION

**Table of Contents**

[**1) Project Overview** 2](#_Toc111643034)

[**2) About the Dataset** 3](#_Toc111643035)

[**2a) Brief description of the data set you chose:** 3](#_Toc111643036)

[**2b) Summary of Data Attributes** 3](#_Toc111643037)

[**2c) Main Objectives of Analysis** 3](#_Toc111643038)

[**3) Data Exploration, Data Cleansing and Features Engineering** 4](#_Toc111643039)

[**3a) Data Exploration:** 4](#_Toc111643040)

[**3b) Data Cleansing Actions:** 6](#_Toc111643041)

[**3c) Features Engineering** 8](#_Toc111643042)

[**4) Summary of Training Different Classifier Models** 12](#_Toc111643043)

[**4a) Machine Learning Algorithm Approaches** 12](#_Toc111643044)

[**I) Data Level Approaches:** 15](#_Toc111643045)

[**II) Algorithm Ensemble Approach:** 15](#_Toc111643046)

[**4b) Summarizing Employed Models** 16](#_Toc111643047)

[**1) Logistic Regression (LR) Models:** 16](#_Toc111643048)

[**2) Random Forest Models:** 18](#_Toc111643049)

[**3) XGB Model** 24](#_Toc111643050)

[**5) Recommended Model** 25](#_Toc111643051)

[**5a) Result Summary** 25](#_Toc111643052)

[**5b) Overall Visual Summary** 25](#_Toc111643053)

[**5c) Individual Model Visual Summary** 26](#_Toc111643054)

[**5d) Model Choice and Justification** 26](#_Toc111643055)

[**6) Summary Key Findings and Insights** 27](#_Toc111643056)

[**6a) Summarizing Model Drivers:** 27](#_Toc111643057)

[**6b) Enlisting Top Contributory Factors** 27](#_Toc111643058)

[**6c) Visualizing Top Contributory Factors to Employee Attrition** 28](#_Toc111643059)

[**7) Link to Other Useful Models** 29](#_Toc111643060)

[**8) Github Link to Assignment Notebook** 29](#_Toc111643061)

# **1) Project Overview**

A fundamental issue facing organisations is attraction and retention of best talent. Given the cost of retraining new employees, it is important for a business to prevent loss of good talent. Hence, identification of key factors driving employee churning or turnover is important for the organization's Human Resource (HR) Department.

It is here that machine Learning models can be very useful to gain deeper insight into underlying factors and their relationship in driving employee turnover.

Hence, the main aim of the following machine learning modelling and analysis is to enable the business to:

\* To identify different factors predict employee churn

\* To gain insight into factors contributing to employee churning

\* To enable the business maximize employee attrition

# **2) About the Dataset**

## **2a) Brief description of the data set you chose:**

This project uses a hypothetical dataset 'IBM HR Analytics Employee Attrition & Performance' which was downloaded from the following link:

<https://www.kaggle.com/datasets/pavansubhasht/ibm-hr-analytics-attrition-dataset?resource=download>

## **2b) Summary of Data Attributes**

The dataset exhibits 1,470 data points (rows) and 35 features (columns) reflecting on employees' background and characteristics and can be downloaded from the following link:

The data also comes with ‘Attrition’ Column to show current employees and leavers which represents the Class we are trying to predict.

## **2c) Main Objectives of Analysis**

Organizational performance is largely dependent on its employees, their quality and experience. Hence, organizations are continuously faced with the challenge to reduce employee attrition and increase retention. Consequently, this analysis is targeted towards answering the following queries

* What are the various factors contributory to employee attrition?
* Which business units face higher employee attrition rate?

As a consequence, implementation of the model will enable the organization to:

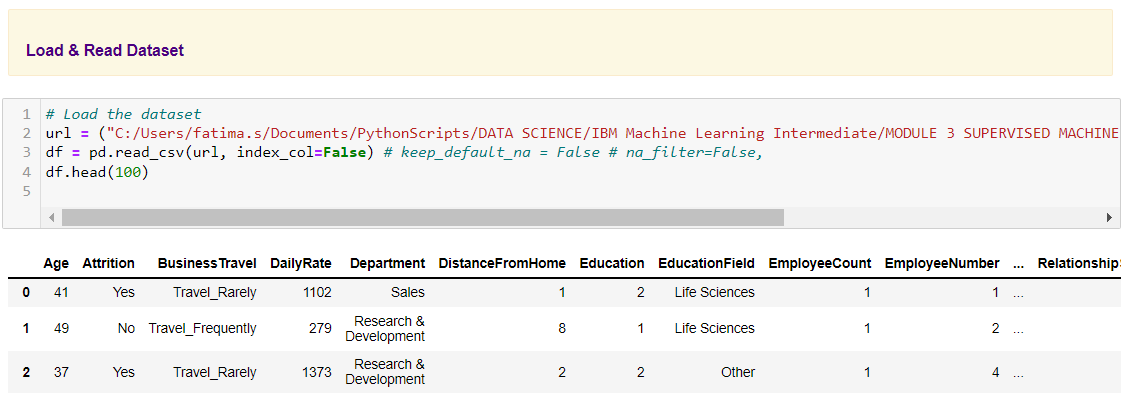
* devise suitable measures to increase employee retention
* to save valuable resources in retraining new employees hired in place of leavers

# **3) Data Exploration, Data Cleansing and Features Engineering**

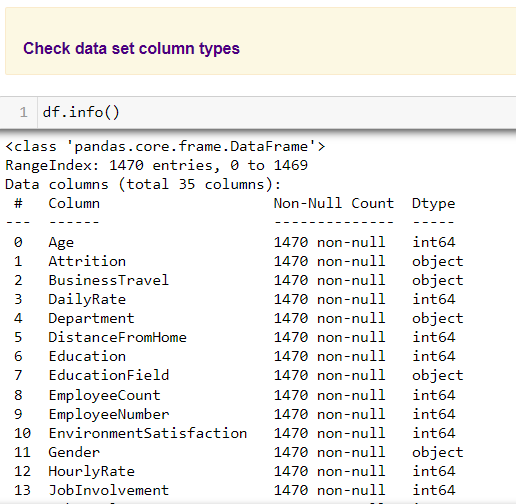
Since the quality of any machine learning model highly depends on quality of data, hence, this stage is not only most important but is also time consuming. Hence, it was conducted in a step-by-step process.

## **3a) Data Exploration:**

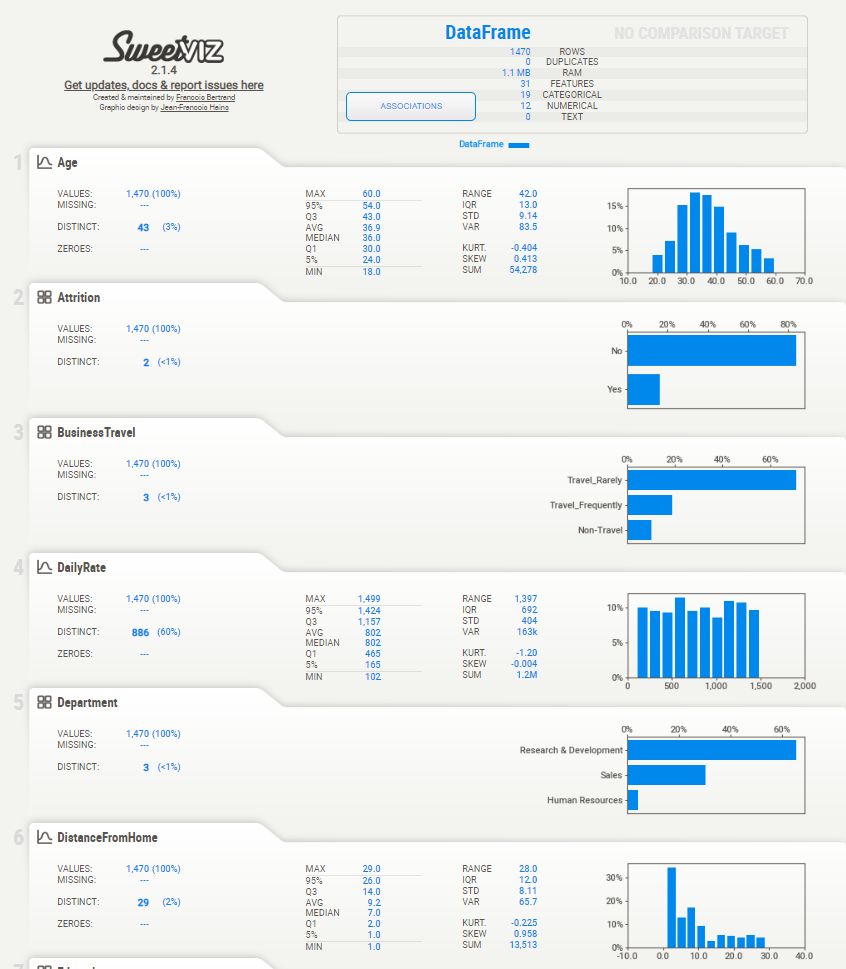
* Data was first loaded into pandas dataframe



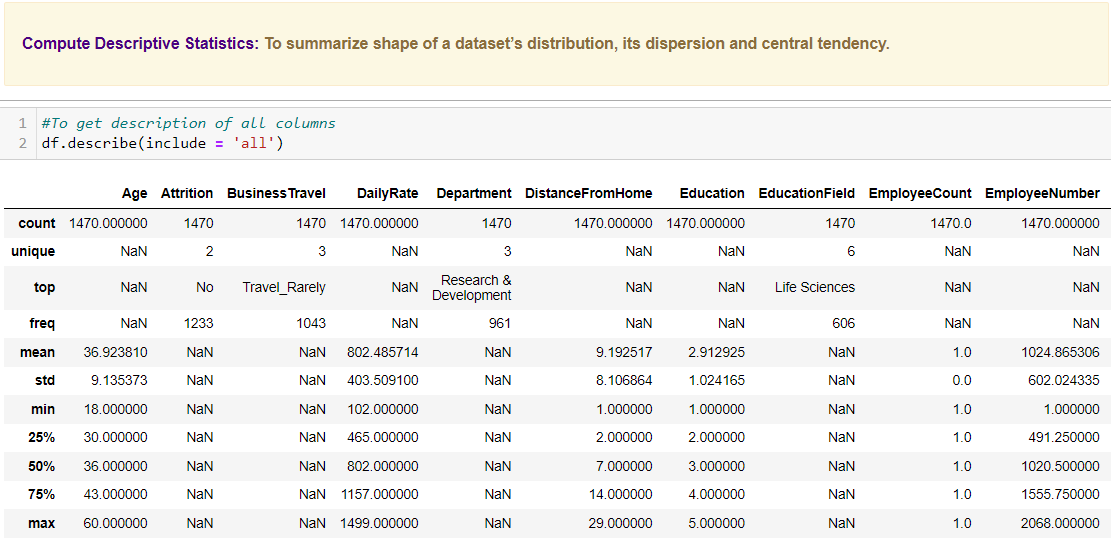
* Column types were explored



* Automated Exploratory Data Analysis was performed using Sweetviz to check

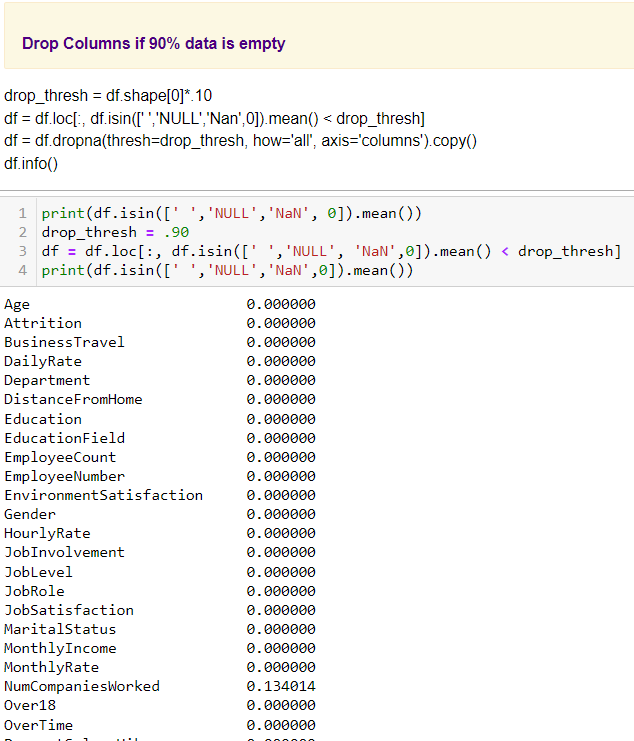


* Descriptive statistics were computed to summarize shape of a dataset’s distribution, its dispersion and central tendency

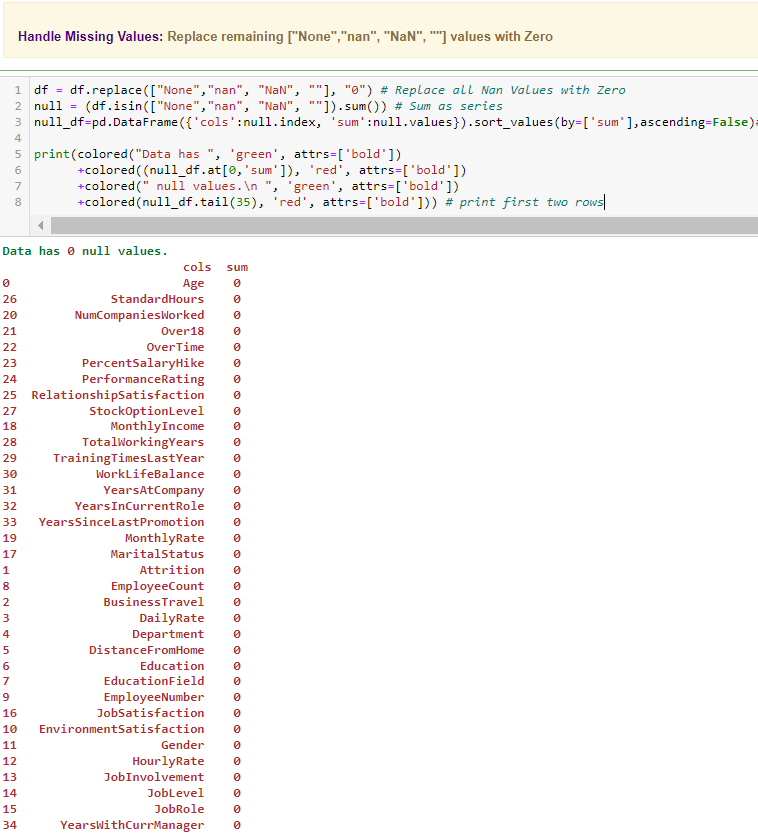


## **3b) Data Cleansing Actions:**

* Empty or nearly empty columns were removed using "drop\_thresh" to drop columns if 90% of data was empty



* Duplicates were dropped using pandas "df.drop\_duplicates()" method



* Null values were summed and Data was found to exhibit zero null values. Thus, no filling of null values was required

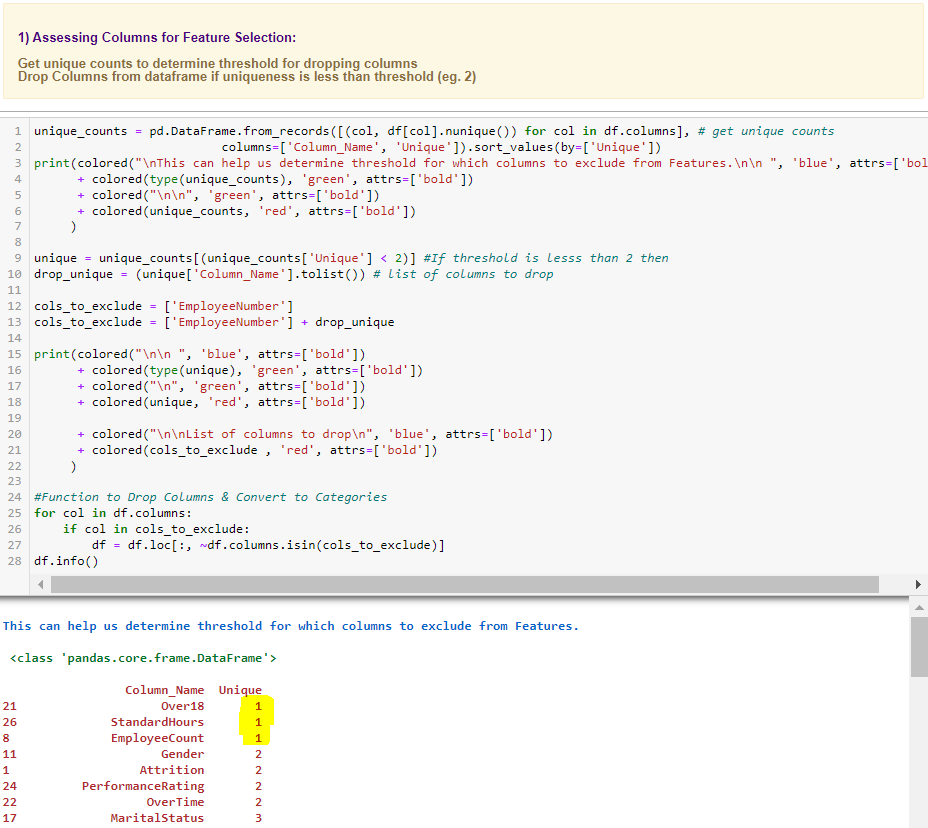
## **3c) Features Engineering**

In machine learning, feature selection is the method to reduce the number of input variables during developing predictive modelling. This reduction in input variables is necessary not only to minimize computational cost of modeling but also to achieve performance improvement of the model.

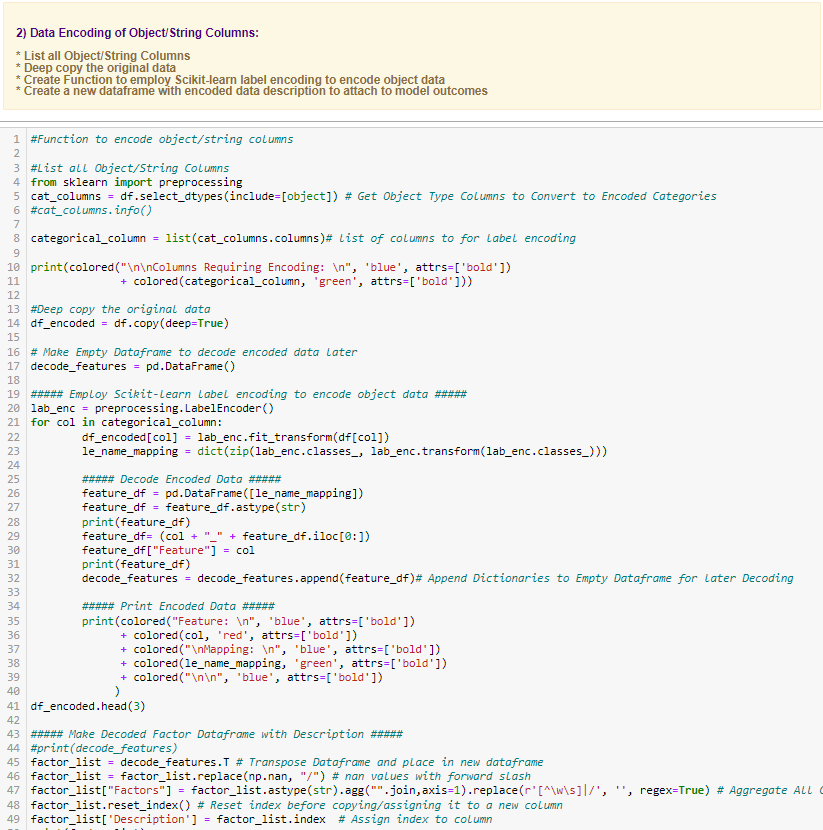
Among widely practices feature selection approaches include statistical-based feature selection methods which use statistical measures to evaluate relationship between each input variable and the target variable and then select those exhibiting strongest relationship with the latter. While these methods can be both speedy and effective, however, the ultimate choice of statistical measure is largely dependant on data types of both of these variables.

Irrespective of the statistical measure being employed, two dominant feature selection techniques, that is supervised and unsupervised, exist where the former can be further categorized into wrapper, filter and intrinsic techniques. Filter-based feature selection methods employs statistical measures to evaluate correlation between input and output variables so that those exhibiting highest correlations are selected. Statistical measures employed in filter-based feature selection are normally univariate in nature since they evaluate relationship of single input variables one by one with target variable, disregarding their interaction with each other.

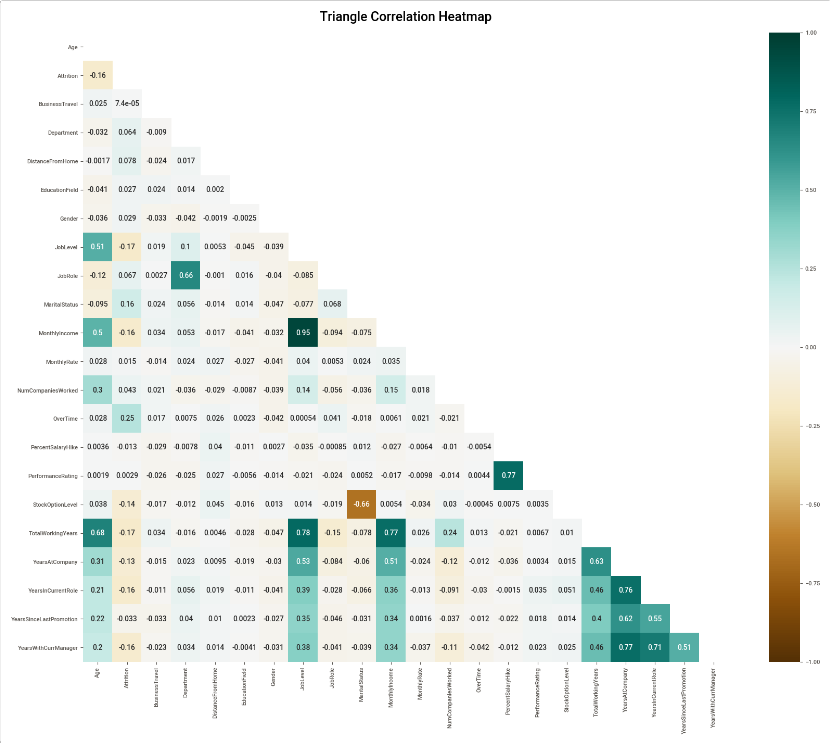
Consequently, adopting filter-based feature selection methods, the employee attrition model approached filter engineering in three steps. Firstly, unique values for all columns were computed after which columns with unique values less than 2 were dropped.



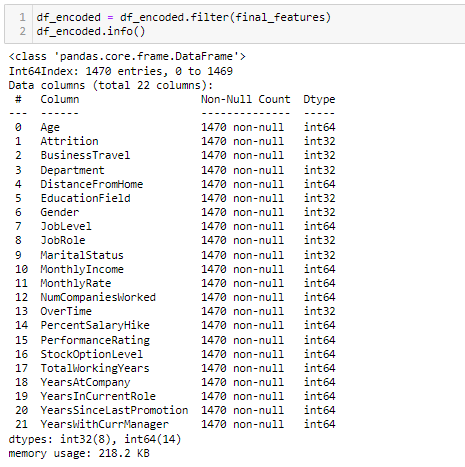
Prior to final features selection Data Encoding of Object or String Columns was carried out to facilitate any statistical computation during features selection process. Hence, after deep copying of original dataset, a function was created and employed to encode object data using Scikit-learn label encoder.



Statistical measures were then employed with supervised filter-based feature selection technique. Using Pearson's Correlation, the first set of features are selected based on the strength of positive correlation with taget variable 'Attrition'. Additionaly, Pearson's Correlation Matrix was also computed to select feature pairs exhibiting positive correlations with each other.



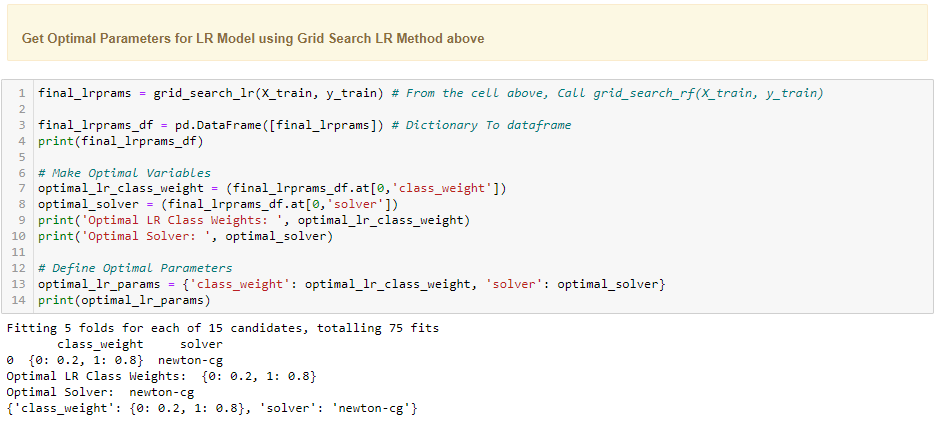
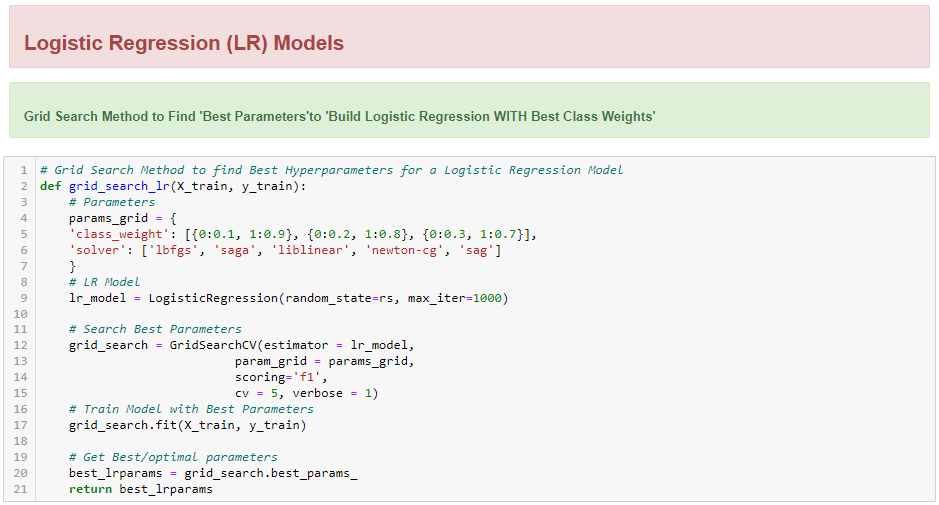
All feature lists were then combined to filter out dataframe columns not included in the ‘final\_features’ list.

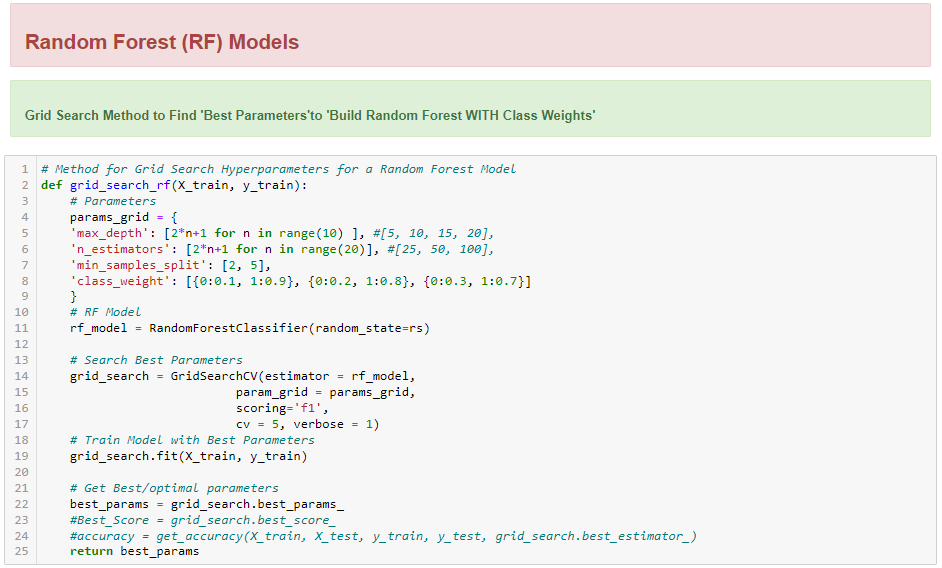


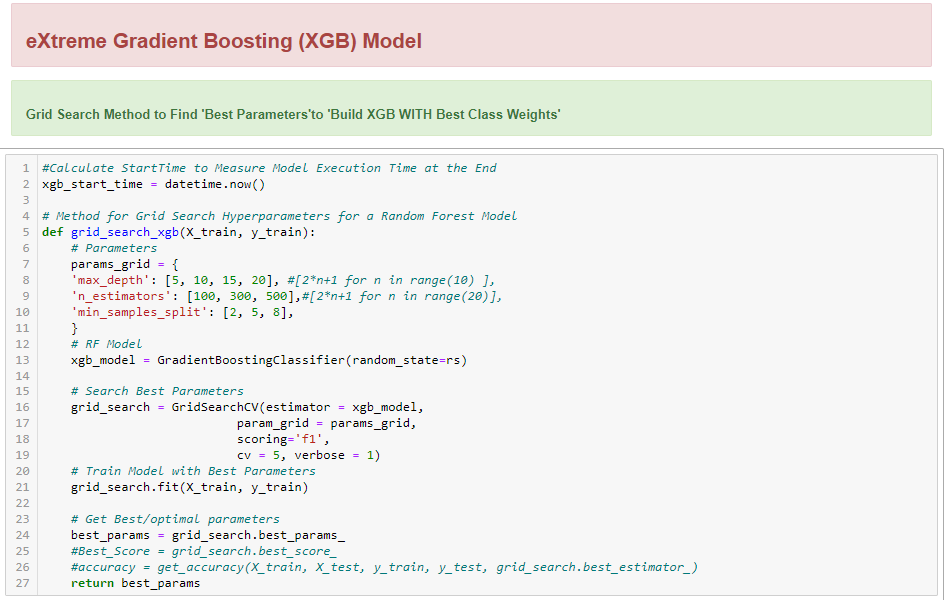
# **4) Summary of Training Different Classifier Models**

## **4a) Machine Learning Algorithm Approaches**

Since EDA revealed a highly imbalanced class distribution, this necessitated achieving appropriate class balancing. While data level and algorithm ensemble approaches do exist for dealing with imbalanced datasets, nevertheless, an automated optimal parameter search method was created to achieve best class reweighting along with isolating other optimal model parameters. This approach was employed because best hyper-parameters are not automatically learnt within estimators and its manual search not only slows down model development but may also lead to ineffective model construction. Hence, exhaustive cv grid search approach was used to pass parameter arguments to the constructor in order to find optimal parameters for each model.







Additionally, the following approaches were also combined with optimal parameters to find a model with best scores:

### **I) Data Level Approaches:**

**i) Synthetic Minority Over-sampling Technique (SMOTE):**

Due to highly imbalance class distribution, employee data contains very few instances of minority class for any classification model to explicitly learn decision boundary. A popular approach to tackle this problem is oversampling minority class examples which are close in the feature space using SMOTE. This approach allowed us to achieve a balanced class distribution.

**ii) Random under-sampling:**

Using random under-sampling examples from majority class were deleted to achieve a balanced class distribution.

**iii) Random over-sampling:**

Random oversampling was employed to duplicate examples from minority class to achieve a balanced class distribution.

### **II) Algorithm Ensemble Approach:**

1. **Boosting:**

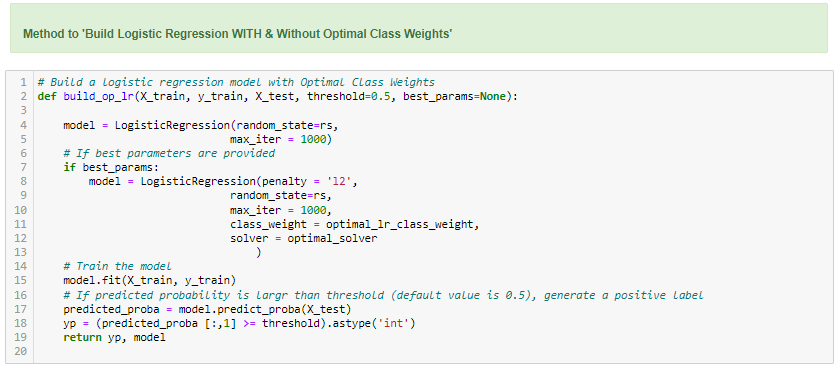
Using boosting, a sequential aggregate of base classifier was created on weighted versions of training data set which focused on misclassified samples at every stage of creating new classifiers based on sample weights that were altered as per classifier’s performance. Boosting was achieved using XGB Classifier.

## **4b) Summarizing Employed Models**

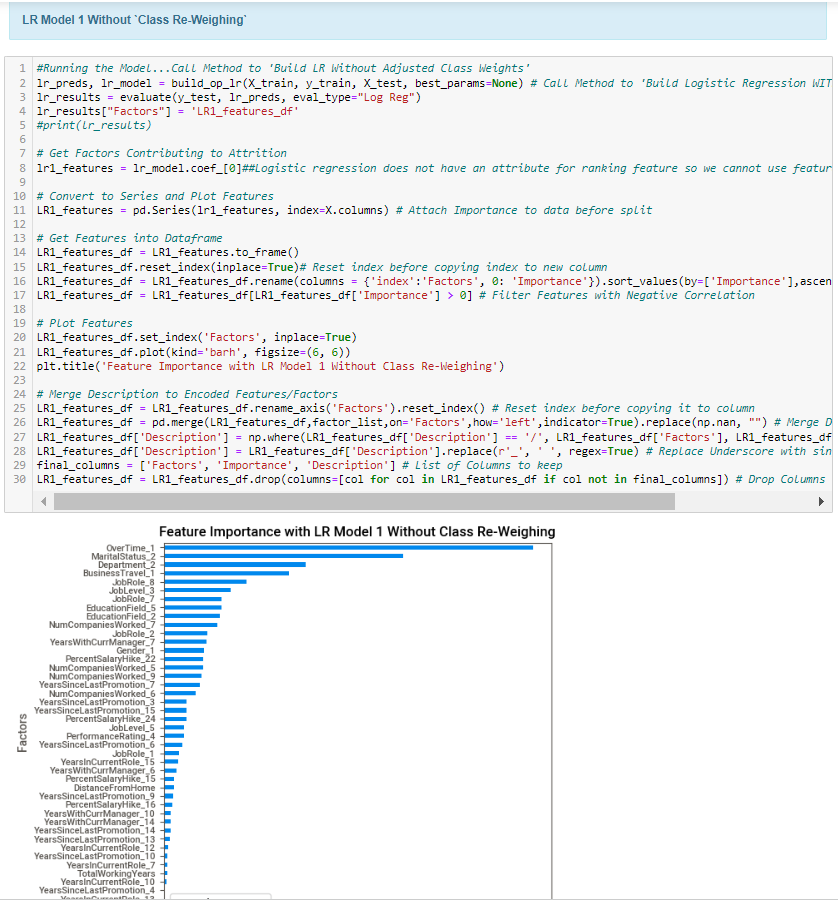
Following three main classifier models have been used to predict employee attrition.

### **1) Logistic Regression (LR) Models:**

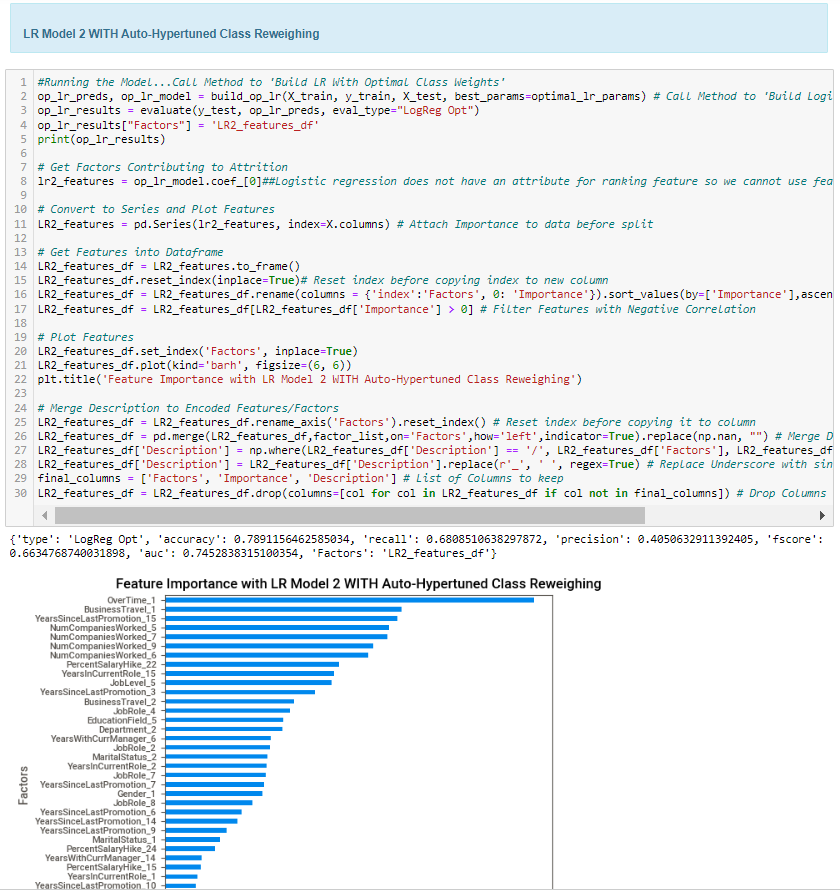
Due to the binary nature of predictive variable 'y' where the output can only result in whether the employees will fall in attrition class or not, logistic regression classification algorithm was employed. A single method was created to measure predictive capability of the following two LR models:



**1a) LR Model 1 Without Class Re-Weighing:** A simple algorithm was created without achieving any class balance or hyperparameter tuning.



**1b) LR Model 2 WITH Auto-Hypertuned Class Reweighing:** A modified algorithm with auto-hypertuned parameters was created using CV grid search method described above.

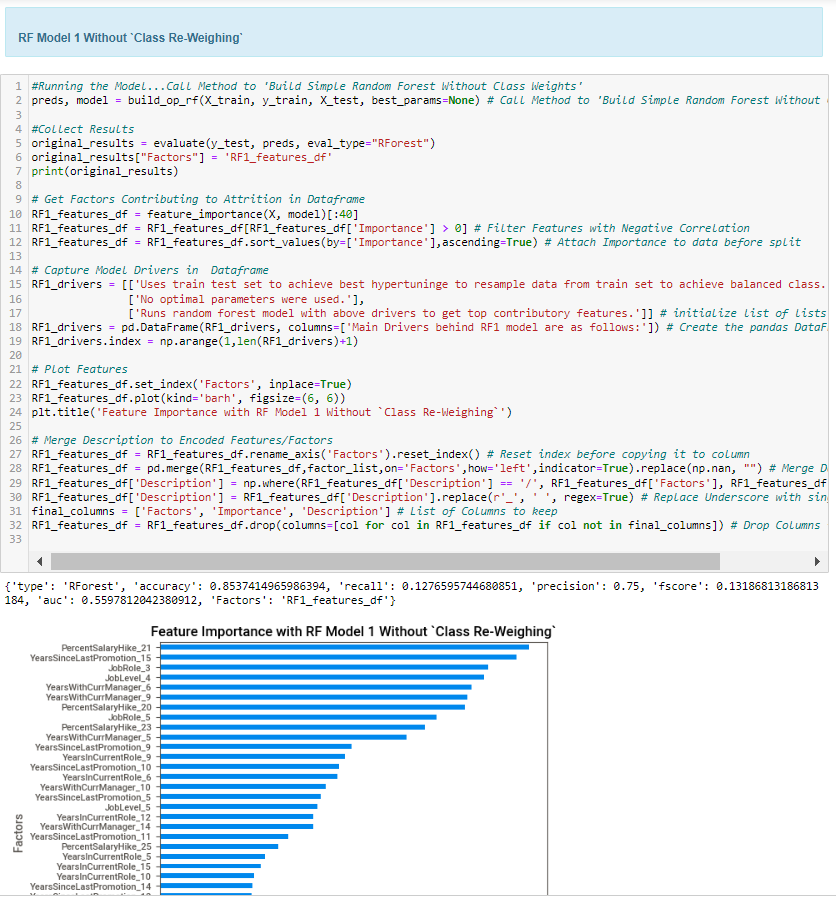


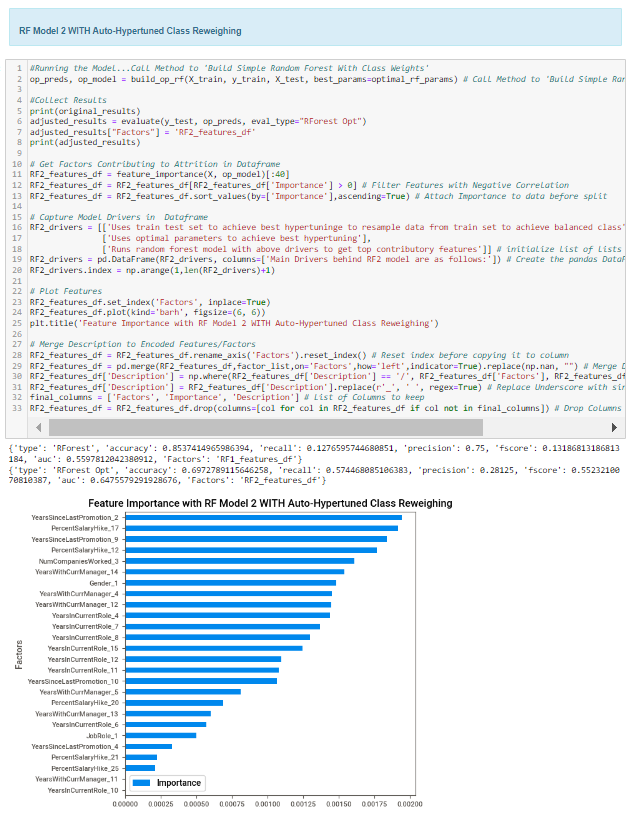
### **2) Random Forest Models:**

A single method was employed to run random forest models with and without optimal parameters and class weighing:

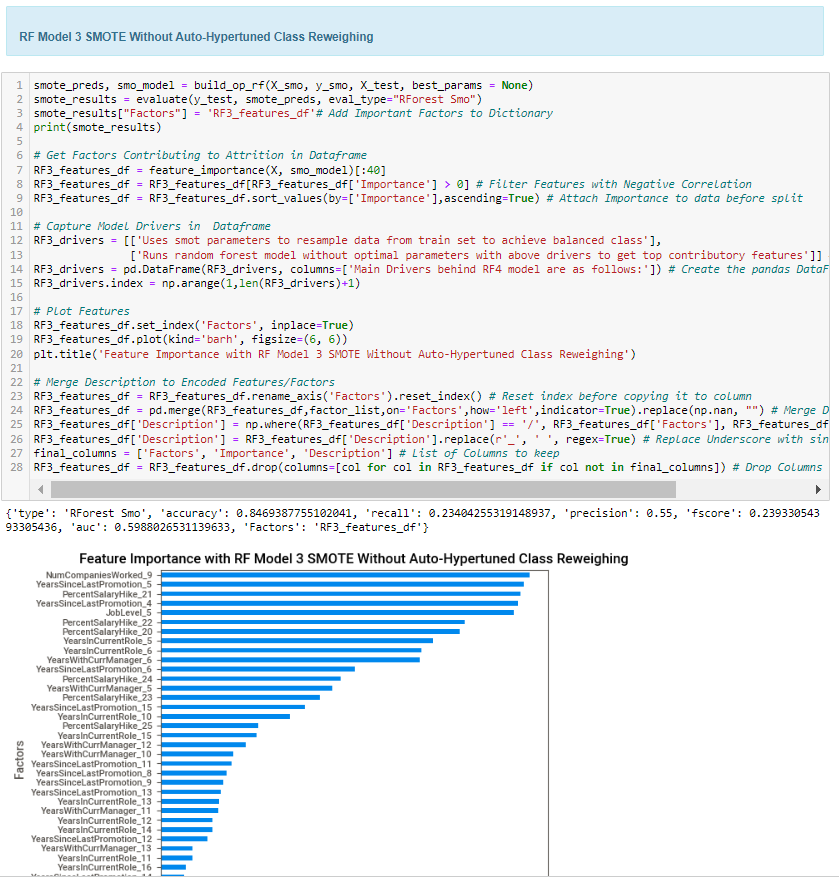


**2a) RF Model 1 Without Class Re-Weighing:** A simple algorithm was created without achieving any class balance or hyperparameter tuning.

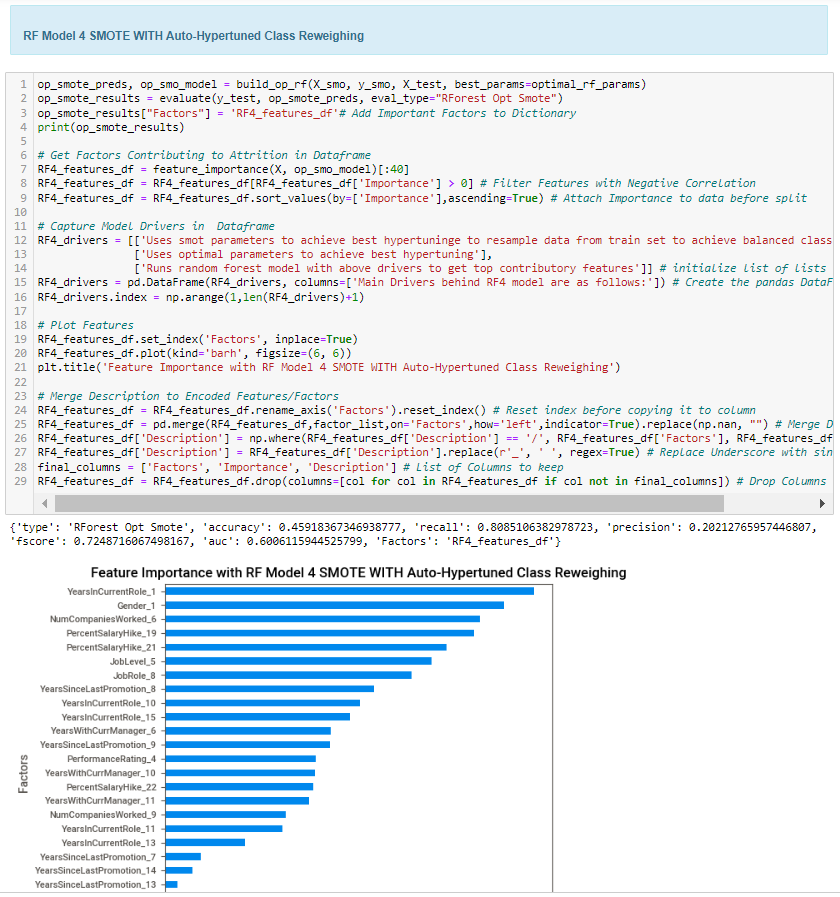


**2b) RF Model 2 WITH Auto-Hypertuned Class Reweighing:** A modified algorithm with auto-hypertuned parameters was created using CV grid search method described above.

**2c) RF Model 3 SMOTE Without Auto-Hypertuned Class Reweighing:** A simple algorithm was created without achieving any class balance or hyperparameter tuning which employed smote sampling to achieve class balance.

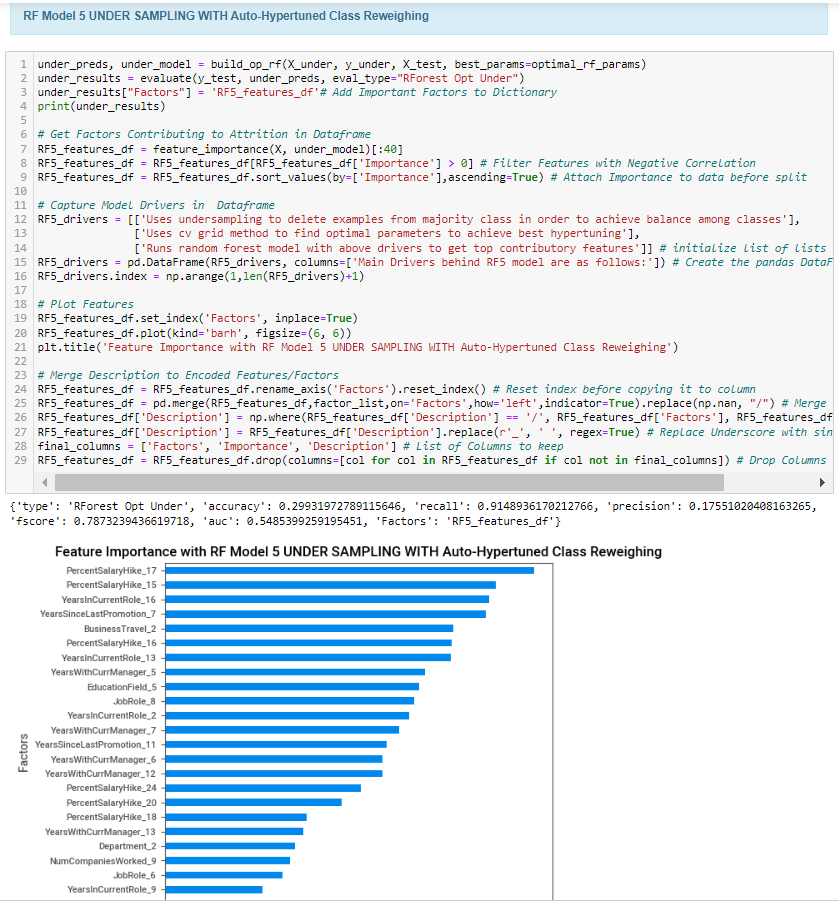


**2d) RF Model 4 SMOTE WITH Auto-Hypertuned Class Reweighing:** A modified algorithm with auto-hypertuned parameters was created using CV grid search method described above which employed smote sampling to achieve class balance.

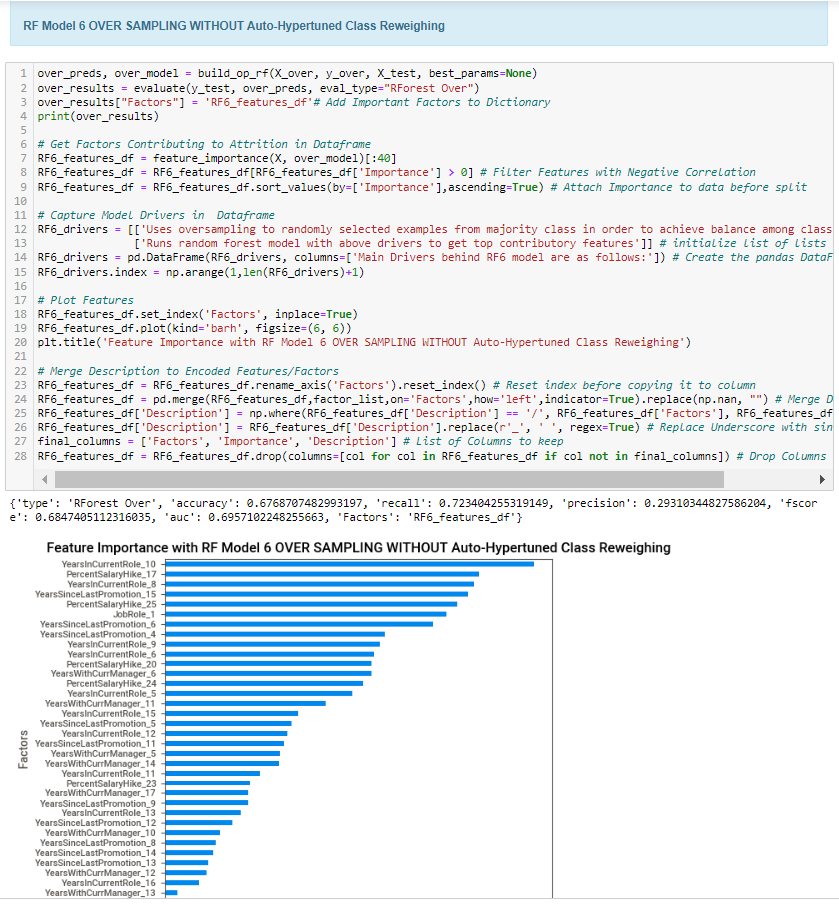


**2e) RF Model 5 UNDER SAMPLING WITH Auto-Hypertuned Class Reweighing:**

A modified algorithm with auto-hypertuned parameters was created using CV grid search method described above which employed random under sampling to achieve class balance.

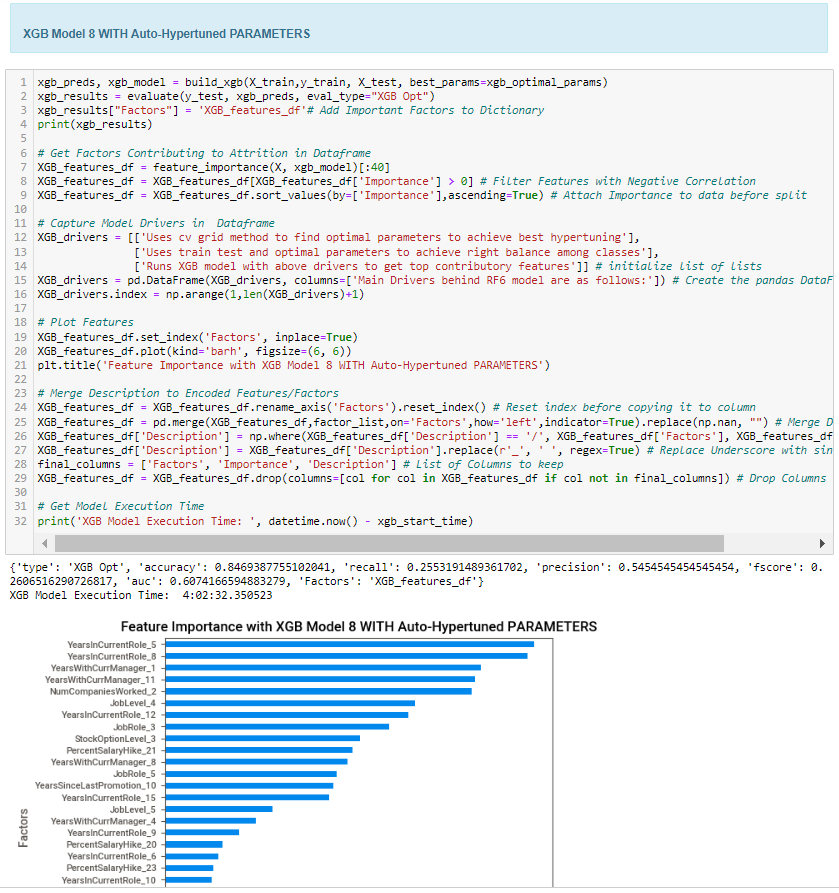


**2f) RF Model 6 OVER SAMPLING WITHOUT Auto-Hypertuned Class Reweighing:** A modified algorithm was created without achieving any class balance or hyperparameter tuning which employed random over sampling to achieve class balance.



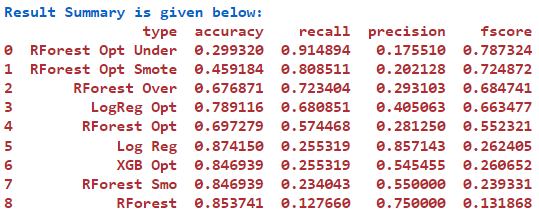
### **3) XGB Model**

A single method was employed to run XGB models with and without optimal parameters and class weighing:

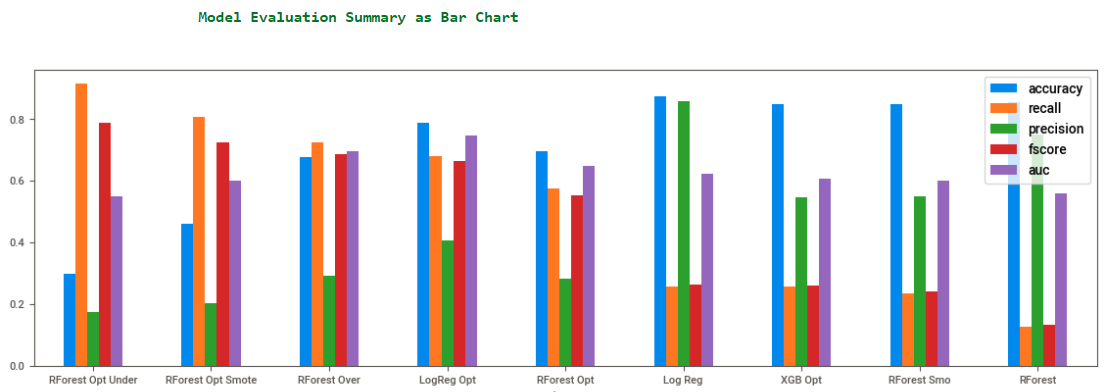


# **5) Recommended Model**

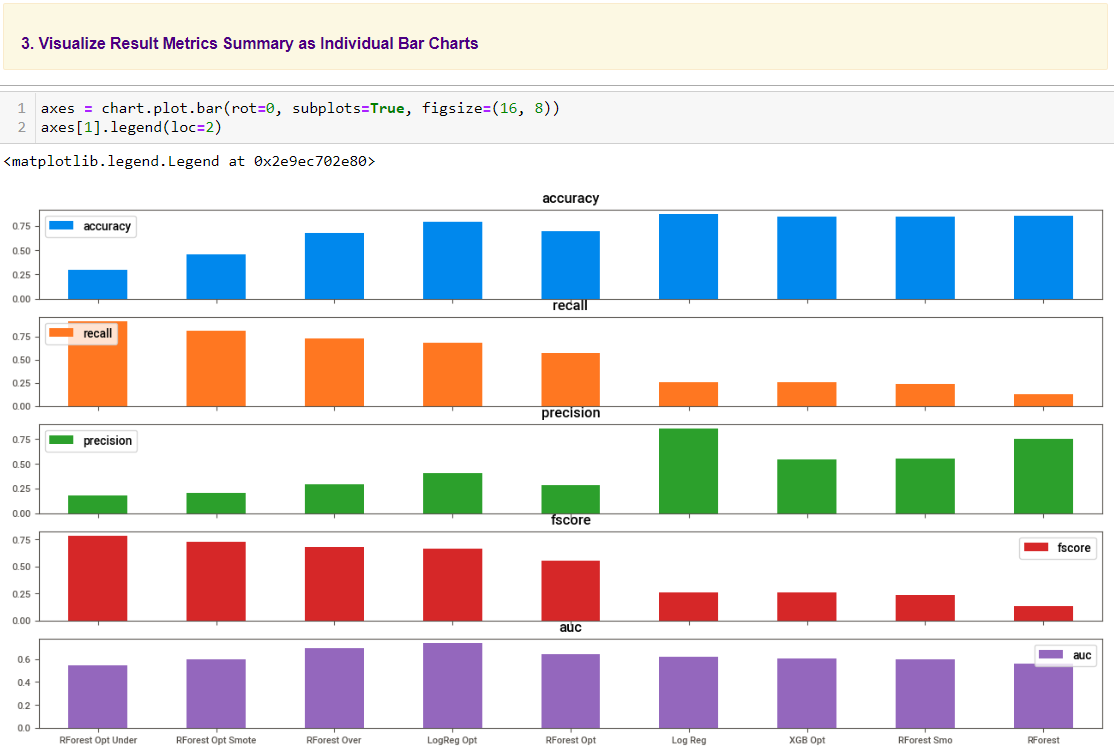
## **5a) Result Summary**



## **5b) Overall Visual Summary**



## **5c) Individual Model Visual Summary**



## **5d) Model Choice and Justification**

For many machine learning tasks with imbalanced datasets, like Employee Attrition, we normally care more about Recall than precision. As a baseline, we want the model to be able to find all possible factors and so, we would allow the model to make false-positive errors because the cost of false positives is usually not very high (maybe it will just cost a false notification email or phone call to confirm with employee).

On the other hand, failing to recognize positive examples (such as employee wanting to leave) can be too costly for the organization. As such, our first priority is to improve model's recall; then we will also want to keep precision as high as possible.

**In this case, the Model with Best Recall and F-Score is RForest Opt Under.** Hence, we will select this model for Employee Attrition Prediction.

# **6) Summary Key Findings and Insights**

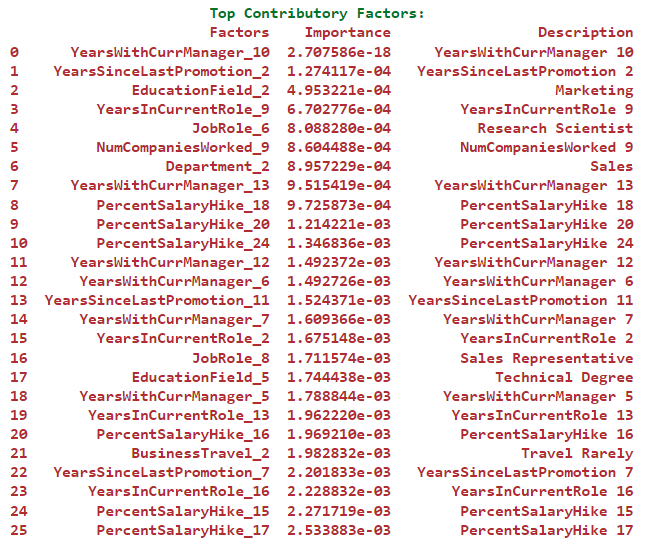
## **6a) Summarizing Model Drivers:**

Main Drivers behind top performing RF5 model are as follows:

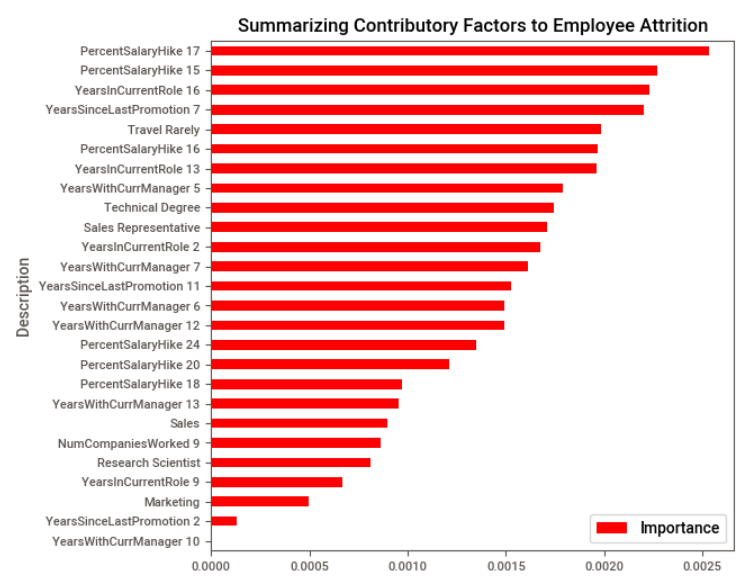
* Uses under-sampling to delete examples from majority class in order to achieve balance among classes
* Uses cv grid method to find optimal parameters to achieve best hyper-tuning
* Runs random forest model with above drivers to get top contributory features

## **6b) Enlisting Top Contributory Factors**

Top Factors Contributing to Employee Attrition in ascending order of importance are as follows:



## **6c) Visualizing Top Contributory Factors to Employee Attrition**



# **7) Link to Other Useful Models**

1. <https://github.com/IBM/employee-attrition-aif360/blob/master/notebooks/employee-attrition.ipynb>
2. <https://github.com/JNYH/employee_attrition/blob/master/employee_attrition.ipynb>
3. <https://github.com/elastic/examples/tree/master/Machine%20Learning/Analytics%20Jupyter%20Notebooks>
4. <https://github.com/ganesh10-india/HR_Analytics-Employee_Attrition-Classification-Models/blob/main/HR_Analytics_Employee_Attrition_Classification_Models.ipynb>

# **8) Github Link to Assignment Notebook**

<https://github.com/FATIMASP/IBM-MACHINE-LEARNING-CERTIFICATION/tree/main/Supervised%20Machine%20Learning:%20Classification>