Blog on HR Analytics Project-

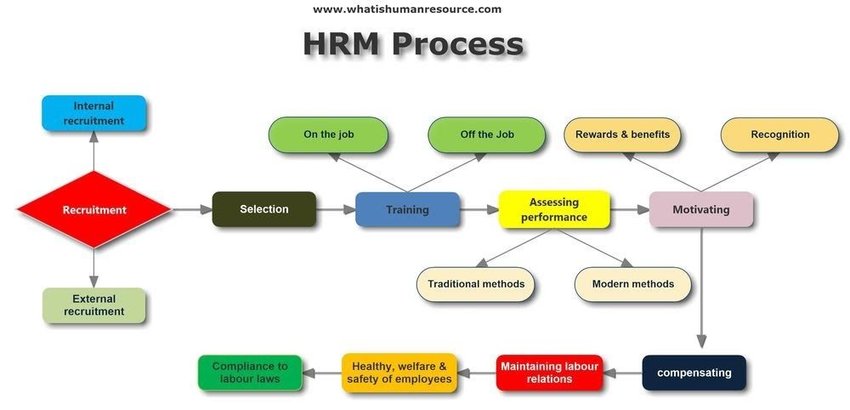
Understanding the Attrition in HR

**Background :**

Human resource is an integral part of any organization. It also considered as one of the pillar for the organization. It consist the practice of recruiting, hiring, deploying and managing an organization’s employees. Every organization has the separate department carrying out these activities known as Human Resource Department. This department carries all the activities regarding the human resource and its management.

This department is solely responsible for making the raw employees the best and most valuable asset for any organization, and hence sometimes we call them Human capital as well. Human Resource has so much importance in the organization that we can see its reference in the mission and vision statements, goals and long term plannings of the organizations. HR department also has the major responsibility of maximizing the return on investment(ROI) done behind the staffing and the related activities.

The Human resource Management is a step-by-step process, which every organization has to follow and each step has its own timing and expense attached with it.



**Problem Definition:**

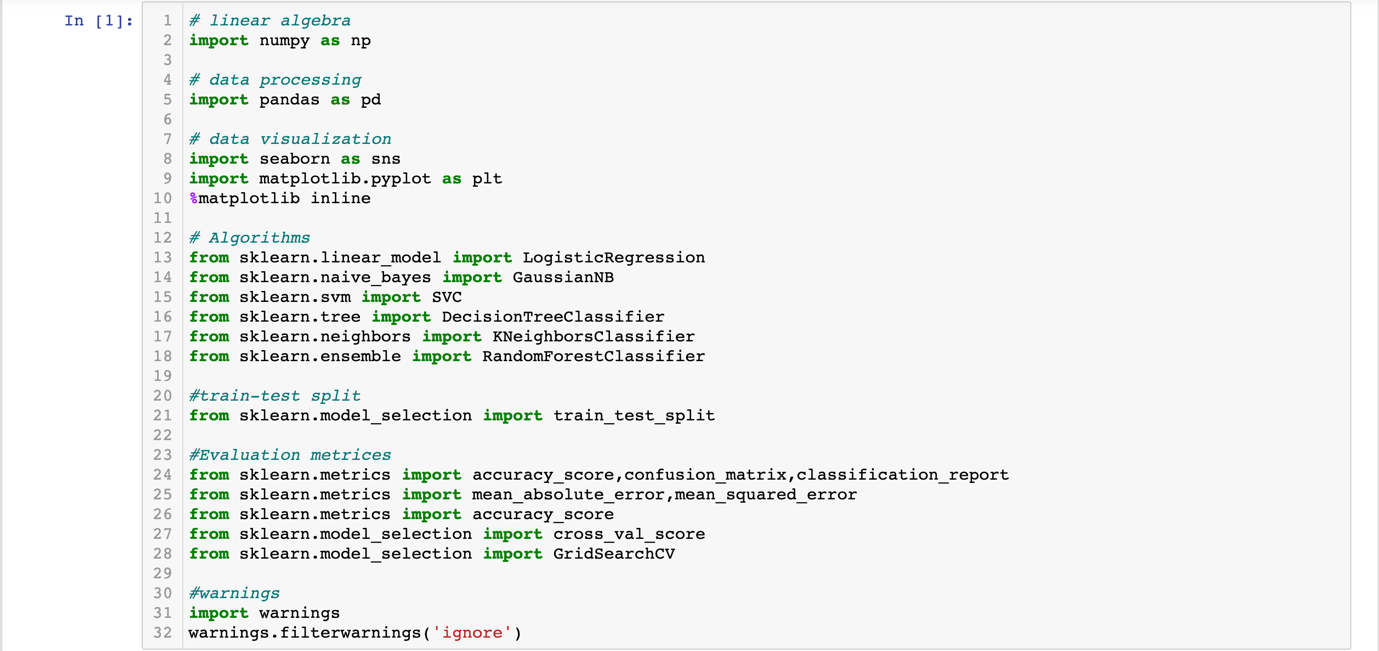
As we can see in the above diagram, HR department is responsible for carrying all the above activities in order to reduce the biggest problems which may occur anytime in future and one of such problem is **attrition of human resource**. It takes a lot of time and expense to make an individual from normal employee and an asset, but when such employee leaves the organizations, it cost the organization the whole time and efforts invested behind that particular employee.

The process of making an individual an employee takes very long time and effort, hence when any employee leaves, organization cannot fill his/her space immediately. Sometimes temporary replacement is possible, but however it’s not feasible option always. Because of this, HR department of any organization has a goal of lowest attrition rate ever.

Even though, HRD tries to take all the precautions to maintain the lowest attritions rates, there are some factors which may lead to the higher attrition rates. Some of these factors are controllable and some are not.

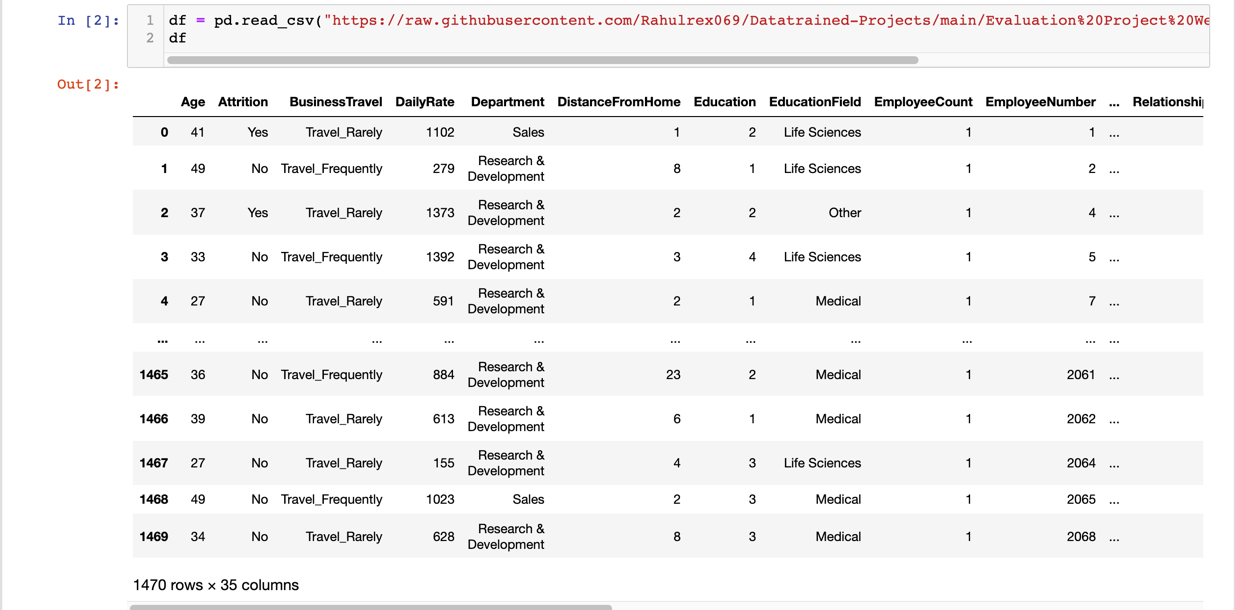
In this article I will go through the whole process of building machine learning model for predicting the whether an employee will leave the organization or not. For this, I have received a data consist of 1470 employees with the 34 feature columns and 1 target column called Attrition.

**Importing libraries:**

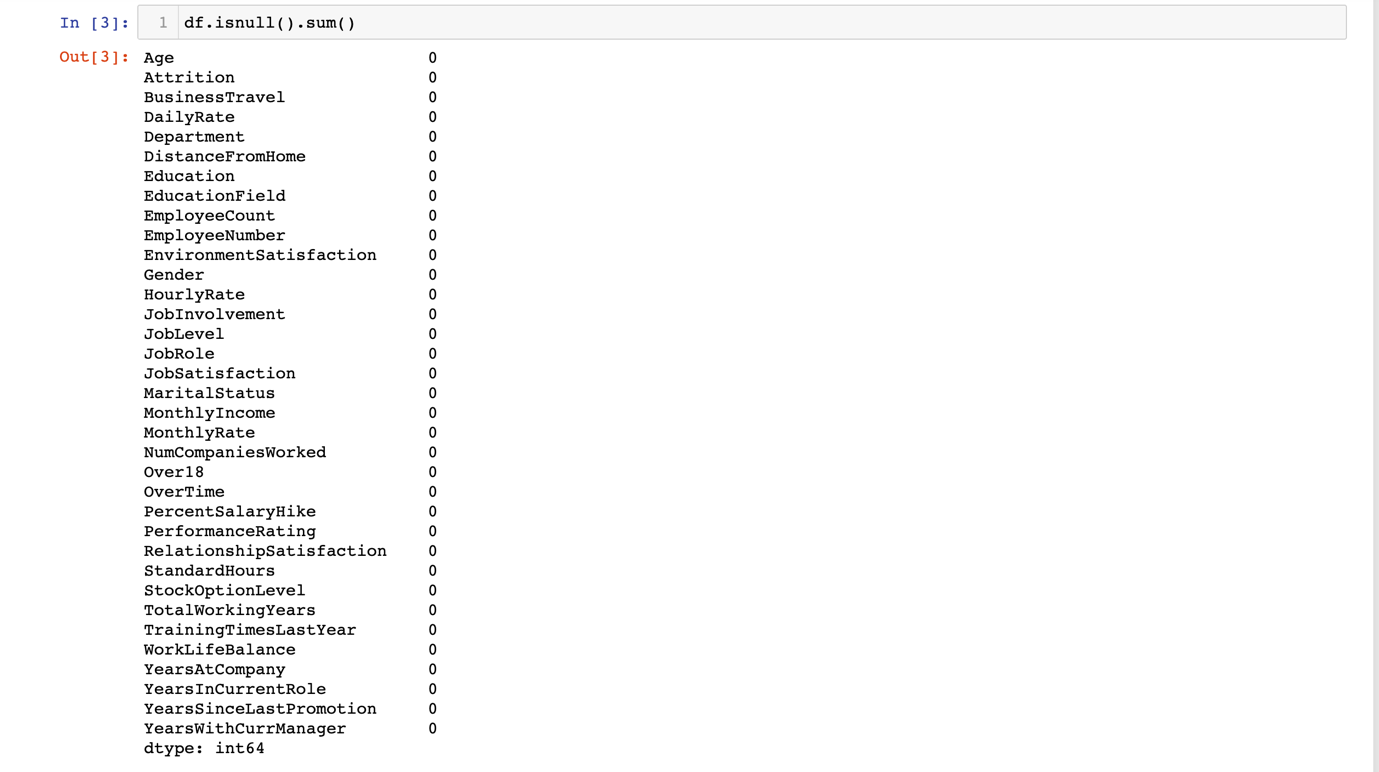


I have imported 5 different algorithms to find out which algorithm will fit the best to this data and go ahead for hyper tuning on it.

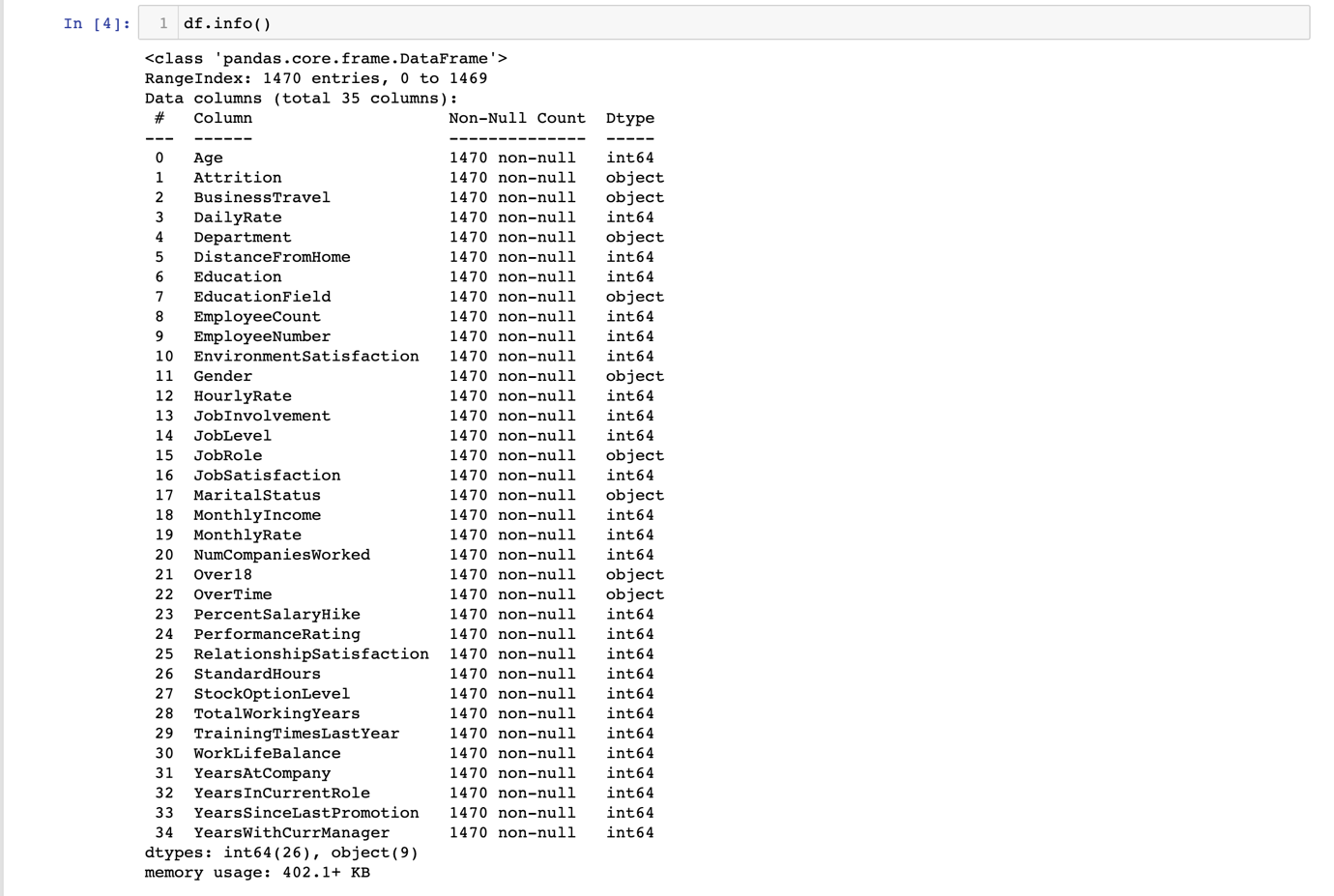
**Getting the Data:**



**Data Exploration/Analysis:**



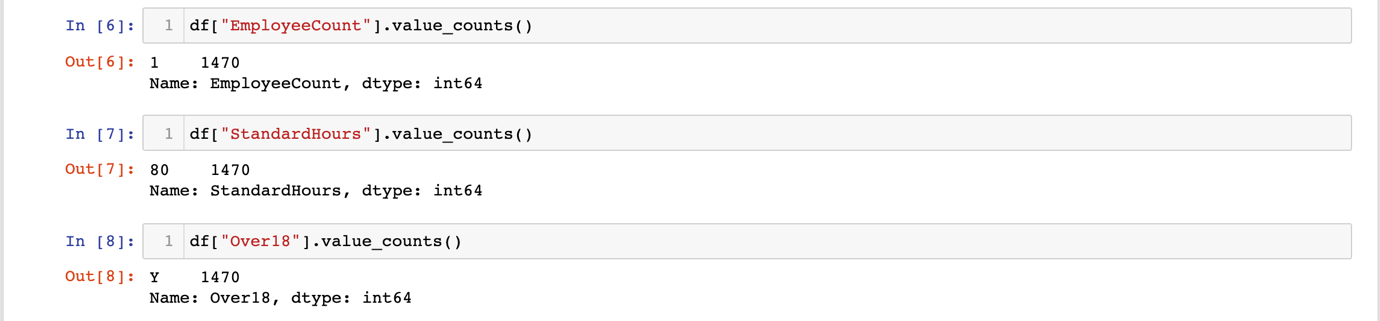
I used isnull() method to check for finding the missing values, but there are no missing values as per this method.



With info() method, I checked the datatypes. Data contains total 34 feature variables and one target variable i.e. Attrition. The total rows in the dataset are 1470 and has no missing value. Out of 34 features, 8 features are object / categorical datatype and 26 are integer datatype. Our target column is categorical in nature and contains binomial data including Yes and No as values, which we need to convert in integers later.

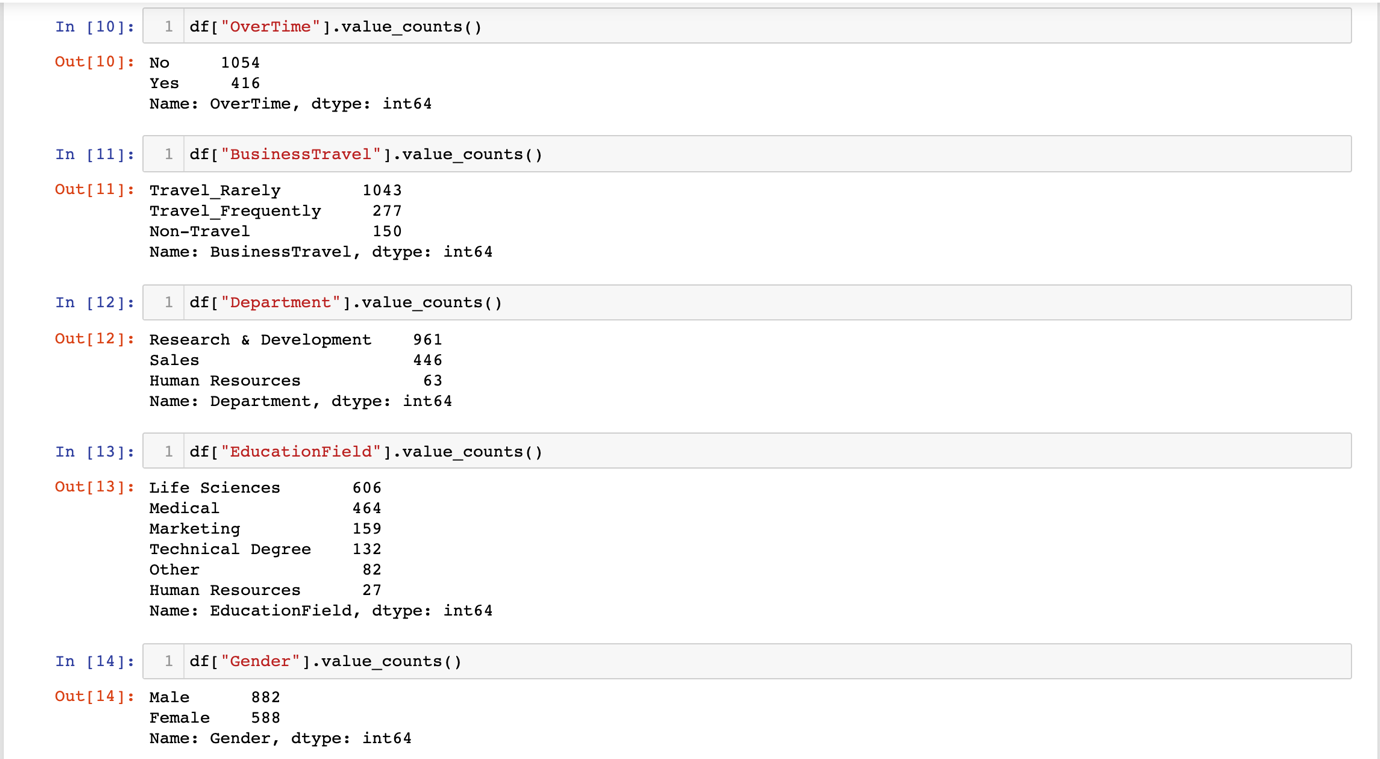


There only 237 cases where employee has left compared to 1233 who has not, creating a problem of class imbalance. Hence we need to balance the classes so that the model can learn better.



These three column has single value which not of any use in model. Hence I will drop it.

**



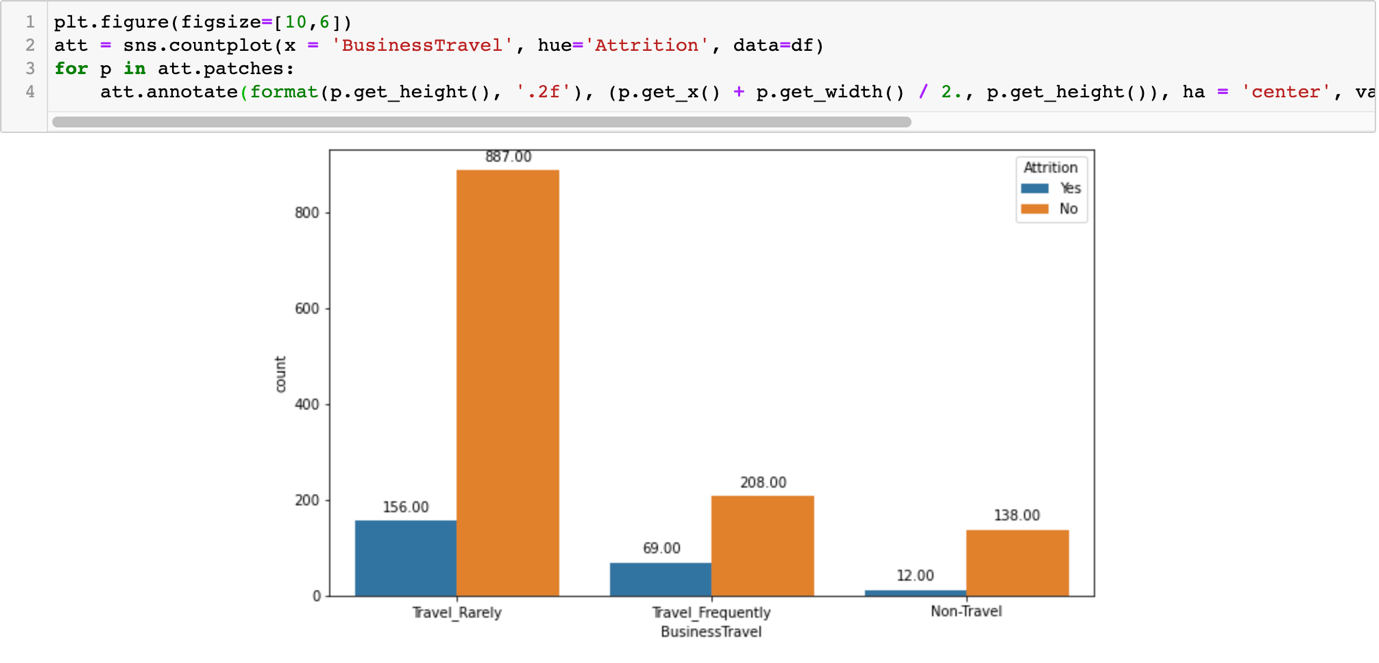


Rest of the categorical columns will be useful for the analysis, hence I will keep them as of now.

**Data Visualization:**

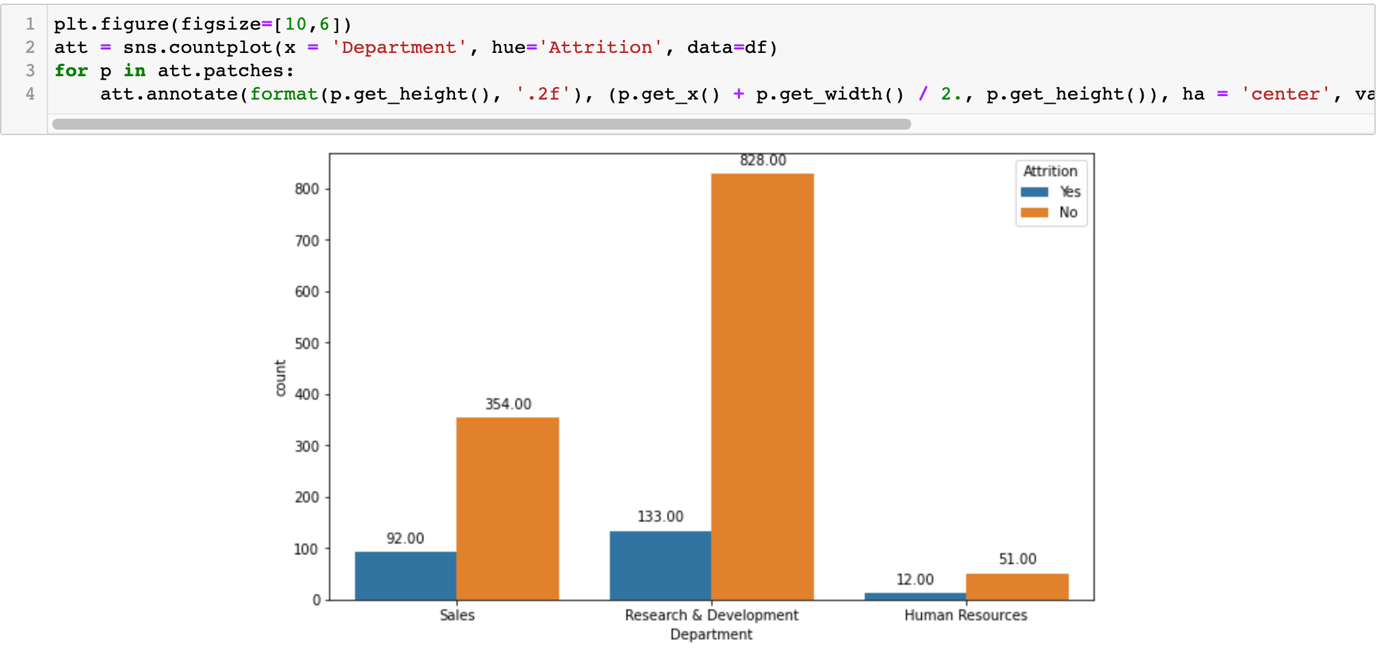
Now lets us check the relations in the features and target with the help of plots.

1. Business Travel and Attrition



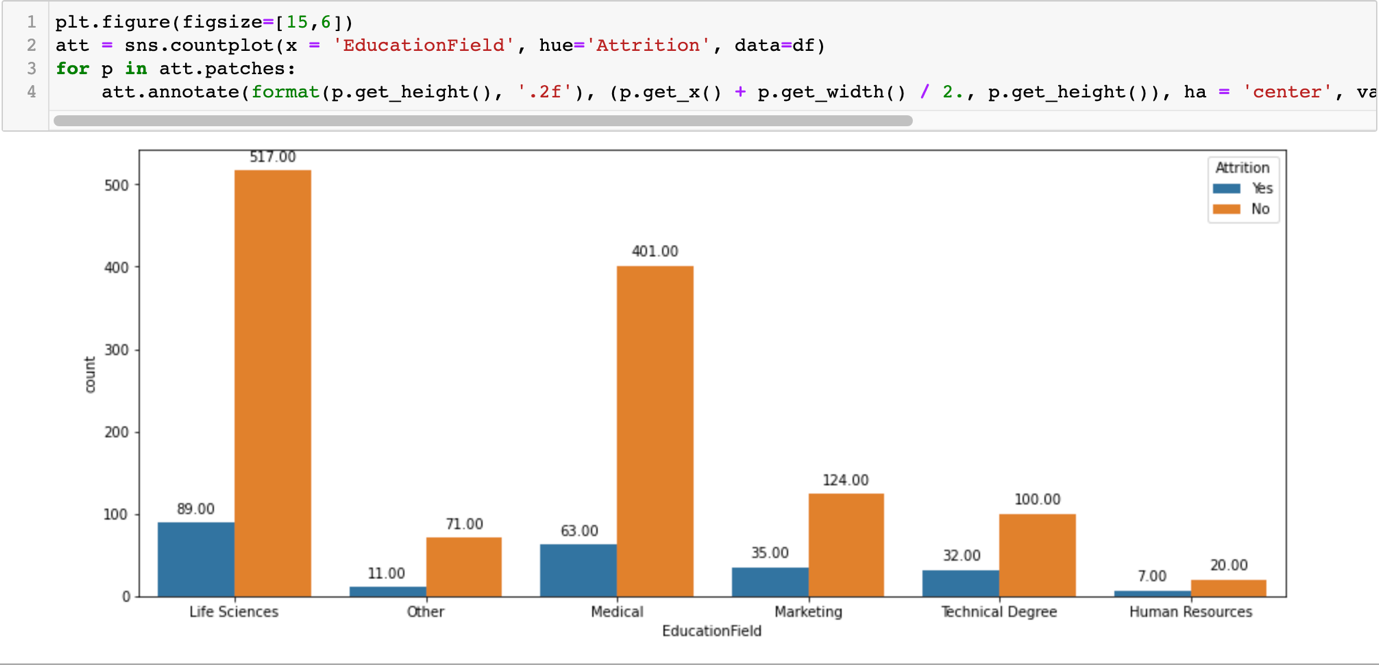
As per the plot, Non-travellers had only 8% of attrition rate which is lowest compared to those who travel rarely and travel frequently.

1. Department and Attrition

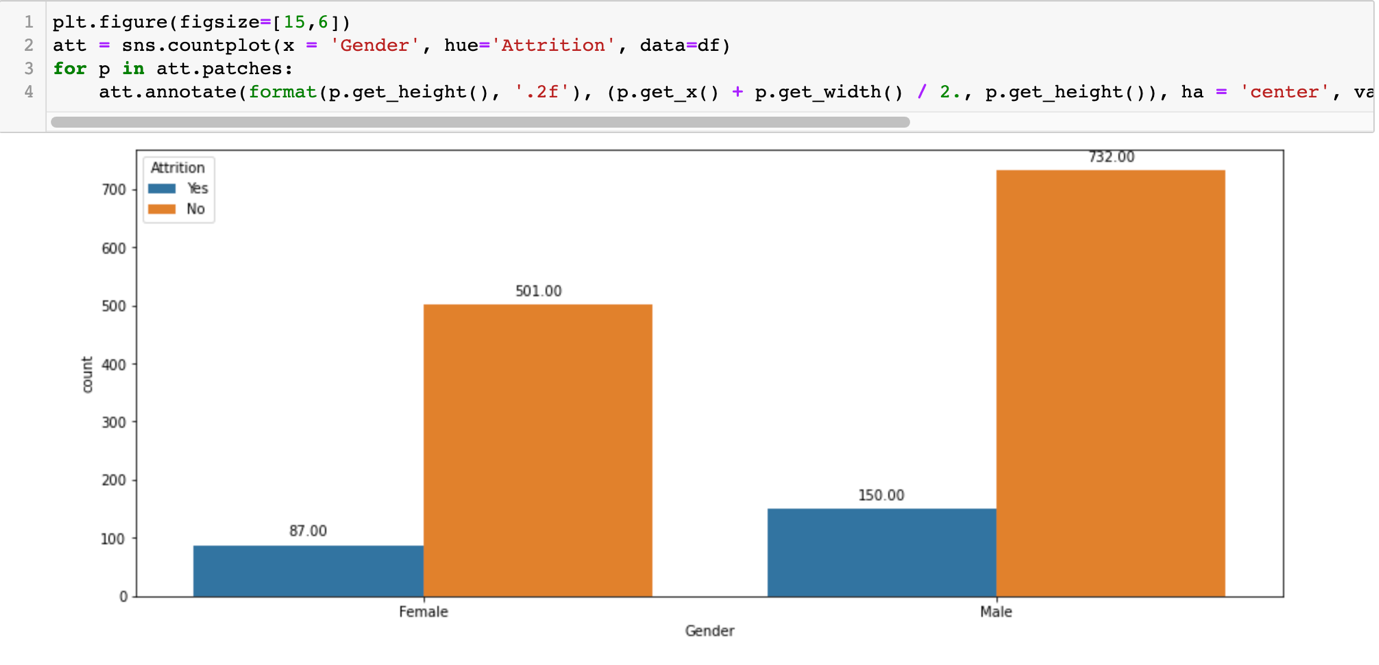


Sales and R&D department are having high attrition rate but comparatively sales department is having the highest rate of attrition among all departments.

1. Education Field and Attrition

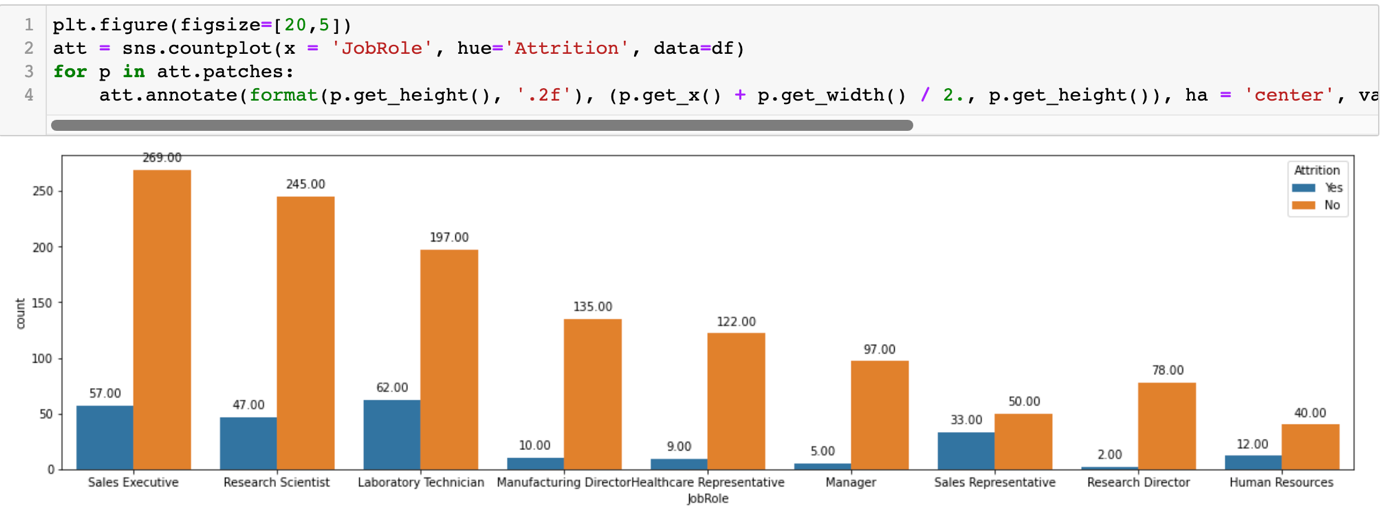
When we see the attrition rate on the basis of their educational field, those who had Human resource as their educational background have highest attrition rate i.e. 26% whereas those who have others as their educational background have the least attrition rate 13%.

1. Gender and Attrition



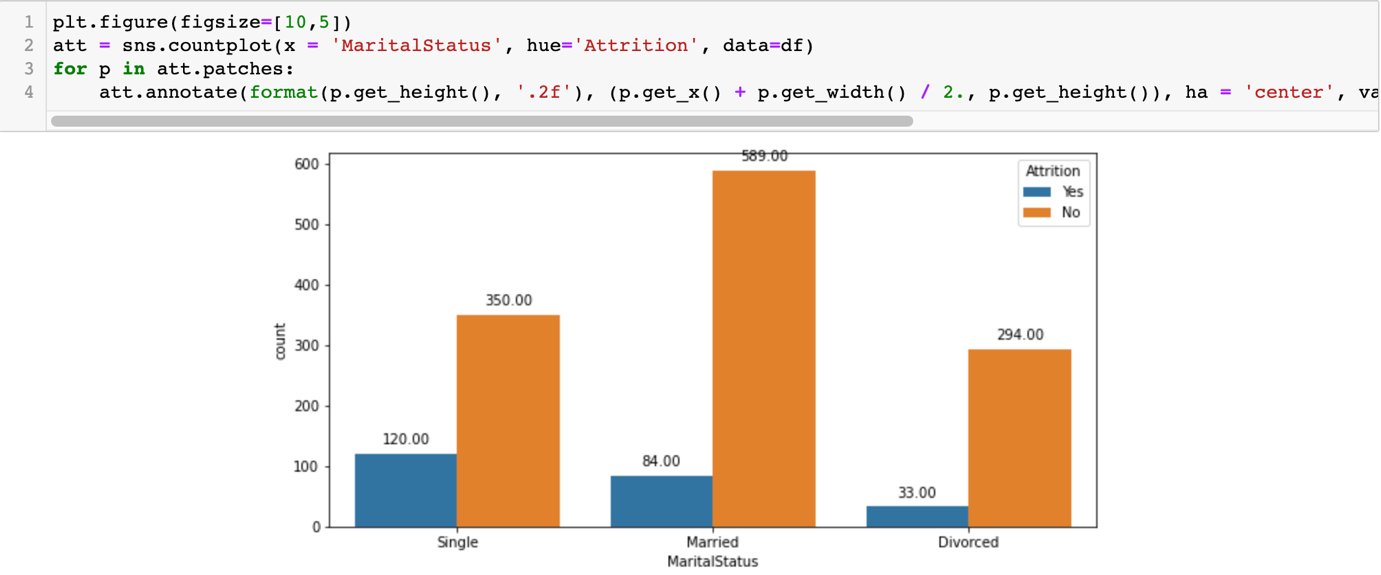
When we look at attrition rate as per the gender, 15% female have left the job where as 17% male have left the job. Hence we can say that there is no significant difference.

1. Job Role and Attrition



The highest attrition we can see is under the role of Sales representative. We will see if we can get further reason why this particular job role have the highest attrition.

1. Marital Status and Attrition



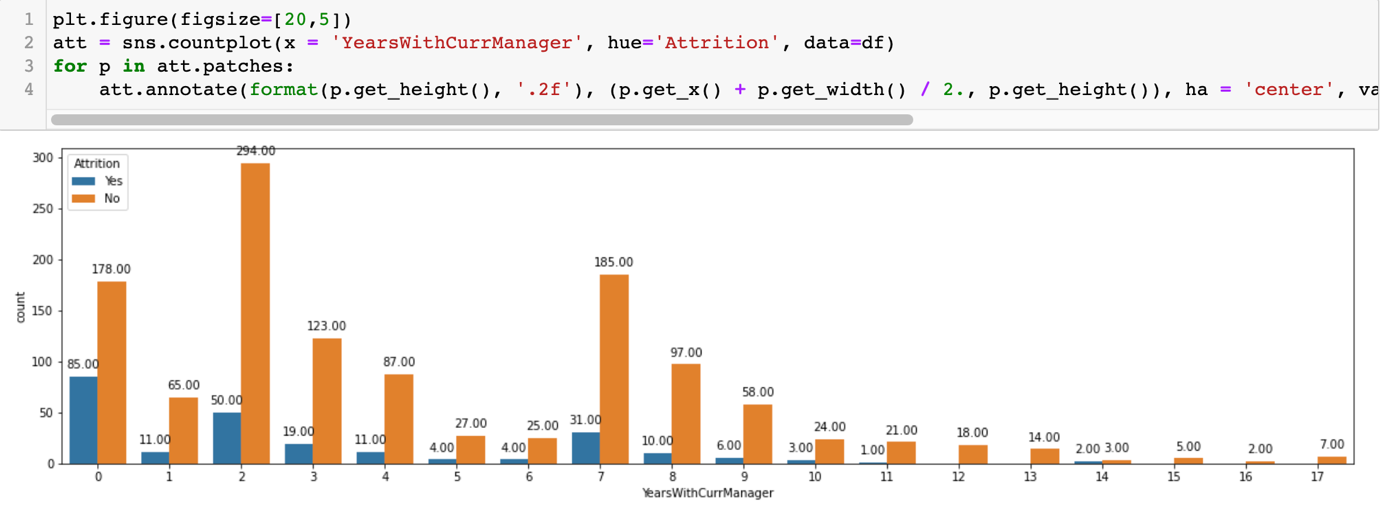
As per martial statuses, Singles are having the highest attrition rate compared to married and Divorced people.

1. Over Time and Attrition



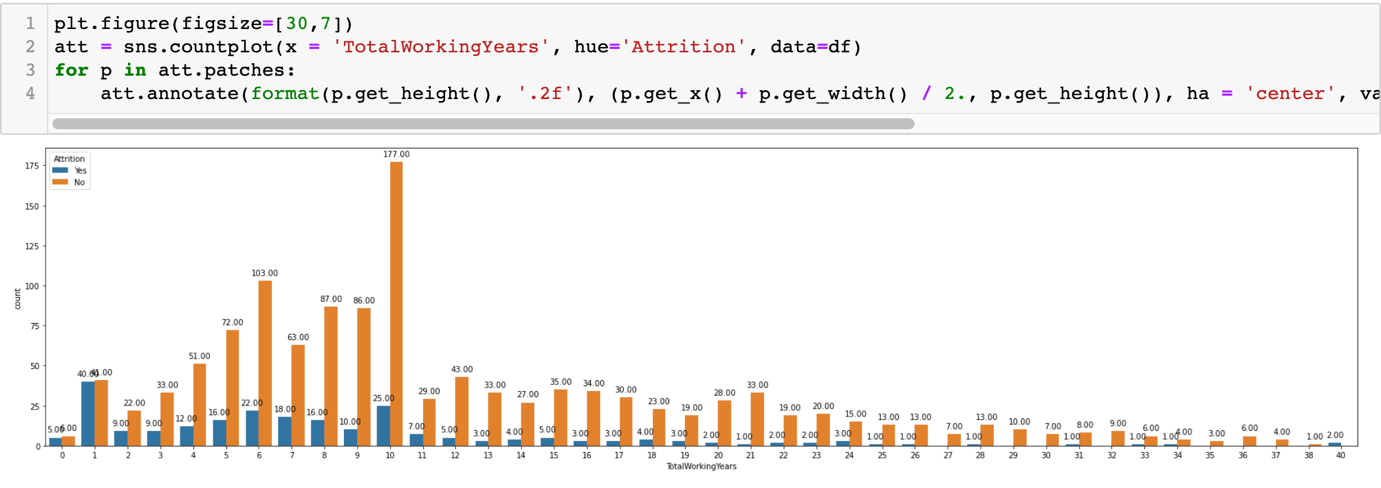
People who are doing overtime are having high attrition rate compared to those who don’t opt for overtime.

1. Years With Current Manager and Attrition



It looks strange but the people under 1 year with the current Managers have the highest attrition rate. This means the people are having trouble in adjusting with their new Managers. Company should make the people familiarize their employees whenever they are assigned under new managers.

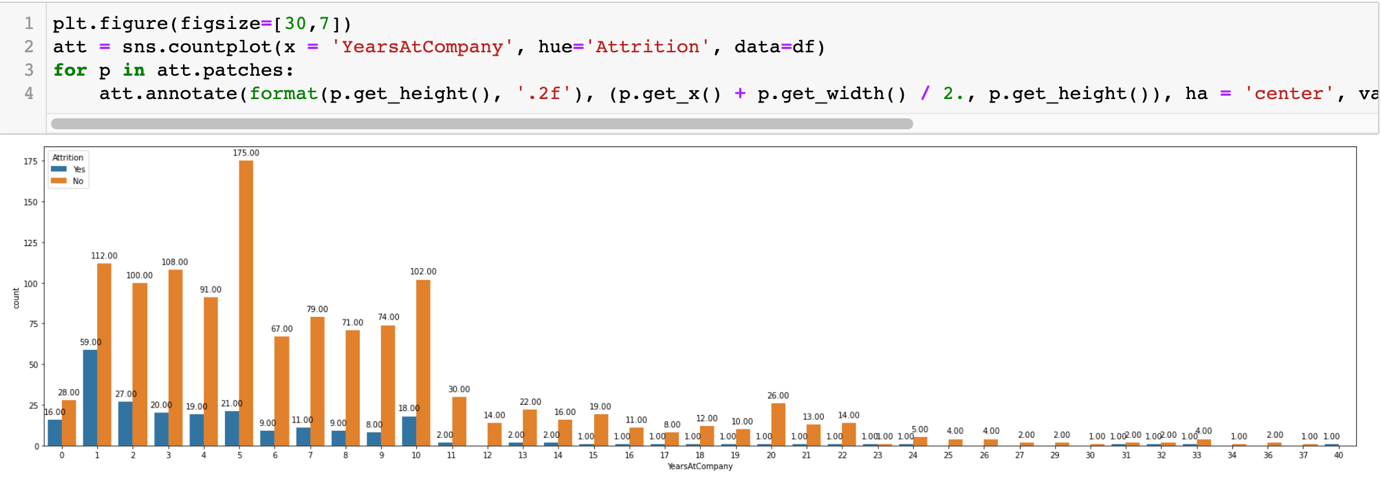
1. Total Working Years and Attrition



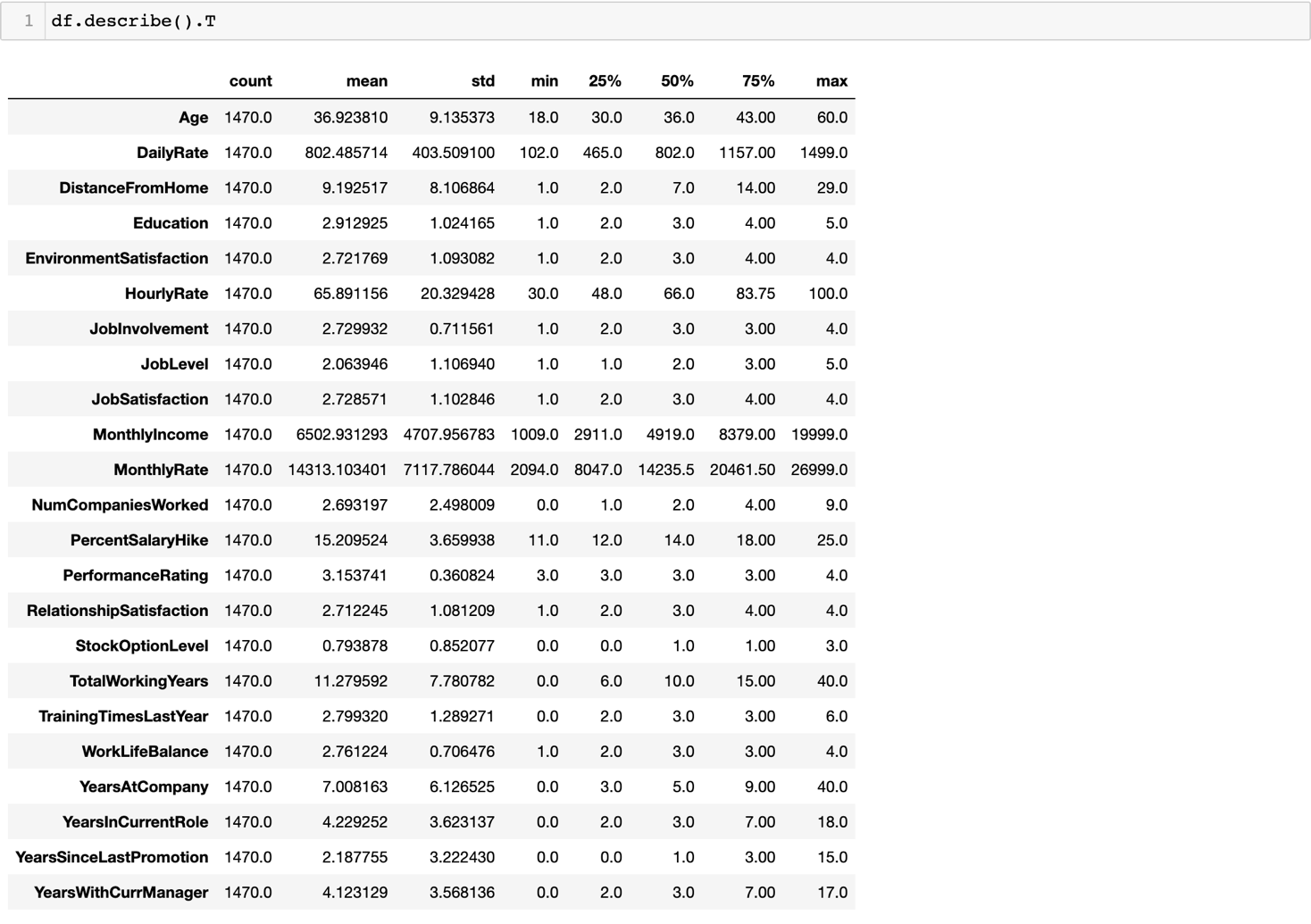
The plot shows that the people in their initial years are having trouble to settle themselves in the organization. As we can see, those who have spent only one years in the any company might leave the job in the search of salary hike or some other benefits.

Once they complete their first year, we can see that the attrition chances are getting down. Hence while hiring new employee, company should consider that the employee should be more than one or years their total work experience.

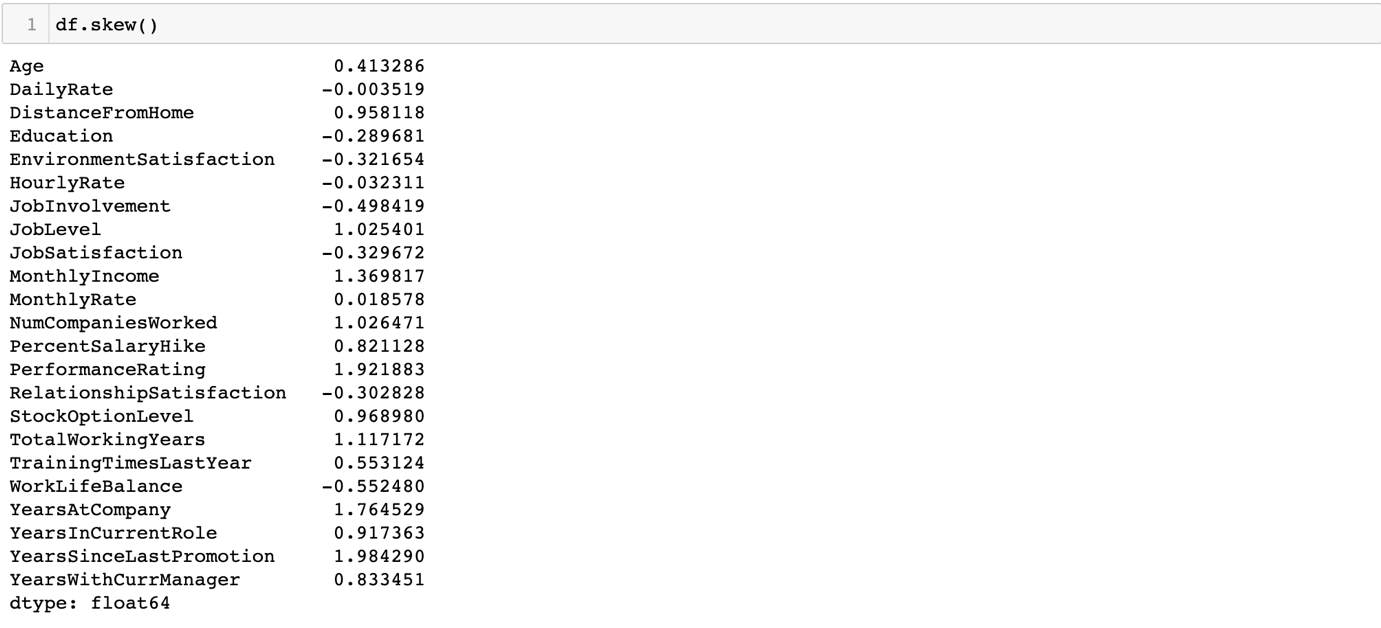
1. Years At Company and Attrition



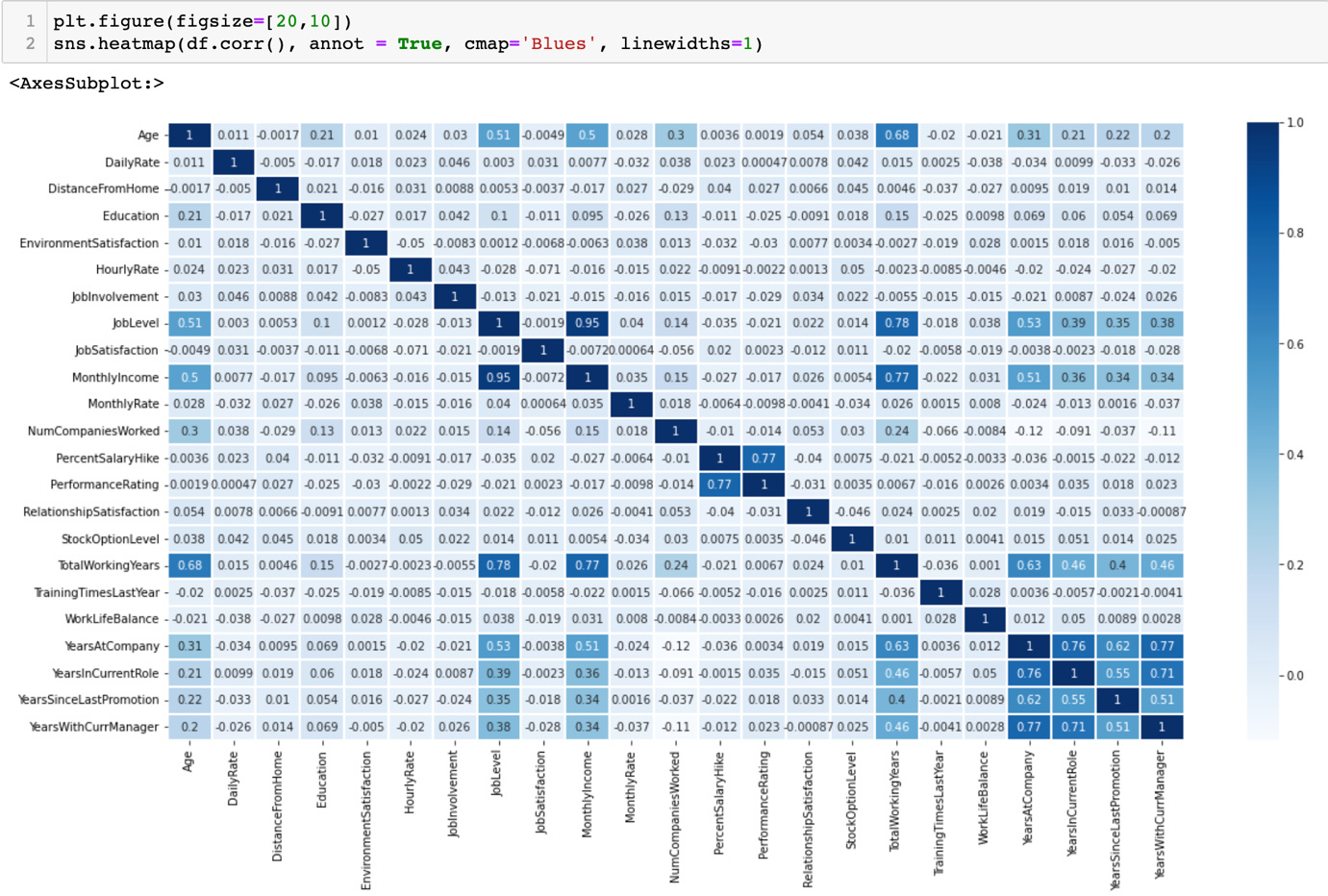
People who are only 1 year old at the company have the highest tendency to leave. The tendency reducing over a period of time. As we can see, in 5th year, the rate of attrition is the least meaning once anyone complete 5 years in the company, until and unless there is strong reason to leave, they will not leave.



* With the help of above describe() method we can get the 5 point summary for the numerical fields in the data. Here we can see the mean, std. deviation, min, 25th%, 50th% (median), 75th% and max for each numerical variable in the data.
* We can also confirm that there are no missing value in any of the column.
* The scales of each column are completely different than each other, hence I have to use the scaler later.
* We also can see that there are some variables having the skewness which we can make sure with the help of skew() method.



As we can confirm that there are some variable have skewness we will try to look for the outliers which are, in most of the cases, reasons for the skewness. But before that let me check the correlation matrix to find out the if there is any problem of multicollinearity.



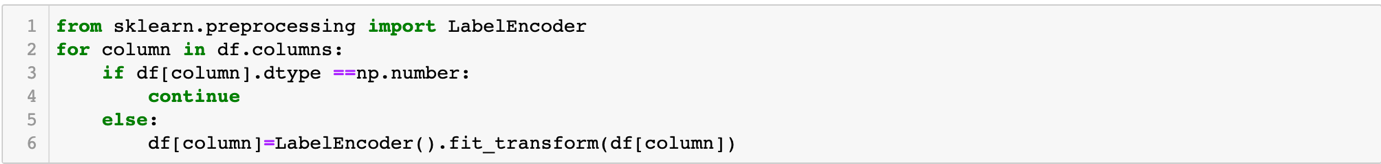
1. Monthly income has high correlation with Job level and Working years, Monthly income is also skewed So I decided to drop Monthly Income.

2. Performance Rating and PercentSalaryHike Highly correlated. Performance Rating is also skewed Hence I decided to drop Performance Rating.

3. YearsAtCompany has high correlation with YearsWithCurManager and YearsInCurRole. So I will drop YearsAtCompany and remove the problem of Multicollinearity.

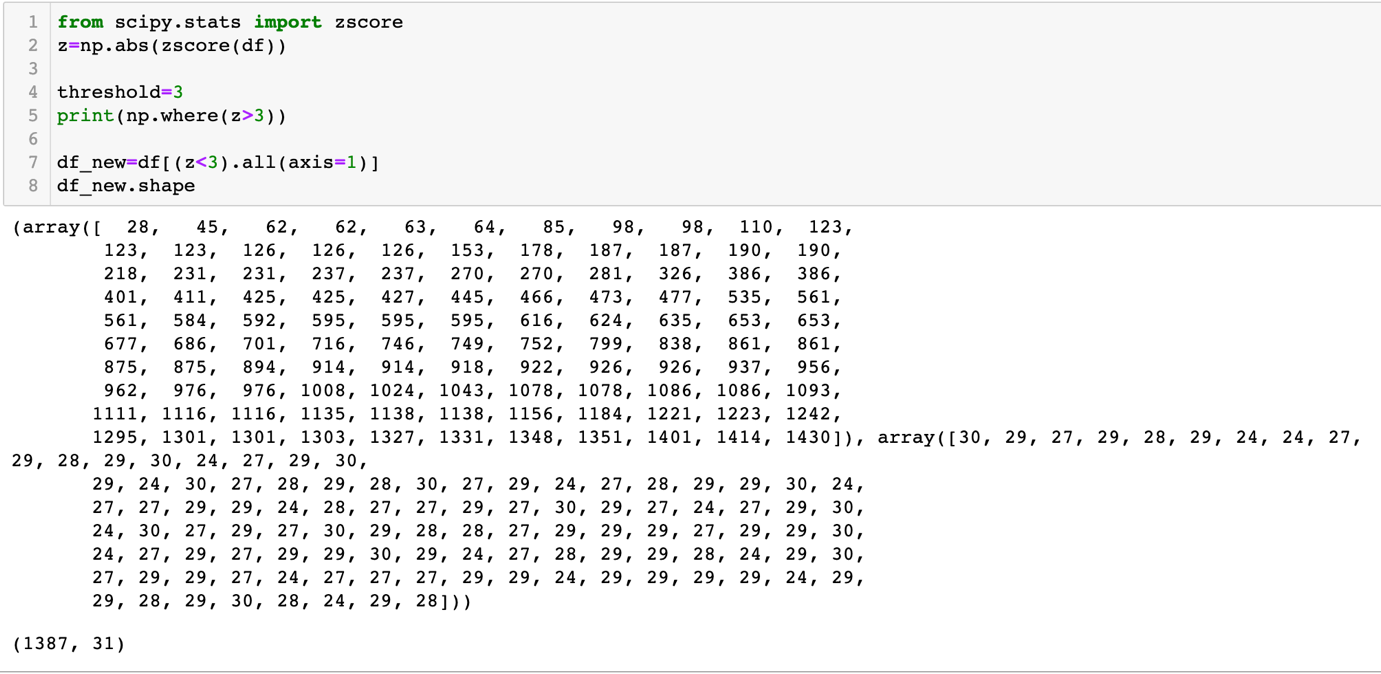


Once this columns are dropped, I will encode the categorical columns so that I can go ahead with the outliers search and then train-test-split.



Now let I wil. check the outliers with the help of z-scores. I have set the threshold as 3, hence If z-score of any datapoint is higher than 3, it will be considered as the outliers.

As per the experts, highest data loss we can afford while removing outliers is 7-8%, beyond which we cannot afford to lose the data. In such cases we have to use other transformation method to reduce the skewness in the datasets.

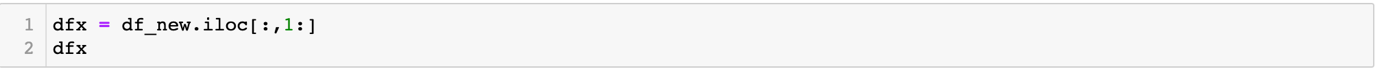


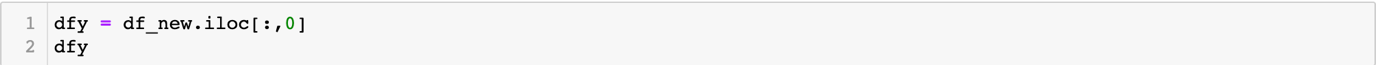
There are only 83 rows getting reduced after removing the outliers i.e. 6%. Hence we can surely afford to move with this new data.

Now I have to separate target column from the dataset, but the problem is our target column is at the 2nd position in the dataset, so first I have to add new variable i.e. new\_age column instead of age and delete the original age variable. This will bring the target column at the 1st position and it will be much more easy for me to separate the target column and feature columns.

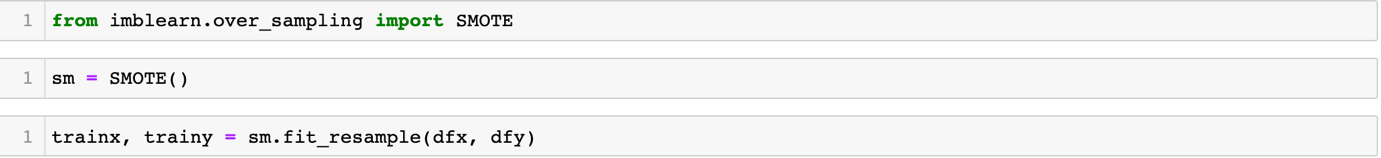


Now let us separate the target and feature columns.





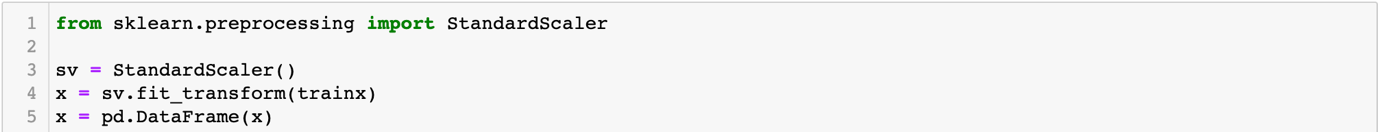
Hereby I have separated the target so that I can tackle the problem of class imbalance we talked about in the start.



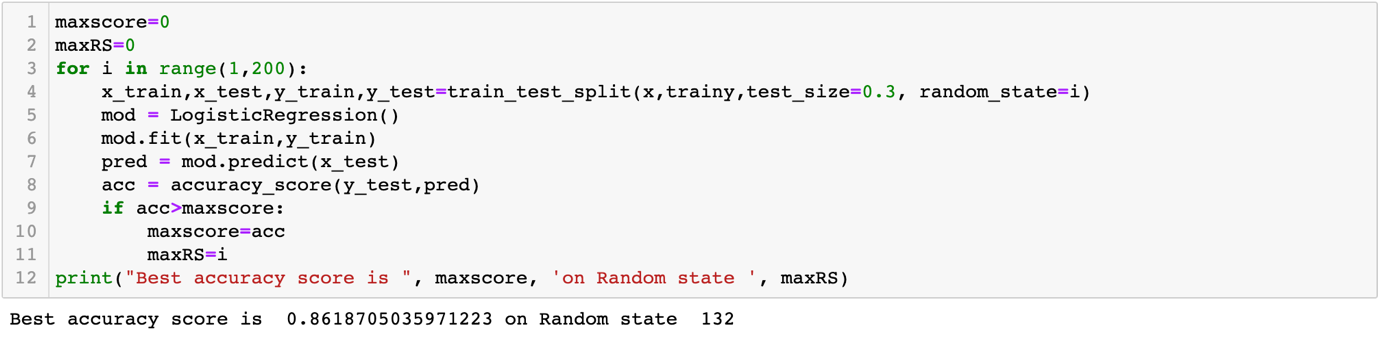
|  |  |
| --- | --- |
| Before SMOTE | After SMOTE |
|  |  |

Earlier we same see, we had 1233 records for No and only 237 records for Yes. But with the help of class imbalance, we have 1158 records for each Yes and No both.

Now I will use standard scaler to bring the data to same scale so that the model we built perform better.

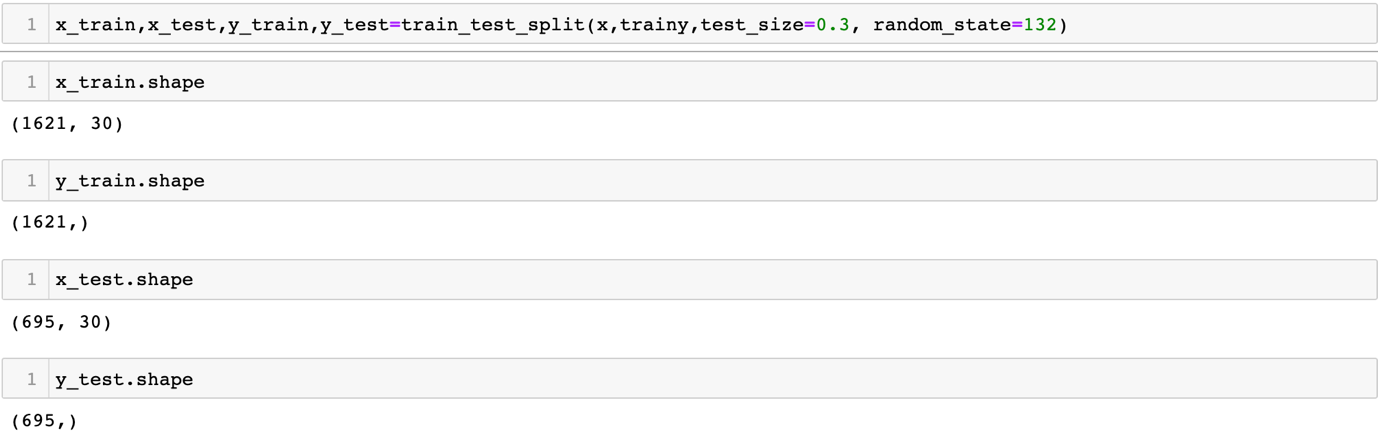


The data has been standardized and ready for train-test-split, but before that I have to find the ideal random state for the splitting. For that I am using LogisticRegression() algorithm.



As we can see, the best radom state we found is 132 with the accuracy of 86% with LogisticRegression() algorithm.

Now let us apply the train-test-split.



I have separated the x\_train, x\_test, y\_train and y\_test. From entire dataset I will use 70% data in training and 30% data for testing the model performance.

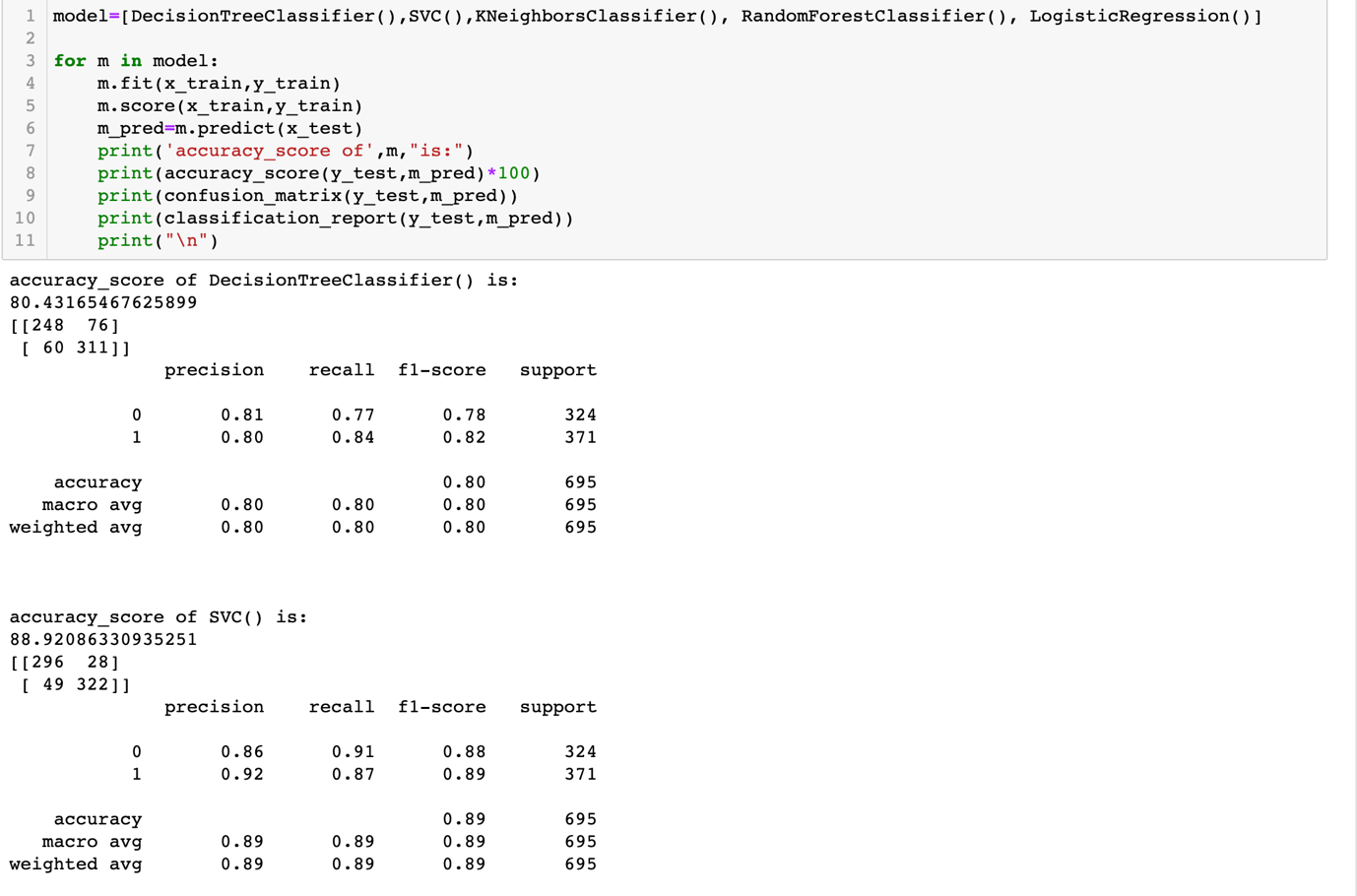
**Building Machine Learning Models:**

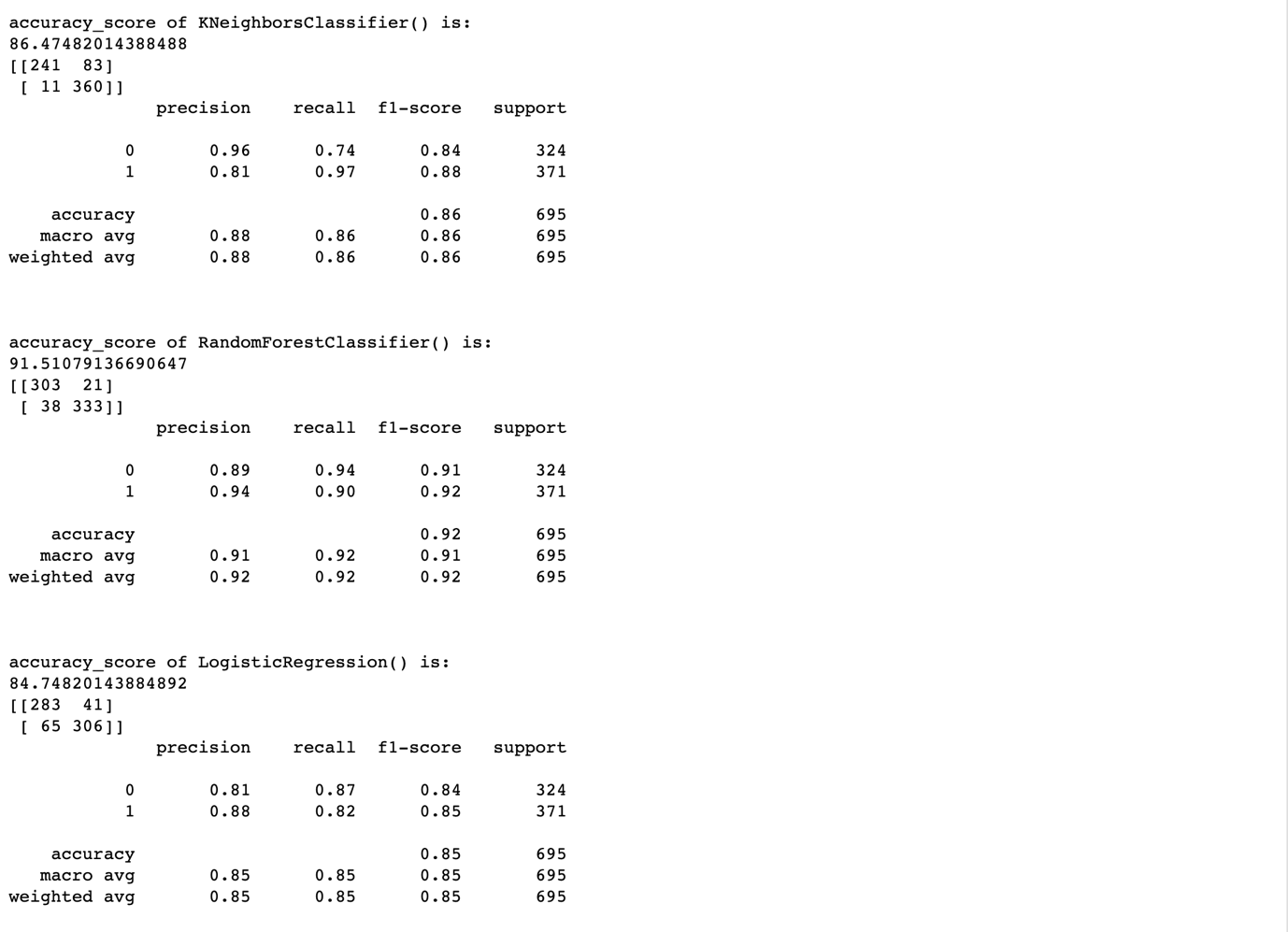
I will use 5 different algorithms as I do not know which algorithm will perform better with this dataset.

1. DecisionTreeClassifier()
2. SVC()
3. KNeighborsClassifier()
4. RandomForestClassifier()
5. LogisticRegression()

And then I will compare their accuracy score with the respective Cross-validation score. The model having the least difference between accuracy score and CV score, will be considered the best model. I will move ahead for hyper tuning the same model.

I will run a for loop which will build 5 different model with each different algorithms one by one. It will print the accuracy score, confusion matrix and classification report which includes precision, recall, f1 score and support.





The models are built, now let me check the CV scores for all the models.

|  |  |  |  |
| --- | --- | --- | --- |
| **Algorithm** | **Accuracy Score** | **CV Score** | **Difference** |
| DecisionTreeClassifier() | 80.43 | 77.11 | 3.32 |
| SVC() | 88.92 | 87.82 | 1.1 |
| KNeighborsClassifier() | 86.47 | 79.62 | 6.85 |
| RandomForestClassifier() | 91.51 | 88.42 | 3.09 |
| LogisticRegression() | 84.74 | 79.01 | 5.73 |

Support Vector Classifier algorithm has the lowest difference between in its accuracy score and CV score i.e. only 1.1%.

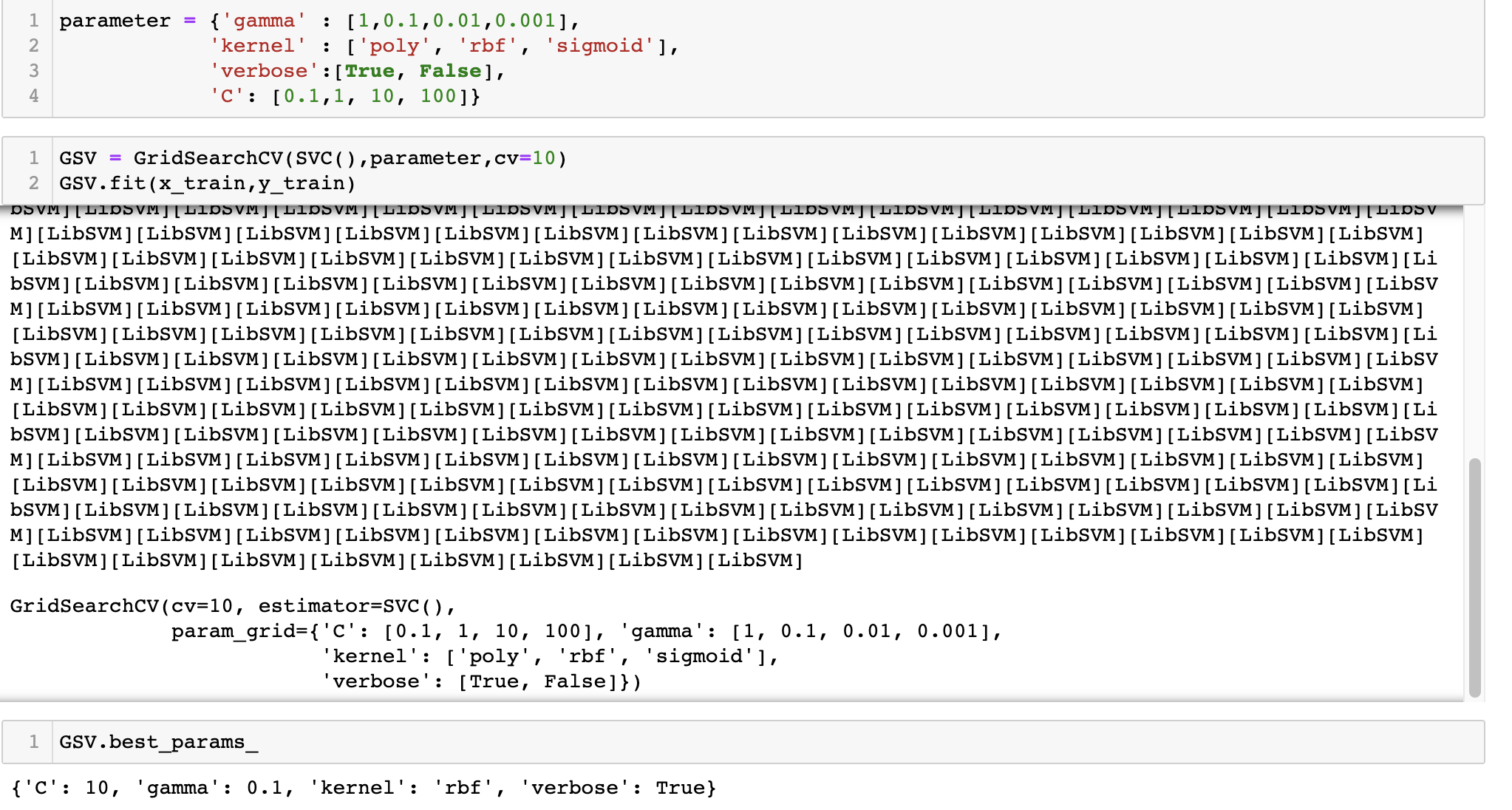
**SVM :**

A support vector machine (SVM) is a supervised machine learning model that uses classification algorithms for two-group classification problems. After giving an SVM model sets of labelled training data for each category, they’re able to categorize the targets.

Compared to newer algorithms like neural networks, they have two main advantages: higher speed and better performance with a limited number of samples (in the thousands). This makes the algorithm very suitable for text classification problems, where it’s common to have access to a dataset of at most a couple of thousands of tagged samples.

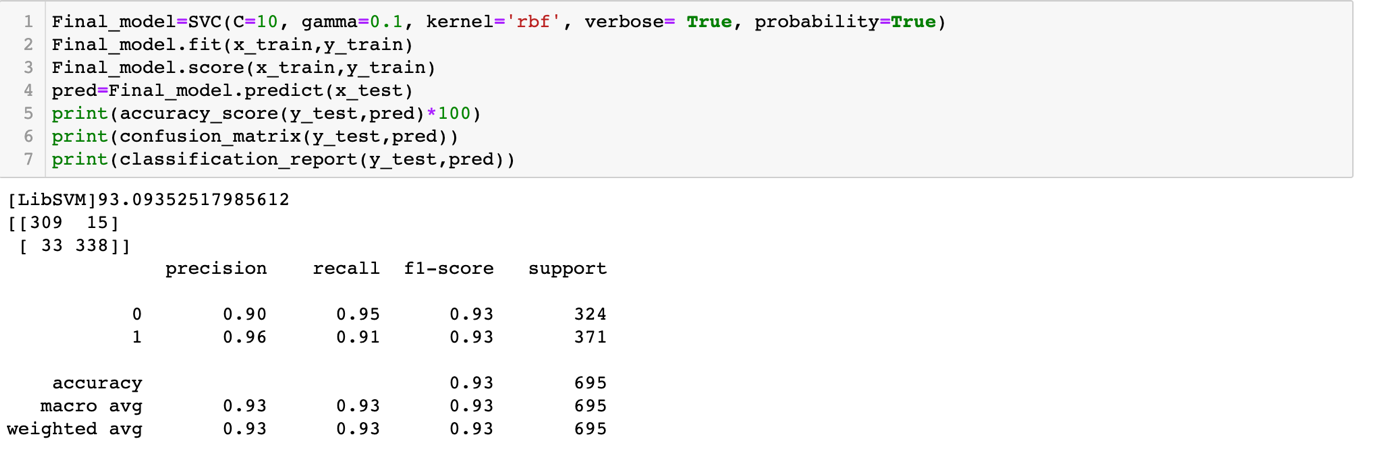
**Hyperparameter Tuning:**

I have selected four parameters on which I will perform the hyperparameter tuning.



I have found the best parameter which we can use in the final model.

**Final Model:**

****

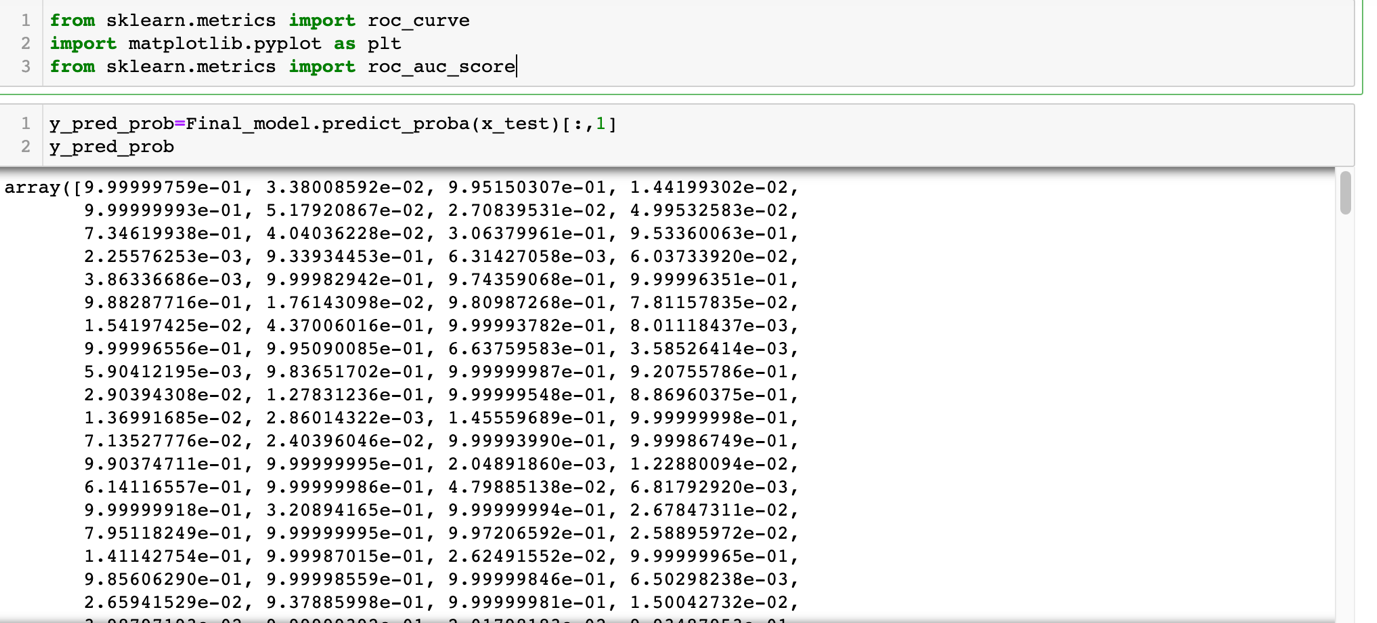
Our model is ready and we have achieved 93% of accuracy which is great.

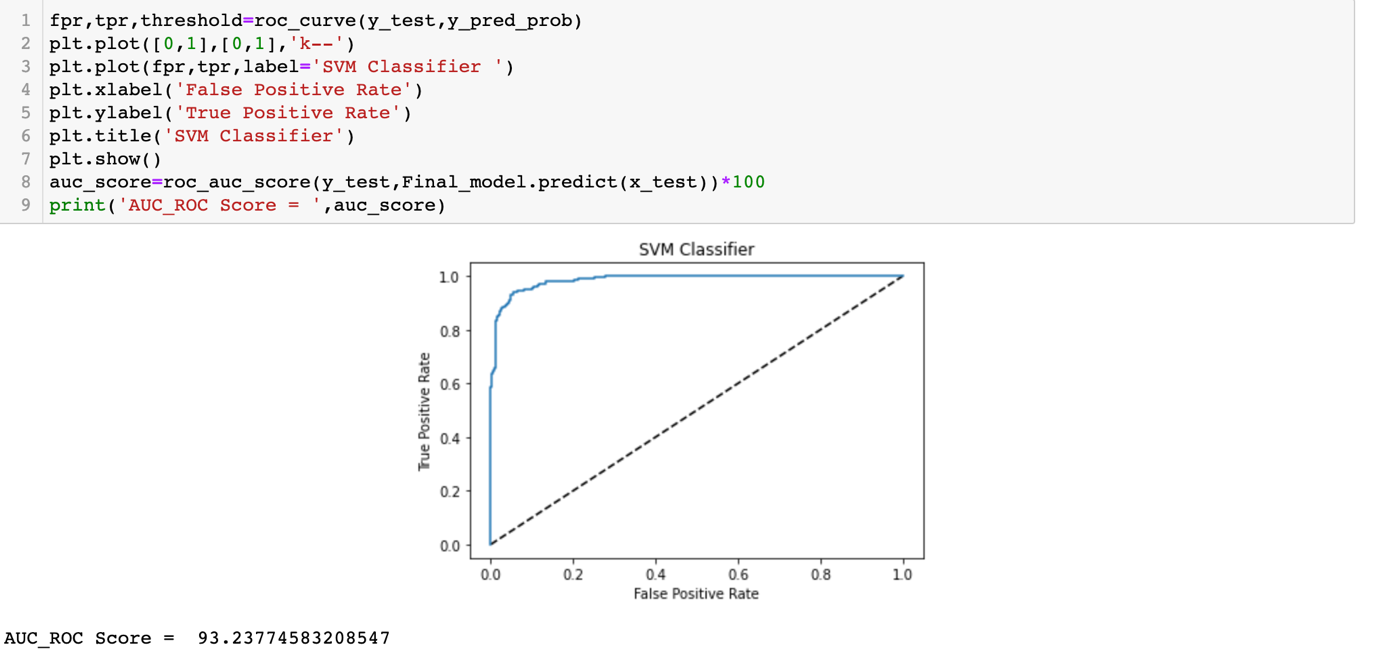
Another matrix that we have used is Confusion matrix. As per our confusion matrix, out of 695, 15+33 = 48 people are wrongly classified.

The weighted average for the precision, recall and f1 score is also good.

**AUR-ROC Curve :**

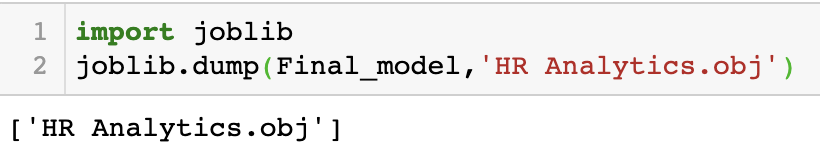
With the help of AUC-ROC curve we can evaluate and compare our binary classifier. This curve plots the true positive rate (recall) against the false positive rate (ratio of incorrectly classified negative instances), instead of plotting the precision versus the recall.

****

****

The dotted line in the middle represents a any random classifier and so our classifier should be as far away from it as possible. Our Support vector classifier model is doing much more better than any random classifier. As per AUC-ROC curve the score we got is very close to what we got from SVC.

**Model Saving(Dumping):**

****

I have saved the model in .obj file format. This will help me to load the model anytime and anywhere I want it.

**Predictions**:



At last, I have compared the actual values of target vs. predicted values, so that we can see how model is predicting the results.

**Summary**:

I initiated this project with the EDA(Exploratory Data Analysis) where I got a understanding of the dataset, checked about missing values and learned which features which were important. During this process I used seaborn and matplotlib to do the visualizations. During the data pre-processing part, I computed missing values, converted features into numeric ones. Then, I started training 5 different machine learning models, selected one of them i.e. SVC and ran cross validation on it. Then I Introduced SVC and tuned it’s performance through optimizing it’s 4 hyperparameter values. I also looked at its confusion matrix, precision, recall and f-score.

I ran the AUC-ROC curve as another measure to validate the model and saved the model for the future reference. At last, I printed the actual and predicted results.