

*Example of machine learning project in Data Science*

HR Analytics Project

Following is an end-to-end HR Analytics project which will guide all new comers who are aspirants of data science. So, please allow me to explain the agenda for this blog post. In this article, I have put down all the techniques in the form of sub-topics that I will be explaining one by one. And those pointers are as follows:  
  
1. Problem Definition

2. Data Analysis

3. EDA Concluding Remarks

4. Pre-processing Pipeline

5. Building Machine Learning Models

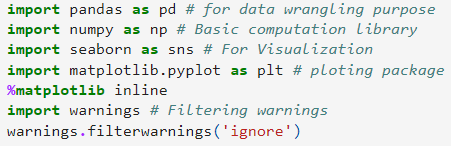
6. Concluding Remarks

Let’s start with the problem definition (short introduction) on the project:

1. **ProblemDefination**

Human Resource Analytics (HR Analytics) is an area in the field of analytics that refers to applying analytic processes to the human resource department of an organization in the hope of improving employee performance and therefore getting a better return on investment.  


Attrition in human resources refers to the gradual loss of employees’ overtime. Attrition affecting companies is a major problem since high employee attrition is its cost to an organization. Job postings, hiring processes, paperwork and new hire trainings are some of the common expenses of losing employees and replacing them. Additionally, regular employee turnover prohibits an organization from increasing its collective knowledge base and experience over time. Errors and issues are more likely if you constantly have new workers too.

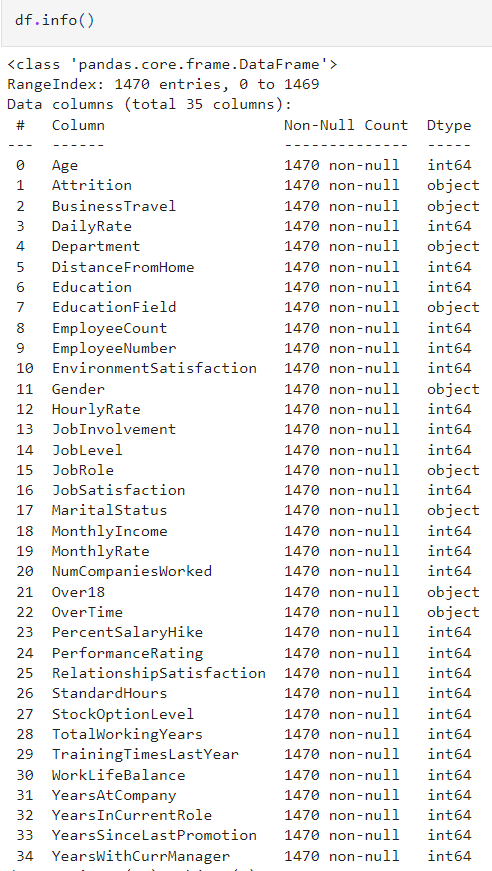
We need to first import all the required libraries:  


Use read\_csv method to imported entire dataset into jupyter notebook and stored into single variable name as df.



**2. Data Analysis -**

We will look at every columns of our data set and try to figure out which column is truly related to solve of problem statement or not.



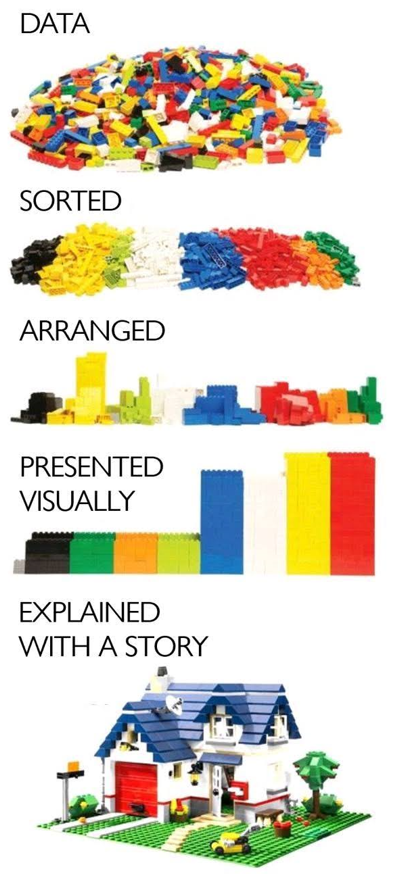
As we have dataset having 1470 rows and 35 columns below. In the 35 columns one is our target means Attrition and rest are features.

**3.EDA**

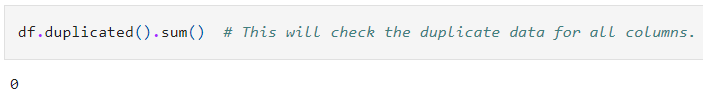
Exploratory Data Analysis is heart for building of machine learning model. EDA process is used to make our dataset into appropriate format, so we can achieve our real target. In this EDA we do entire dataset analysis via using various using tools and python libraries.

From the below picture, i can explain:

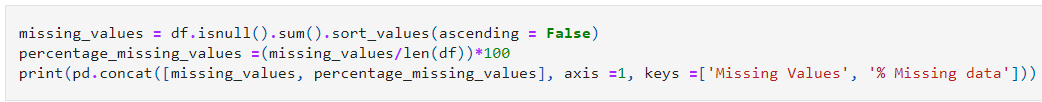
1. For any sort of Machine learning Model data is the most important thing that you must have. There are so many way to collect the data. I will not go into depth here but I can say primary and secondary there two are the main sources of data collection
2. Then I sorted the entire data as per their features and Arrange into some format
3. By the different means of visualization technique, I have visualized the data and find a way to explain the whole story.

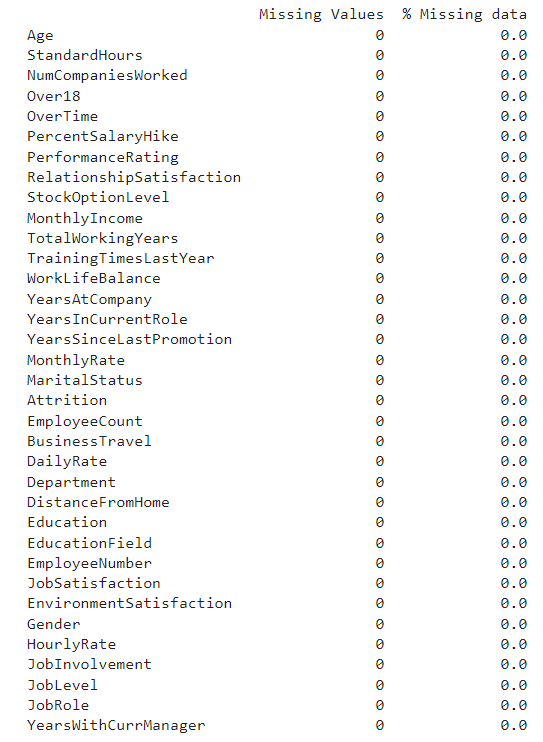


Since dataset is large, Let check for any entry which is repeated or duplicated in dataset



Missing value check



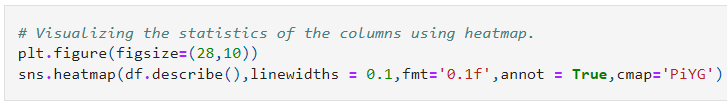


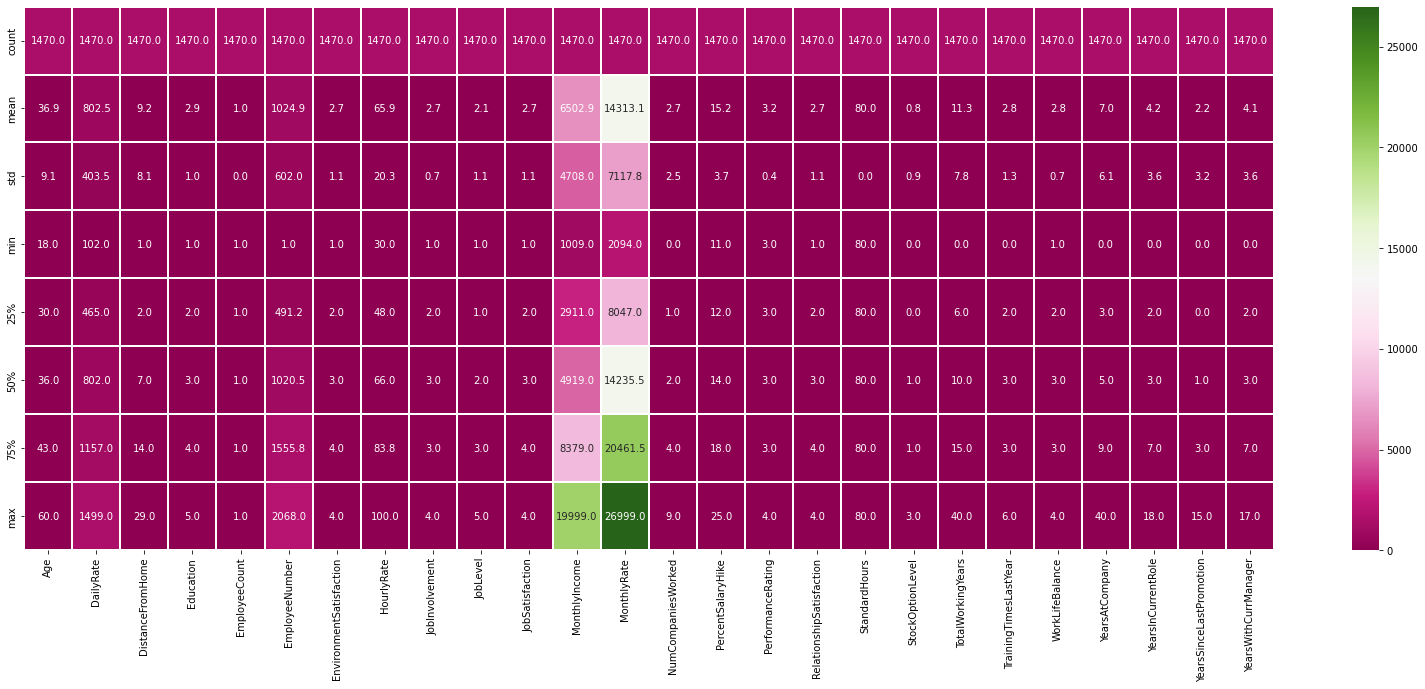
Luckily for us, there is no missing data! this will make it easier to work with the dataset.

Dataset doesnot contain Any duplicate entry, Missing Values.

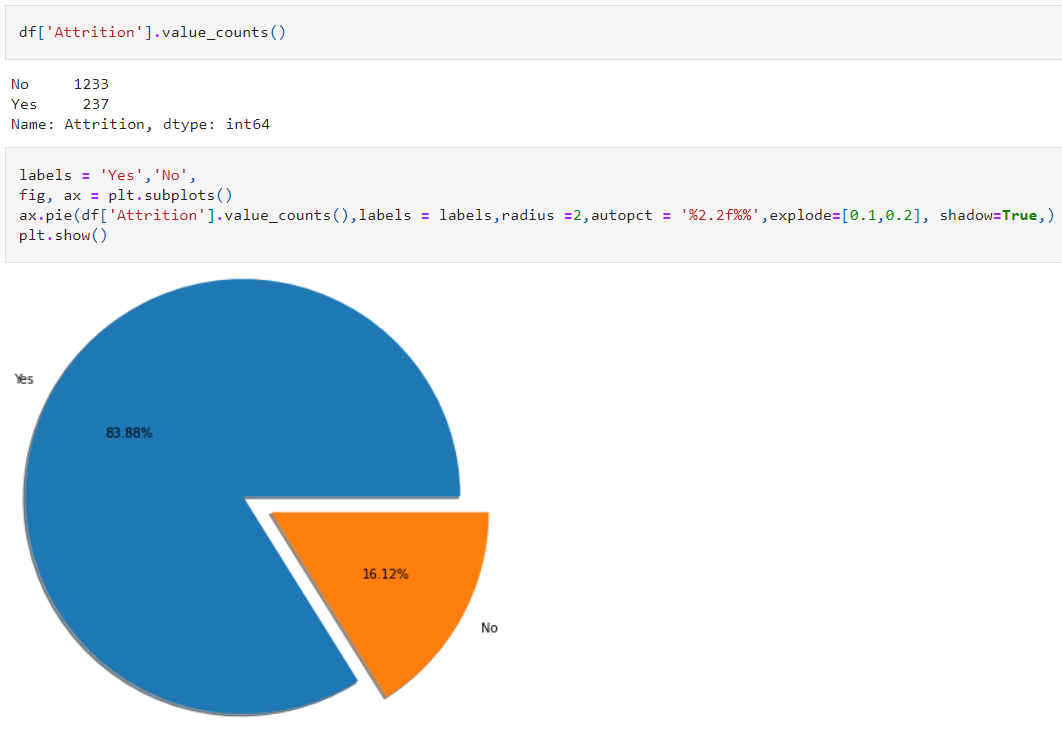
So, Yes To Go !!!

**Statistical Matrix**



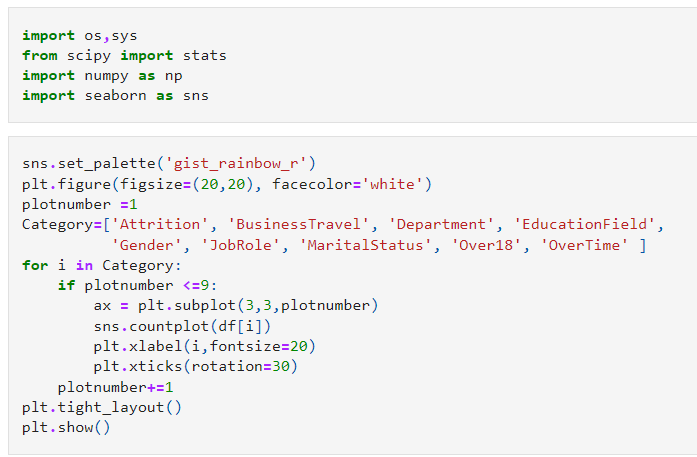


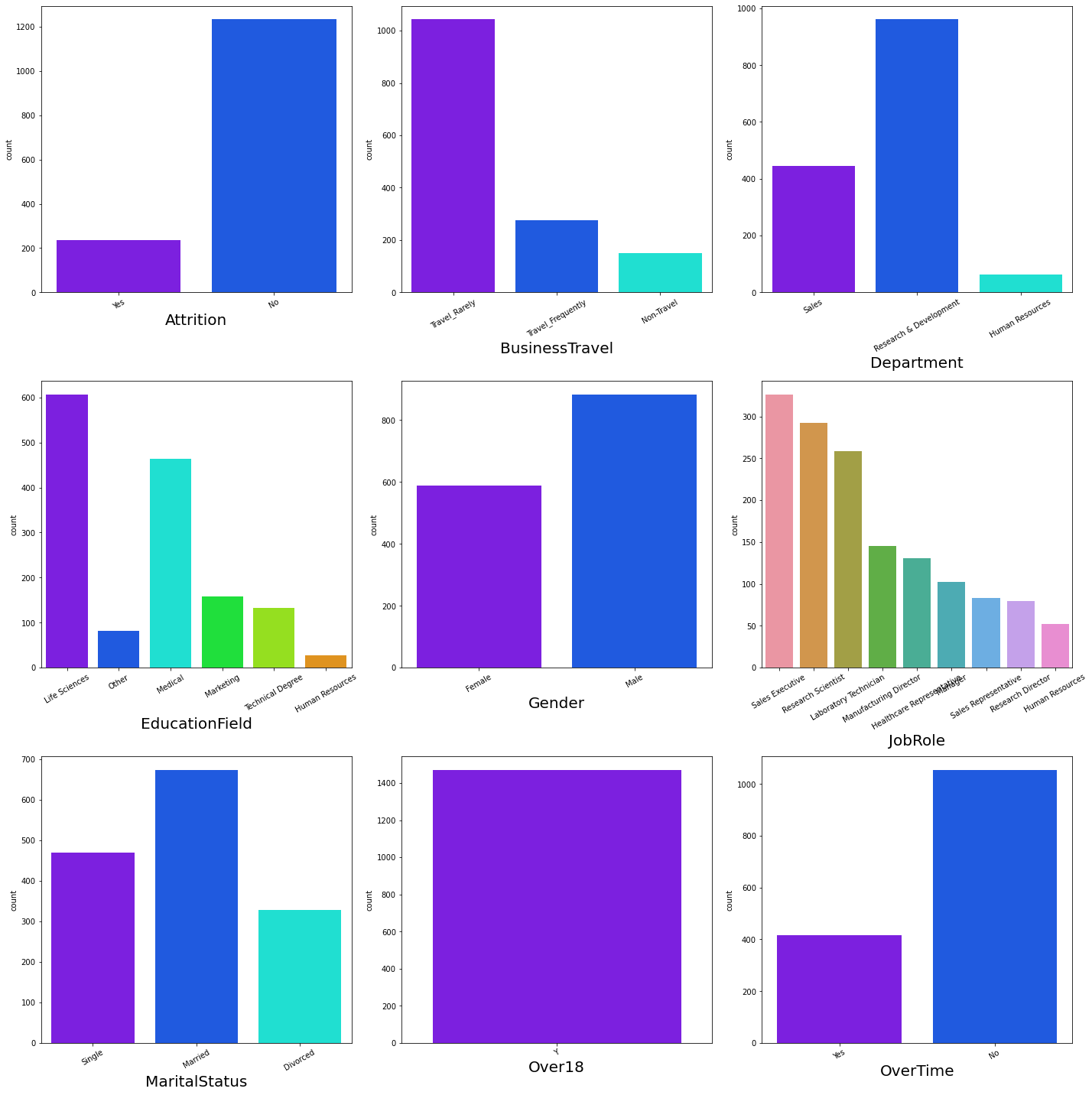
**Lets do some Statistical Analysis. Start with target Variable.**

****

83.88% (1237 employees) Employees did not leave the organization while 16.12% (237 employees) did leave the organization making our dataset to be considered imbalanced since more people stay in the organization than they actually leave.

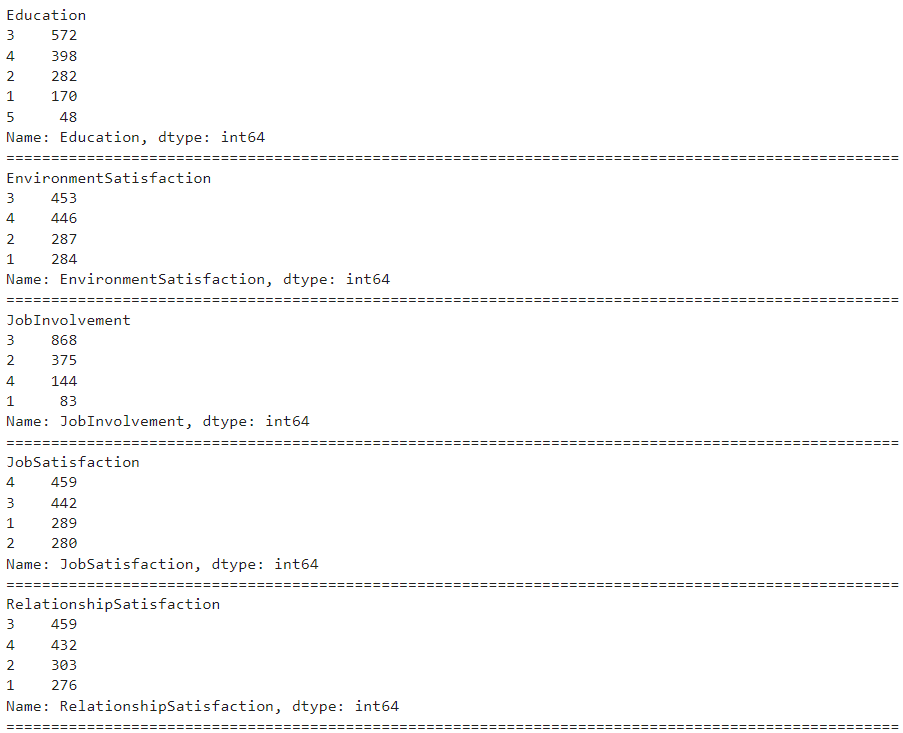
Before arrive at key questions which need to answer about HR Attrition, let try to gain some insight about individual features like distribution of different subcategories, different insight about Human Resource in company like education,job level, working domain. Start with Enlisting Value counts & Sub-categories of different categorial features available

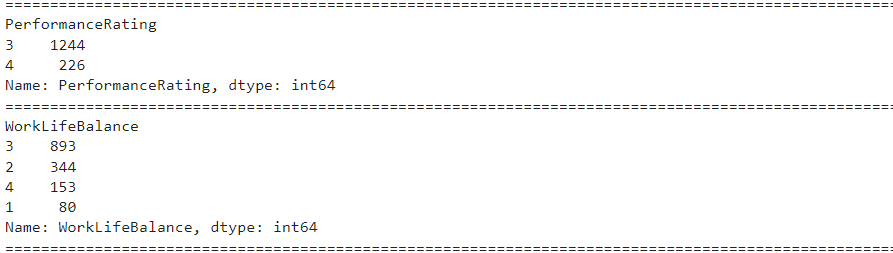
****

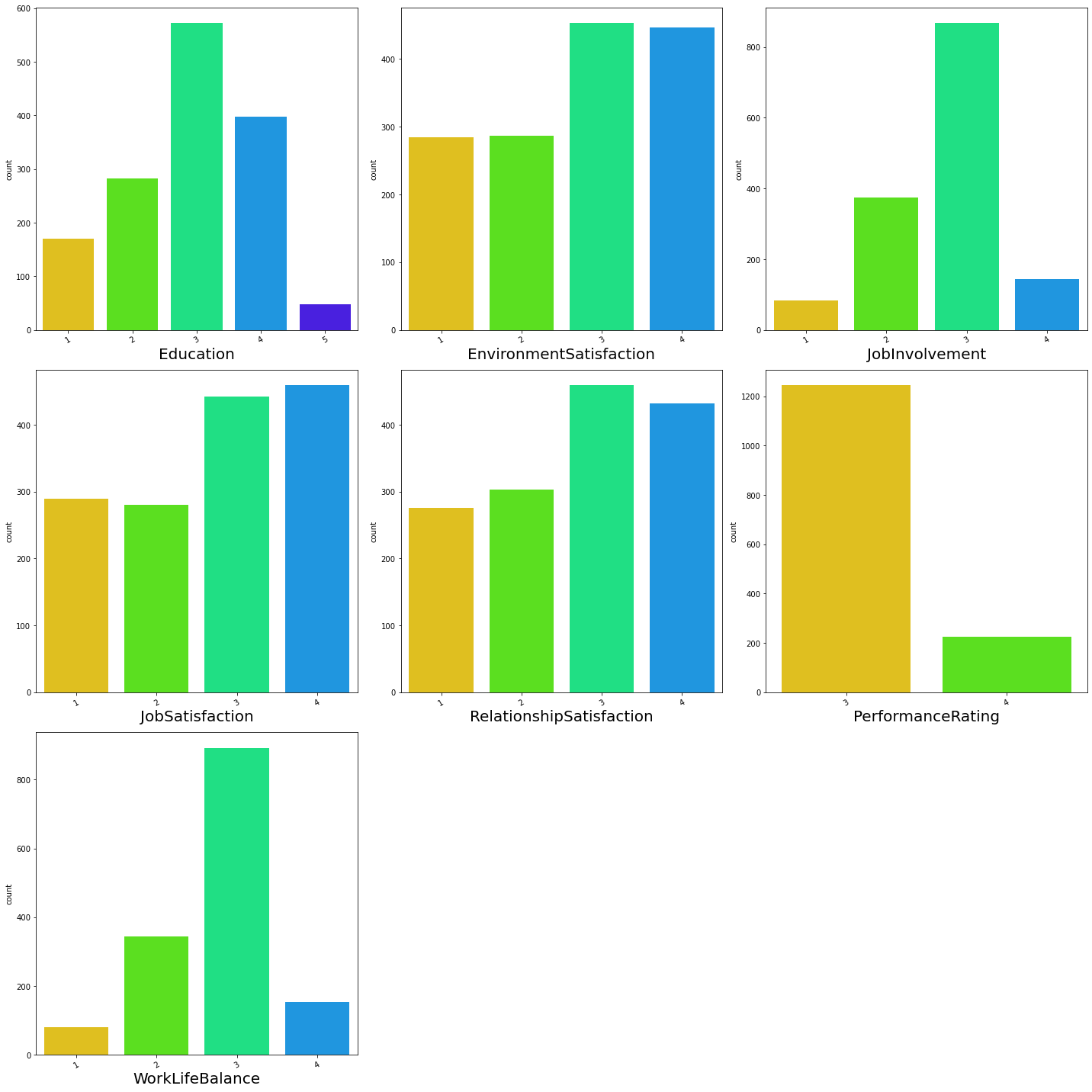


**Enlisting Value counts & Sub-categories of different Ordinal features available**

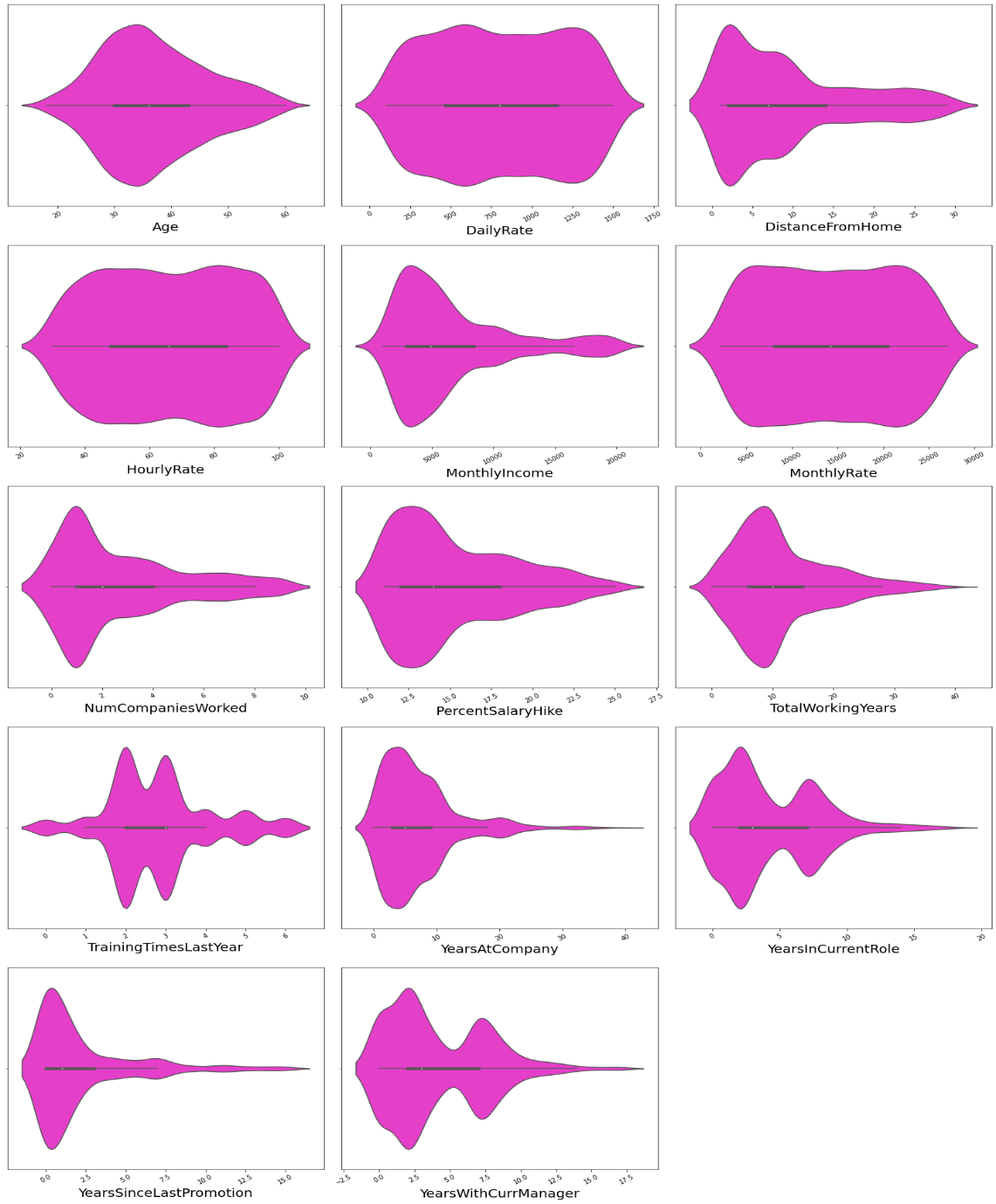
****

****

****

****

****

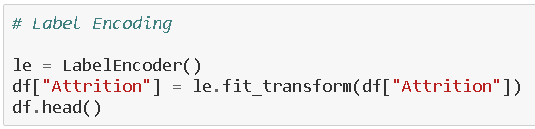


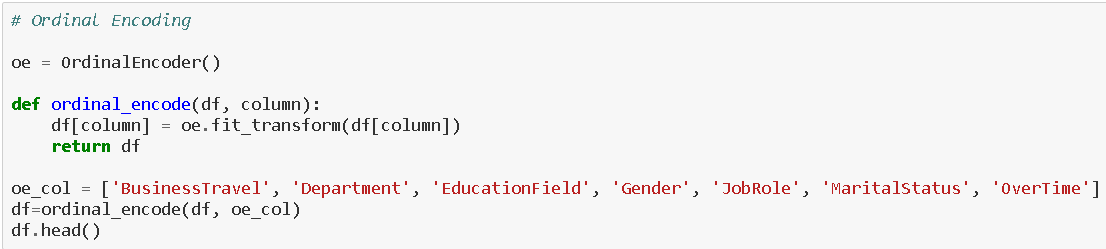
For Majority of people have spend 3 to 10 years at company. Most of people staying company upto 2 years after promotion. Majority of people are are train 2-3 times in last year.If employees leaves job then it loss investment for company. Majority of people stay in same role for maximum 4 yrs. Majority of Employees have salary hike of 10 to 15%.

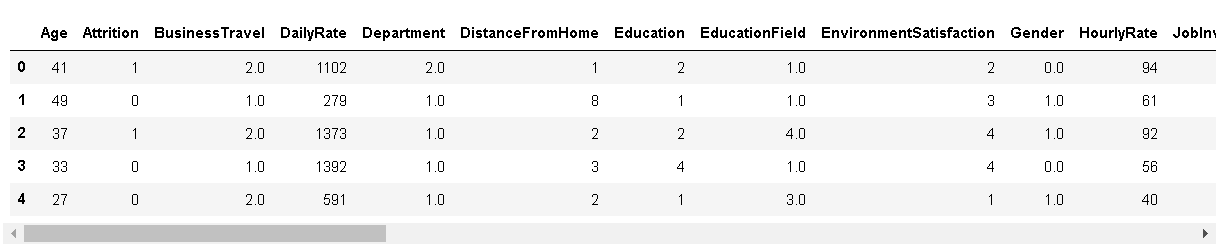
1. **Pre-Processing Pipeline**

Data processing is a process of preparing the raw data and making it suitable for it a machine learning model. It is the first and crucial step while creating a machine learning model.

When creating machine learning project, it is not always a case that become across the clean and formatted data and while doing any operation with data it is a mandatory to clean it and put in formatted way. So, for this we use data pre-processing task.

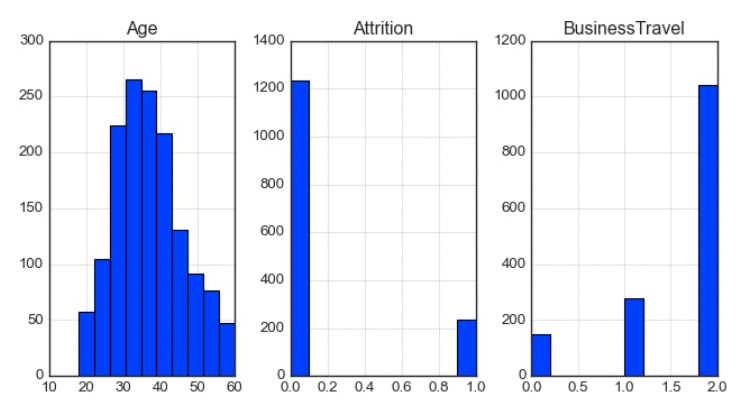






Here you can see all categorical data has been encoded.

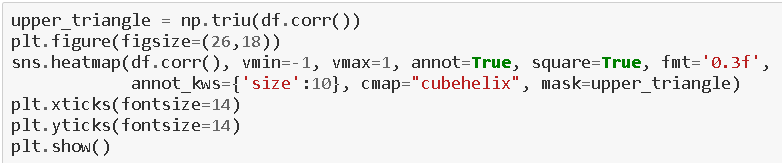


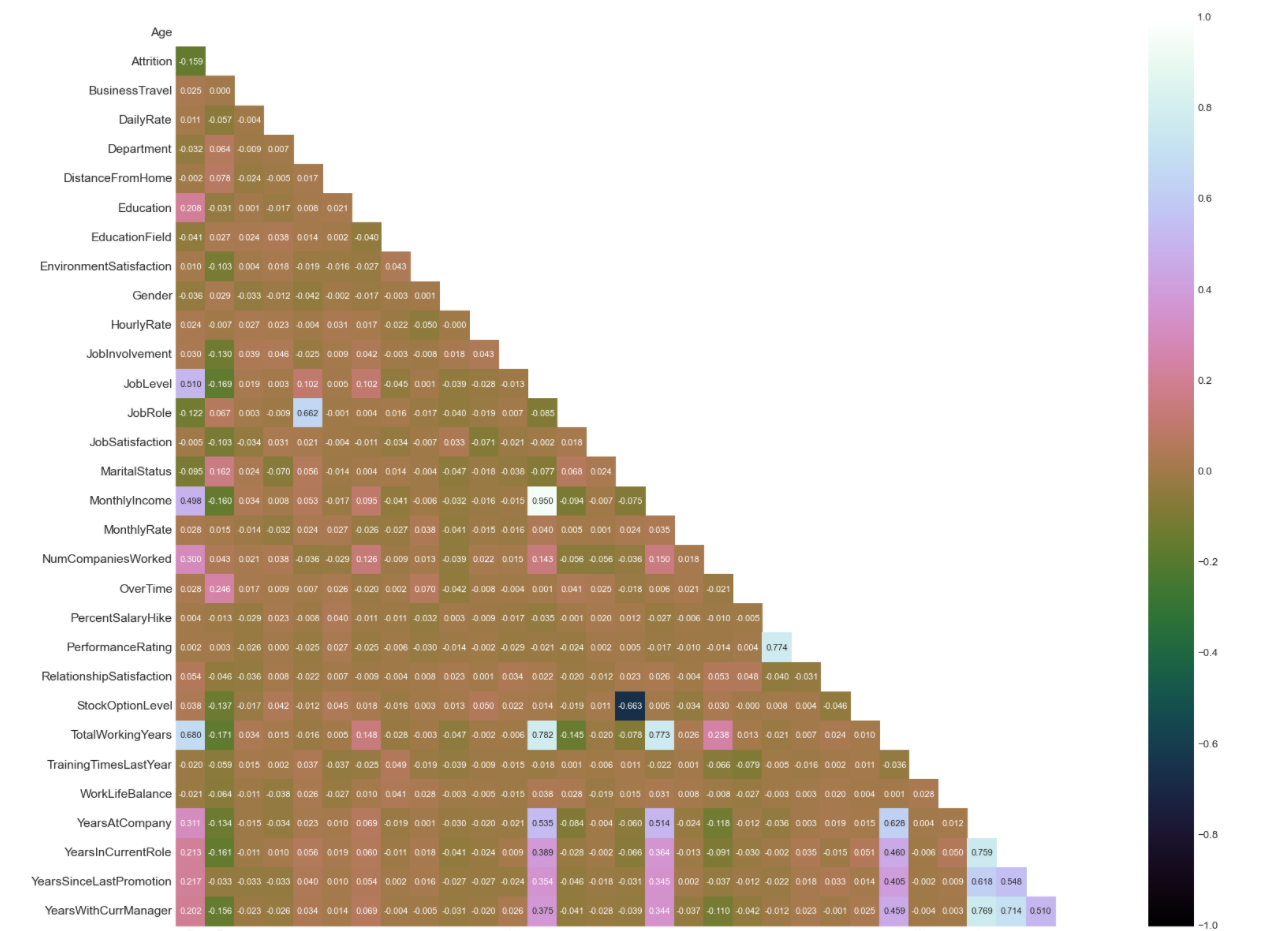


I have plotted histogram after using Encoder and it only give the distribution of numerical

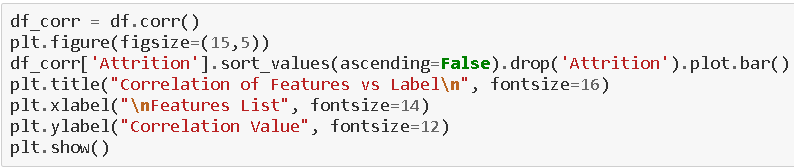
Columns present in our dataset.

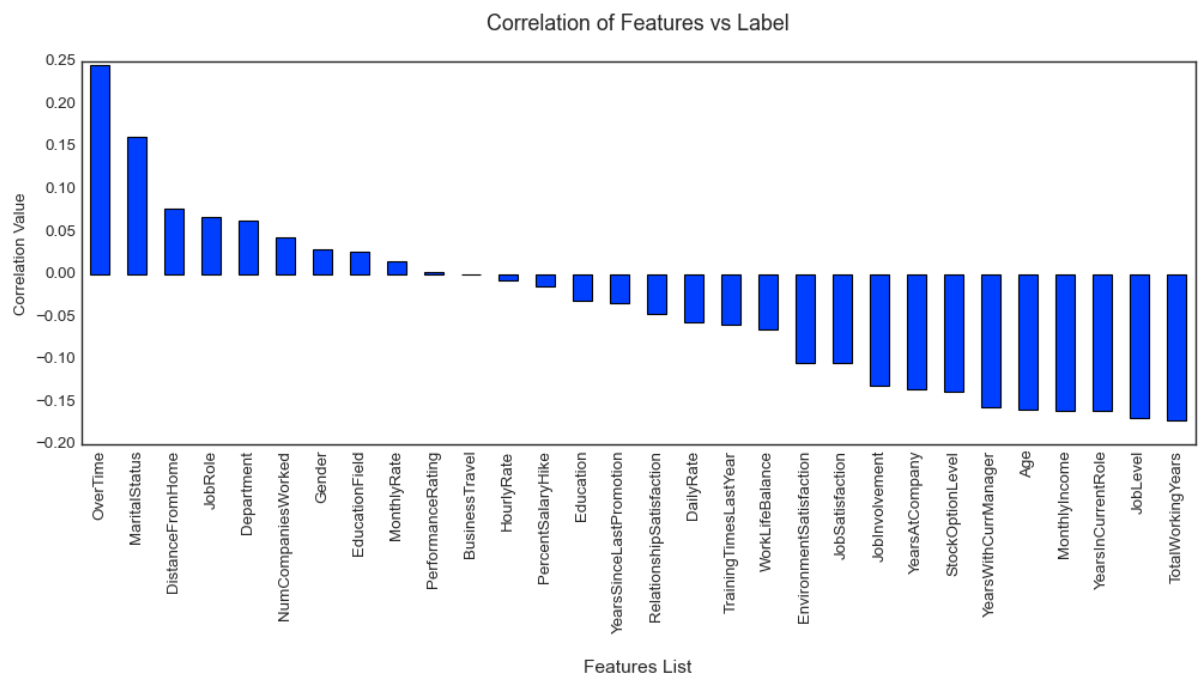
I now feel the need to check for correlation details in our dataset through a Heatmap. For those who still feel a confusion on correlation details let me break it down in two simple points that there are Positive correlation - A correlation of +1 indicates a perfect positive correlation, meaning that both variables move in the same direction together and Negative correlation - A correlation of –1 indicates a perfect negative correlation, meaning that as one variable goes up, the other goes down. The code to see this information is displayed below.



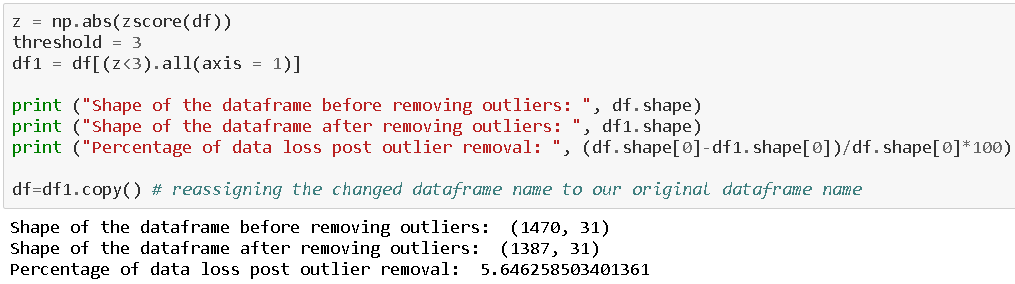


Obviously, because high number of columns present in our dataset I an not able to look at the numeric value of graph generated by the help of sns.heatmap. We can only have some information of multicollearnity by colour .

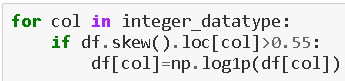




For the above bar graph at least we are to find which of our columns of our dataset either negatively or positively correlates to our label target and also have some numeric information too.



From the above z score function we have removed outliers present in our dataset .Since outlier was also part of our dataset. So, in this way we have lost our 5% data of our dataset. But no problem we can afford max 10% loss ourdatset if dataset is large.



As we it is very clear to all of us that our skewness acceptable range is +/-.5 , the skewness lying between this range is acceptable otherwise we need to treat As we have treated above.



We had to split our dataset columns into two parts namely X and Y. X variable containing all the feature columns and Y variable containing our label/target variable.

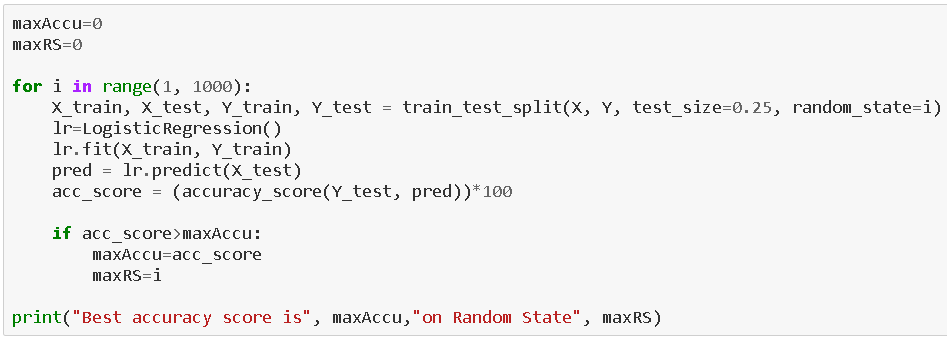
Also we have to take care of target imbalance issue.



By the help of this Smote method we have resolved the issue of imbalance .So, that our label doesn’t get biased.



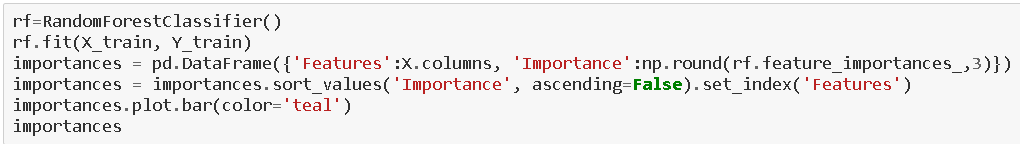
From the abovestandardScaler method we can scale that all the columns of our dataset so our dataset refrain from biasing for any particular column.

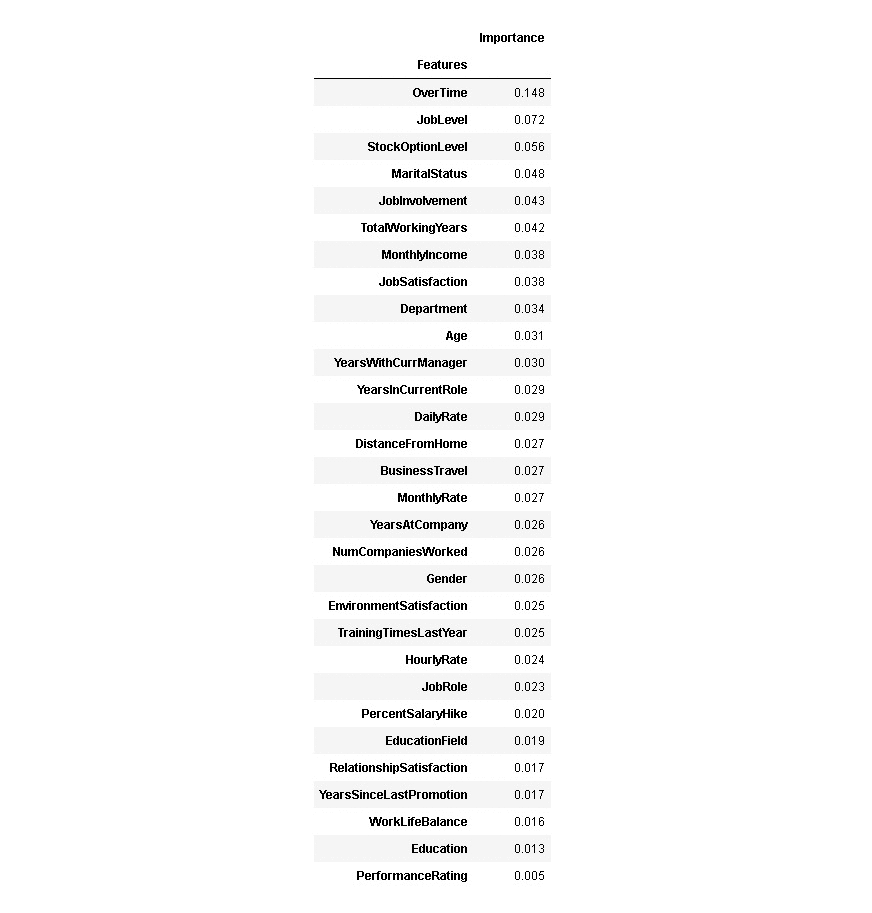


To find the best Random State for our Machine Learning Model .from the above I have sent 25% data for testing and 75% data for training. Actually, It’s up to you how precisely you want to make your model is.

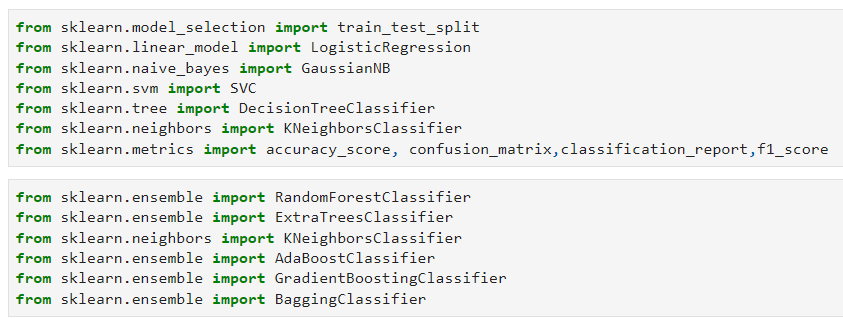


Before building the proper Machine learning model we have to decide how much data do you want to send as test and how much data do you want for training .It totally up to you.

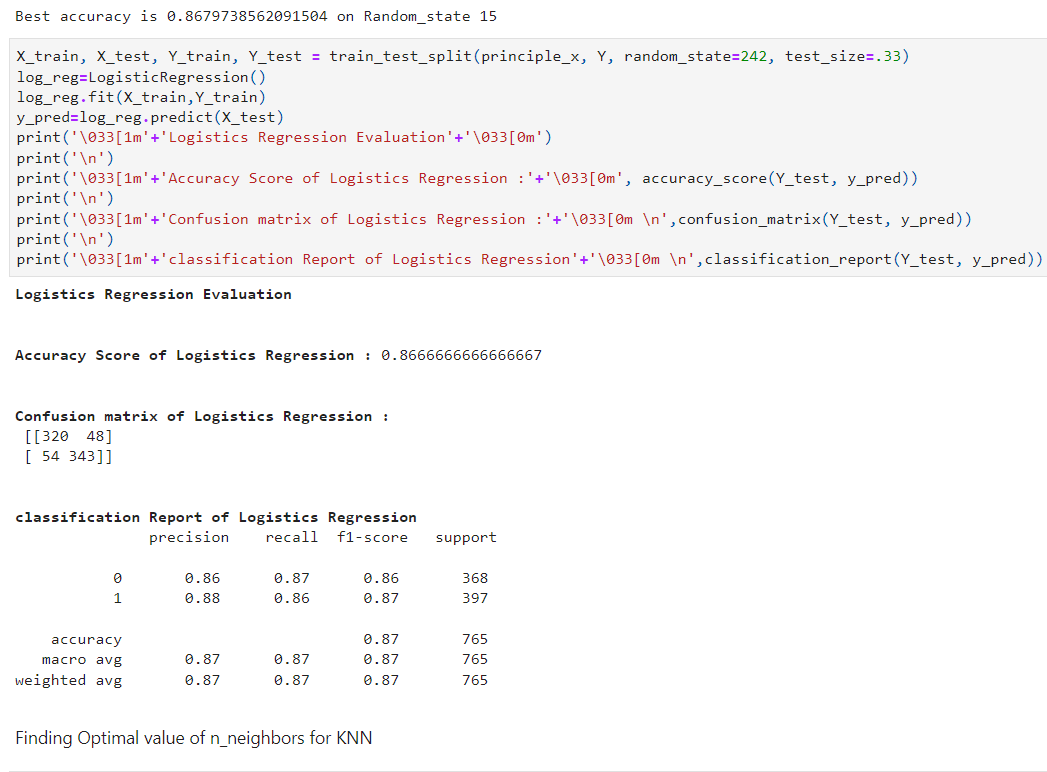


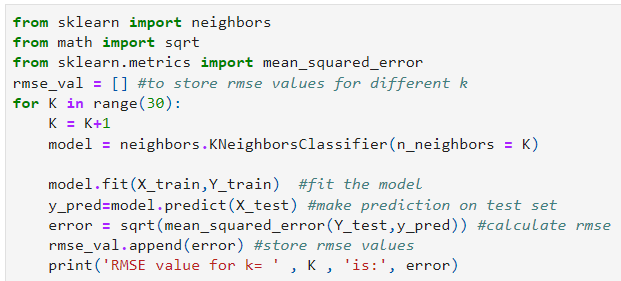


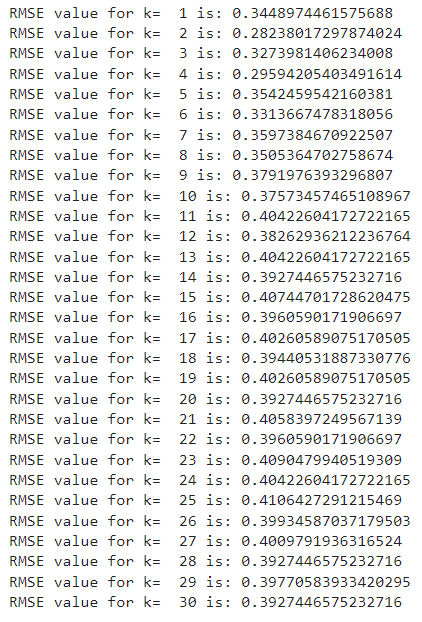
1. **Building Machine Learning Models**  
     
   In order to build a classification method I have imported the necessary libraries and created a function that contains our entire machine learning model creation and its evaluation metrics steps. This makes our job easier since later on we just need to feed the model’s name and get the result without repeating/rewriting the same code again and again.

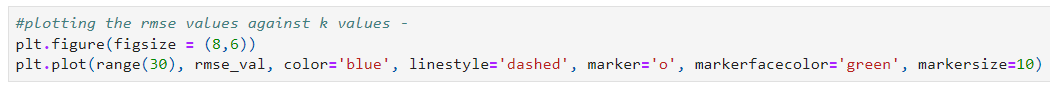


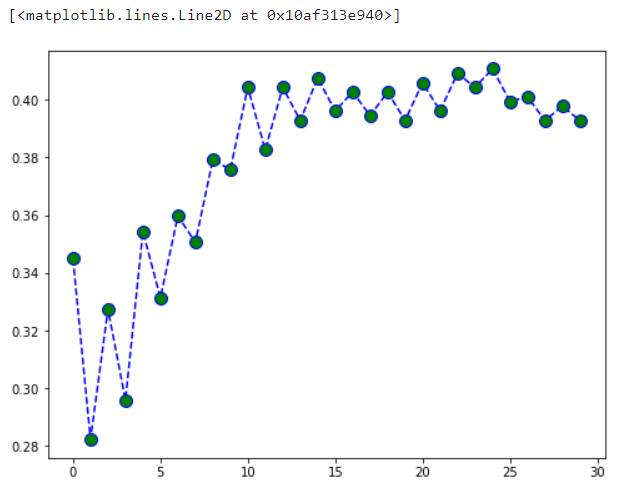












At k= 2, we get the minimum RMSE value which approximately 0.30032661958503204, and shoots up on further increasing the k value. We can safely say that k=2 will give us the best result in this case

**Applying other classification algorithm**



**Classification ML Algorithm Evaluation Matrix SVC() is**

**Accuracy Score :**

0.9150326797385621

**Confusion matrix :**

[[342 26]

[ 39 358]]

**Classification Report :**

precision recall f1-score support

0 0.90 0.93 0.91 368

1 0.93 0.90 0.92 397

accuracy 0.92 765

macro avg 0.91 0.92 0.91 765

weighted avg 0.92 0.92 0.92 765

============================================================================================================

**Classification ML Algorithm Evaluation Matrix GaussianNB() is**

**Accuracy Score :**

0.8483660130718954

**Confusion matrix :**

[[315 53]

[ 63 334]]

**Classification Report :**

precision recall f1-score support

0 0.83 0.86 0.84 368

1 0.86 0.84 0.85 397

accuracy 0.85 765

macro avg 0.85 0.85 0.85 765

weighted avg 0.85 0.85 0.85 765

============================================================================================================

**Classification ML Algorithm Evaluation Matrix DecisionTreeClassifier() is**

**Accuracy Score :**

0.7869281045751634

**Confusion matrix :**

[[285 83]

[ 80 317]]

**Classification Report :**

precision recall f1-score support

0 0.78 0.77 0.78 368

1 0.79 0.80 0.80 397

accuracy 0.79 765

macro avg 0.79 0.79 0.79 765

weighted avg 0.79 0.79 0.79 765

============================================================================================================

**Classification ML Algorithm Evaluation Matrix KNeighborsClassifier(n\_neighbors=22) is**

**Accuracy Score :**

0.8431372549019608

**Confusion matrix :**

[[274 94]

[ 26 371]]

**Classification Report :**

precision recall f1-score support

0 0.91 0.74 0.82 368

1 0.80 0.93 0.86 397

accuracy 0.84 765

macro avg 0.86 0.84 0.84 765

weighted avg 0.85 0.84 0.84 765

============================================================================================================

**Classification ML Algorithm Evaluation Matrix RandomForestClassifier() is**

**Accuracy Score :**

0.8993464052287582

**Confusion matrix :**

[[337 31]

[ 46 351]]

**Classification Report :**

precision recall f1-score support

0 0.88 0.92 0.90 368

1 0.92 0.88 0.90 397

accuracy 0.90 765

macro avg 0.90 0.90 0.90 765

weighted avg 0.90 0.90 0.90 765

============================================================================================================

**Classification ML Algorithm Evaluation Matrix AdaBoostClassifier() is**

**Accuracy Score :**

0.8496732026143791

**Confusion matrix :**

[[310 58]

[ 57 340]]

**Classification Report :**

precision recall f1-score support

0 0.84 0.84 0.84 368

1 0.85 0.86 0.86 397

accuracy 0.85 765

macro avg 0.85 0.85 0.85 765

weighted avg 0.85 0.85 0.85 765

============================================================================================================

**Classification ML Algorithm Evaluation Matrix GradientBoostingClassifier() is**

**Accuracy Score :**

0.8823529411764706

**Confusion matrix :**

[[323 45]

[ 45 352]]

**Classification Report :**

precision recall f1-score support

0 0.88 0.88 0.88 368

1 0.89 0.89 0.89 397

accuracy 0.88 765

macro avg 0.88 0.88 0.88 765

weighted avg 0.88 0.88 0.88 765

============================================================================================================

**Classification ML Algorithm Evaluation Matrix BaggingClassifier() is**

**Accuracy Score :**

0.8352941176470589

**Confusion matrix :**

[[318 50]

[ 76 321]]

**Classification Report :**

precision recall f1-score support

0 0.81 0.86 0.83 368

1 0.87 0.81 0.84 397

accuracy 0.84 765

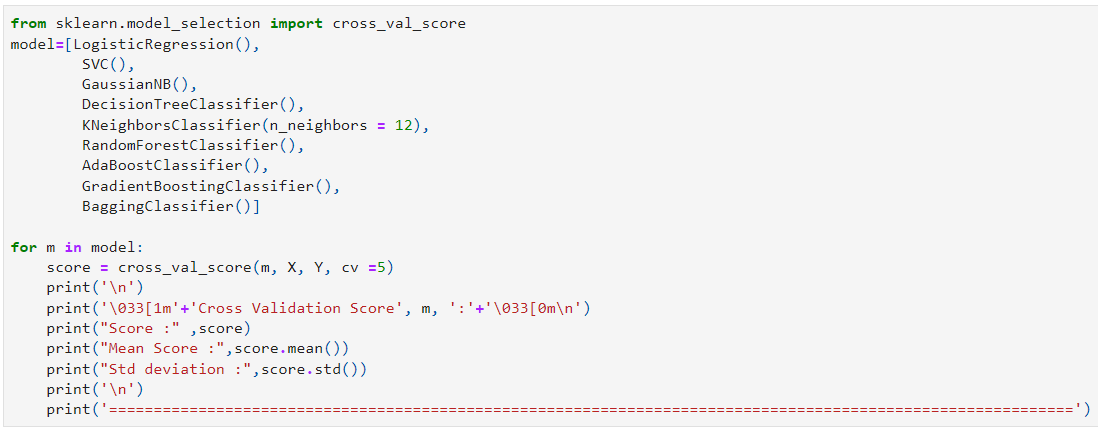
macro avg 0.84 0.84 0.84 765

weighted avg 0.84 0.84 0.84 765

============================================================================================================

**We can see that RandomForestClassifier() gives us good Accuracy and maximum f1 score. so we will continue further investigation with crossvalidation of above model**

**CrossValidation :**



**Cross Validation Score LogisticRegression() :**

Score : [0.62715517 0.69762419 0.73434125 0.6587473 0.7300216 ]

Mean Score : 0.6895779027332986

Std deviation : 0.04135532870381719

============================================================================================================

**Cross Validation Score SVC() :**

Score : [0.58189655 0.62634989 0.60259179 0.60043197 0.60691145]

Mean Score : 0.6036363297832724

Std deviation : 0.014216956502689568

============================================================================================================

**Cross Validation Score GaussianNB() :**

Score : [0.67241379 0.77537797 0.74514039 0.74946004 0.76457883]

Mean Score : 0.7413942057049229

Std deviation : 0.036138102161389025

============================================================================================================

**Cross Validation Score DecisionTreeClassifier() :**

Score : [0.63362069 0.91576674 0.9049676 0.86393089 0.89632829]

Mean Score : 0.8429228420347062

Std deviation : 0.10607229302878102

============================================================================================================

**Cross Validation Score KNeighborsClassifier(n\_neighbors=12) :**

Score : [0.70474138 0.71922246 0.72354212 0.73866091 0.71274298]

Mean Score : 0.7197819691666046

Std deviation : 0.011372183249743545

============================================================================================================

**Cross Validation Score RandomForestClassifier() :**

Score : [0.70474138 0.97408207 0.96976242 0.96544276 0.97840173]

Mean Score : 0.9184860728383107

Std deviation : 0.10695960881199645

============================================================================================================

**Cross Validation Score AdaBoostClassifier() :**

Score : [0.60775862 0.93736501 0.93736501 0.92656587 0.95464363]

Mean Score : 0.8727396291055335

Std deviation : 0.13279559072168926

============================================================================================================

**Cross Validation Score GradientBoostingClassifier() :**

Score : [0.58405172 0.97192225 0.96328294 0.95680346 0.97192225]

Mean Score : 0.8895965219334176

Std deviation : 0.1528784700258559

============================================================================================================

**Cross Validation Score BaggingClassifier() :**

Score : [0.67025862 0.94816415 0.95464363 0.93304536 0.96976242]

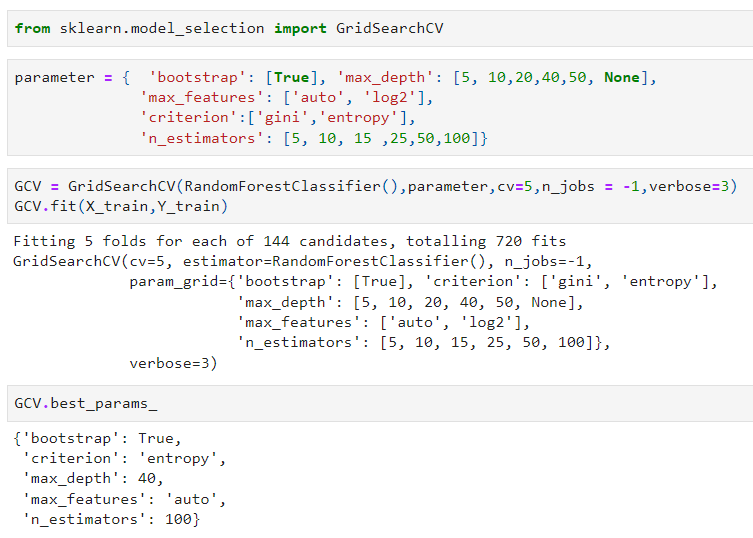
Mean Score : 0.8951748342891189

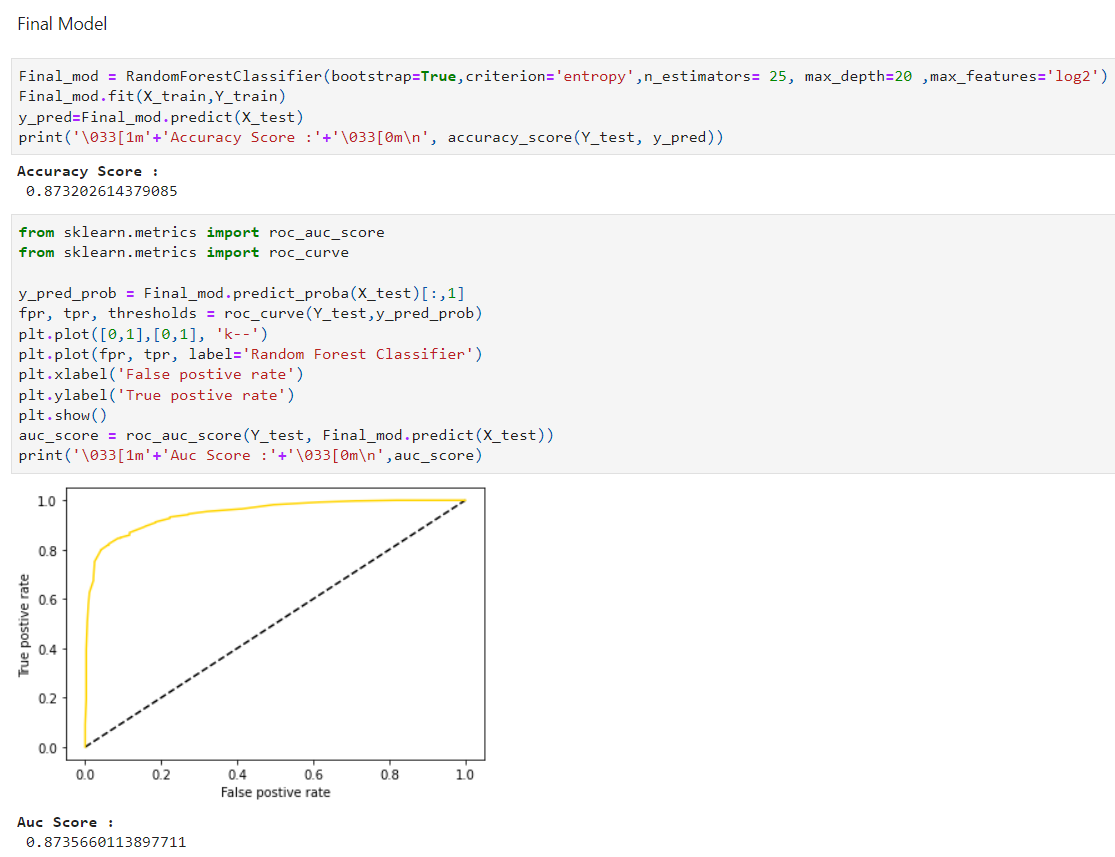
Std deviation : 0.11307448174574268

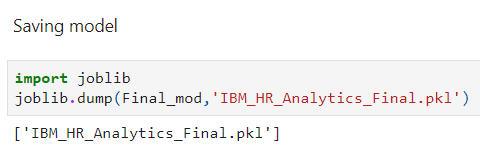
============================================================================================================

**On basis of maximum score in crossvalidation of Random Forest Classifier. we will apply Hyperparameter tuning on Random Forest model**

**Hyper Parameter Tuning : GridSearchCV**







1. **Concluding Remarks**

Let me go back to each step I have taken in this project. from beginning we have

Understanding the problem definition to go through EDA Processes. We went to some certain pre-processing steps and finally build Machine learning Model.

For any of machine learning project my suggestion is first you have to understand the problem on ground level .if you don’t allow yourself to work with diligence .if you don’ t work harder anything that you are doing or will do , not only in case of machine learning but also in life cycle would be futile. Maybe, my endeavour assist you whenever you will get stuck

**Disclaimer**

I am new comer here in data science domain with some knowledge 6-month Data Science Course from Data Trained Institute. I have shared my effort in this blog to someone who is stepping in this field and can take some advantage from it. but to be honest it’s definitely inspired by others I saw many blog of this project on internet who have worked on this projects before me I just went through each of project and concluded my best way to make this blog.

**THANK YOU**

**Submitted By – AKSHAY PAWAR**