**MACHINE LEARNING**

**(Churn Prediction)**

*Summer Internship Report Submitted in partial fulfillment of the requirement for undergraduate degree of*

**Bachelor of Technology**

**In**

**Computer Science Engineering**

**By**

**Akhila Gopalakrishna Penukonda**

**221710309045**

*Under the Guidance of*

**Mr. Hanumantha Rao Kondamuri**

**( Assistant Professor )**



Department Of Computer Science Engineering

GITAM School of Technology

GITAM (Deemed to be University)

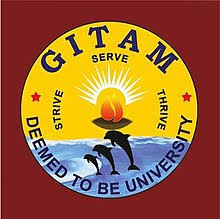
Hyderabad-502329

June 2019

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING SCHOOL OF TECHNOLOGY**

**GANDHI INSTITUTE OF TECHNOLOGY AND MANAGEMENT(GITAM) (Declared as Deemed-to-be-University u/s 3 of UGC Act 1956)**

**HYDERABAD CAMPUS**



**DECLARATION**

I submit this industrial training work entitled ​**“CHURN PREDICTION** ” to GITAM (Deemed To Be University), Hyderabad in partial fulfillment of the requirements for the award of the degree of “​**Bachelor of Technology**​” in “**Computer Science** **Engineering**​”. I declare that it was carried out independently by me under the guidance of ​**Mr. Hanumantha Rao Kondamuri** ​,

Assistant Professor, GITAM (Deemed To Be University), Hyderabad, India.

The results embodied in this report have not been submitted to any other University or Institute for the award of any degree or diploma.

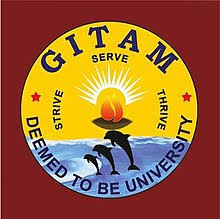
Place: HYDERABAD StudentName:Akhila Gopalakrishna Penukonda

Date:20th-JULY-2020 StudentRollno:221710309045

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING SCHOOL OF TECHNOLOGY**

**GANDHI INSTITUTE OF TECHNOLOGY AND MANAGEMENT(GITAM) (Declared as Deemed-to-be-University u/s 3 of UGC Act 1956)**

**HYDERABAD CAMPUS**

****

**CERTIFICATE**

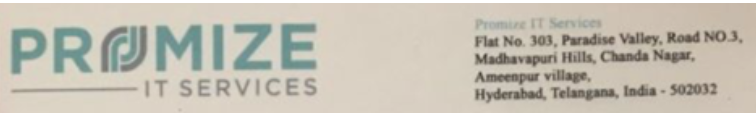
This is to certify that the Industrial Training Report entitled “**CHURN PREDICTION**” ​is being submitted by Akhila Gopalakrishna Penukonda (221710309045 ) in partial fulfillment of the requirement for the award of Bachelor of Technology ​in Computer Science Engineering at GITAM (Deemed To Be University), Hyderabad during the academic year 2020-21 .

It is faithful record work carried out by her at the Computer Science Engineering Department​, GITAM University Hyderabad Campus under my guidance and supervision.

**Mr. Hanumantha Rao Kondamuri Dr.Phani Kumar S**

**( Assistant Professor ) Professor and HOD**

**Department of CSE**

****

**CERTIFICATE**

This is to certify that Internship titled **“CHURN PREDICTION”** is the bona fide work carried out by student of theGITAM (Deemed To Be University), Hyderabad ,in partial fulfillment for the award of ​**Bachelor of Technology**​ in Computer Science Engineering​ . During the period 25th May 2020 - 30th  July 2020 at **PROMIZE IT SERVICES PRIVATE LIMITED**. During this period her conduct was found to be very good and she has shown good technical skills.

****

**ACKNOWLEDGEMENT**

Apart from my effort, the success of this internship largely depends on the encouragement and guidance of many others. I take this opportunity to express my gratitude to the people who have helped me in the successful competition of this internship.I would like to thank respected ​**Dr. N. Siva Prasad,** ​Pro Vice Chancellor, GITAM Hyderabad and ​**Dr. CH. Sanjay,** ​Principal, GITAM Hyderabad.I would like to thank respected ​**Dr. K. Manjunathachari,** ​Head of the Department of Electronics and Communication Engineering for giving me such a wonderful opportunity to expand my knowledge for my own branch and giving me guidelines to present an internship report. It helped me a lot to realize what we study for.I would like to thank the respected faculties ​**Mr.** ​ **Hanumantha Rao Kondamuri** who helped me to make this internship a successful accomplishment.I would also like to thank my friends who helped me to make my work more organized and well-stacked till the end.

StudentName:Akhila Gopalakrishnan Penukonda

StudentRollno:221710309045

**CONTENTS**

1. **Chapter 1: Machine learning**

* 1.2 IMPORTANCE OF MACHINE LEARNING
* 1.2 IMPORTANCE OF MACHINE LEARNING
* 1.3 USES OF MACHINE LEARNING
* 1.4 TYPES OF LEARNING ALGORITHMS

1.4.1 Supervised Learning

1.4.2 Unsupervised learning

1.4.3 Semi Supervised Learning

- 1.5 RELATION BETWEEN DATA MINING,MACHINE LEARNING AND DEEP LEARNING

1. **Chapter 2: Python**

* 2.1 INTRODUCTION TO PYTHON
* 2.2 HISTORY OF PYTHON
* 2.3 FEATURES OF PYTHON
* 2.4 HOW TO SETUP PYTHON

2.4.1 Installation(using python IDLE)

2.4.2 Installation(using Anaconda)

* 2.5 PYTHON VARIABLE TYPES:

2.5.1 Python Number

2.5.2 Python Strings

2.5.3 Python Lists

2.5.4 Python Tuples

2.5.5 PythonDictionary

* 2.6 PYTHON FUNCTION

2.6.1 Defining a Function

2.6.2 Calling a Function

* 2.7 PYTHON USING OOPs CONCEPTS:

2.7.1 Class

1. **Chapter 3: Case study**

* 3.1 PROBLEM STATEMENT
* 3.2 DATA SET
* 3.3 OBJECTIVE OF THE CASE STUDY

1. **Chapter 4: Model Building**

* 4.1 PREPROCESSING OF THE DATA

4.1.1 GETTING THE DATASET

4.1.2 IMPORTING THE LIBRARIES

4.1.3 IMPORTING THE DATA-SET

4.1.4 HANDLING MISSING VALUES

4.1.5 CATEGORICAL DATA

* 4.2. DATA PROCESSING
* 4.3 DATA VISUALIZATIONS WITH STATISTICAL METHODS
* 4.4 TRAINING THE MODEL
* 4.5 EVALUATING THE CASE STUDY
* 4.6 BUILDING THE MODE

4.6.1Building the model (using splitting)

4.6.2 Comparing the performance of all three models:

4.6.3 Hyperparameter tuning using GridSearchCV

4.6.3.1 What is Hyperparameter tuning ?

4.6.3.2 What is GridSearchCV?

1. **Conclusion**
2. **References**

**LIST OF FIGURES**

**CHAPTER 1: MACHINE LEARNING**

**1.1 INTRODUCTION:**

​Machine Learning(ML) is the scientific study of algorithms and statistical models that computer systems use in order to perform a specific task effectively without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of Artificial Intelligence(AI).

**1.2 IMPORTANCE OF MACHINE LEARNING:**

Consider some of the instances where machine learning is applied: the self-driving Google car, cyber fraud detection, online recommendation engines—like friend suggestions on Facebook, Netflix showcasing the movies and shows you might like, and “more items to consider” and “get yourself a little something” on Amazon—are all examples of applied machine learning. All these examples echo the vital role machine learning has begun to take in today’s data-rich world.

Machines can aid in filtering useful pieces of information that help in major advancements, and we are already seeing how this technology is being implemented in a wide variety of industries.With the constant evolution of the field, there has been a subsequent rise in the uses, demands, and importance of machine learning. Big data has become quite a buzzword in the last few years; that’s in part due to increased sophistication of machine learning, which helps analyze those big chunks of big data. Machine learning has also changed the way data extraction, and interpretation is done by involving automatic sets of generic methods that have replaced traditional statistical techniques. The process flow depicted here represents how machine learning works.

****

Figure1 : the process flow

**1.3 USES OF MACHINE LEARNING:**

Earlier in this article, we mentioned some applications of machine learning. To understand the concept of machine learning better, let’s consider some more examples: web search results, real-time ads on web pages and mobile devices, email spam filtering, network intrusion detection, and pattern and image recognition. All these are by-products of applying machine learning to analyze huge volumes of data

Traditionally, data analysis was always being characterized by trial and error, an approach that becomes impossible when data sets are large and heterogeneous. Machine learning comes as the solution to all this chaos by proposing clever alternatives to analyzing huge volumes of data. By developing fast and efficient algorithms and data-driven models for real-time processing of data, machine learning can produce accurate results and analysis.

**1.4 TYPES OF LEARNING ALGORITHMS:**

The types of machine learning algorithms differ in their approach, the type of data they input and output, and the type of task or problem that they are intended to solve.

**1.4.1 Supervised Learning :**

When an algorithm learns from example data and associated target responses that can consist of numeric values or string labels, such as classes or tags, in order to later predict the correct response when posed with new examples comes under the category of supervised learning.

Supervised machine learning algorithms uncover insights, patterns, and relationships from a labelled training dataset – that is, a dataset that already contains a known value for the target variable for each record. Because you provide the machine learning algorithm with the correct answers for a problem during training, it is able to “learn” how the rest of the features relate to the target, enabling you to uncover insights and make predictions about future outcomes based on historical data.

Examples of Supervised Machine Learning Techniques are Regression, in which the algorithm returns a numerical target for each example, such as how much revenue will be generated from a new marketing campaign.

Classification, in which the algorithm attempts to label each example by choosing between two or more different classes. Choosing between two classes is called binary classification, such as determining whether or not someone will default on a loan. Choosing between more than two classes is referred to as multiclass classification.

**1.4.2 Unsupervised Learning:**

​When an algorithm learns from plain examples without any associated response, leaving to the algorithm to determine the data patterns on its own. This type of algorithm tends to restructure the data into something else, such as new features that may represent a class or a new series of uncorrelated values. They are quite useful in providing humans with insights into the meaning of data and new useful inputs to supervised machine learning algorithms.

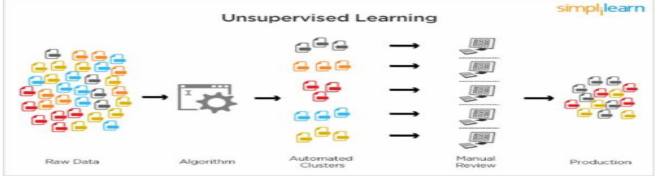


Figure 2 : Unsupervised Learning

Popular techniques where unsupervised learning is used also include self-organizing maps, nearest neighbor mapping, singular value decomposition, and k-means clustering. Basically, online recommendations, identification of data outliers, and segment text topics are all examples of unsupervised learning.

**1.4.3 Semi Supervised Learning:**

As the name suggests, semi-supervised learning is a bit of both supervised and unsupervised learning and uses both labeled and unlabeled data for training. In a typical scenario, the algorithm would use a small amount of labeled data with a large amount of unlabeled data.

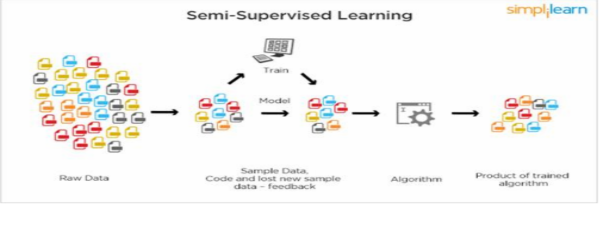


Figure 3 : Semi Supervised Learning

**1.5 RELATION BETWEEN DATA MINING,MACHINE LEARNING AND DEEP LEARNING:**

Machine learning and data mining use the same algorithms and techniques as data mining, except the kinds of predictions vary. While data mining discovers previously unknown patterns and knowledge, machine learning reproduces known patterns and knowledge—and further automatically applies that information to data, decision-making, and actions.

Deep learning, on the other hand, uses advanced computing power and special types of neural networks and applies them to large amounts of data to learn, understand, and identify complicated patterns. Automatic language translation and medical diagnoses are examples of deep learning.

**CHAPTER 2:PYTHON**

Basic programming language used for machine learning is : PYTHON

**2.1 INTRODUCTION TO PYTHON:**

● Python is a high-level, interpreted, interactive and object-oriented scripting language.  
 ● Python is a general purpose programming language that is often applied in scripting roles  
 ● Python is Interpreted: Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is like PERL and PHP.  
 ● Python is Interactive: You can sit at a Python prompt and interact with the interpreter directly to write your programs.  
 ● Python is Object-Oriented: Python supports the Object-Oriented style or technique of programming that encapsulates code within objects.  
   
**2.2 HISTORY OF PYTHON:**● Python was developed by GUIDO VAN ROSSUM in early 1990’s

● Its latest version is 3.7 , it is generally called as python3  
   
**2.3 FEATURES OF PYTHON:**

**●** Easy-to-learn: Python has few keywords, simple structure, and a clearly defined syntax, This allows the student to pick up the language quickly.

● Easy-to-read: Python code is more clearly defined and visible to the eyes.  
● Easy-to-maintain: Python's source code is fairly easy-to-maintaining.

● A broad standard library: Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.

● Portable: Python can run on a wide variety of hardware platforms and has the same interface on all platforms.  
 ● Extendable: You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.  
● Databases: Python provides interfaces to all major commercial databases.  
● GUI Programming: Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.

**2.4 HOW TO SETUP PYTHON:**● Python is available on a wide variety of platforms including Linux and Mac OS X. Let's understand how to set up our Python environment.  
● The most up-to-date and current source code, binaries, documentation, news, etc., is available on the official website of Python.

**2.4.1 Installation(using python IDLE):**● Installing python is generally easy, and nowadays many Linux and Mac OS distributions include a recent python.  
 ● Download python from www.python.org  
 ● When the download is completed, double click the file and follow the instructions  
 to install it.  
 ● When python is installed, a program called IDLE is also installed along with it. It provides a graphical user interface to work with python.

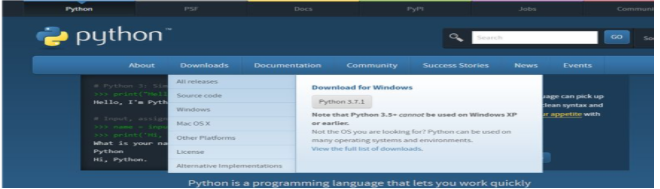


Figure 4 : Python download

**2.4.2 Installation(using Anaconda):**● Python programs are also executed using Anaconda.  
● Anaconda is a free open source distribution of python for large scale data processing, predictive analytics and scientific computing.  
● Conda is a package manager that quickly installs and manages packages.

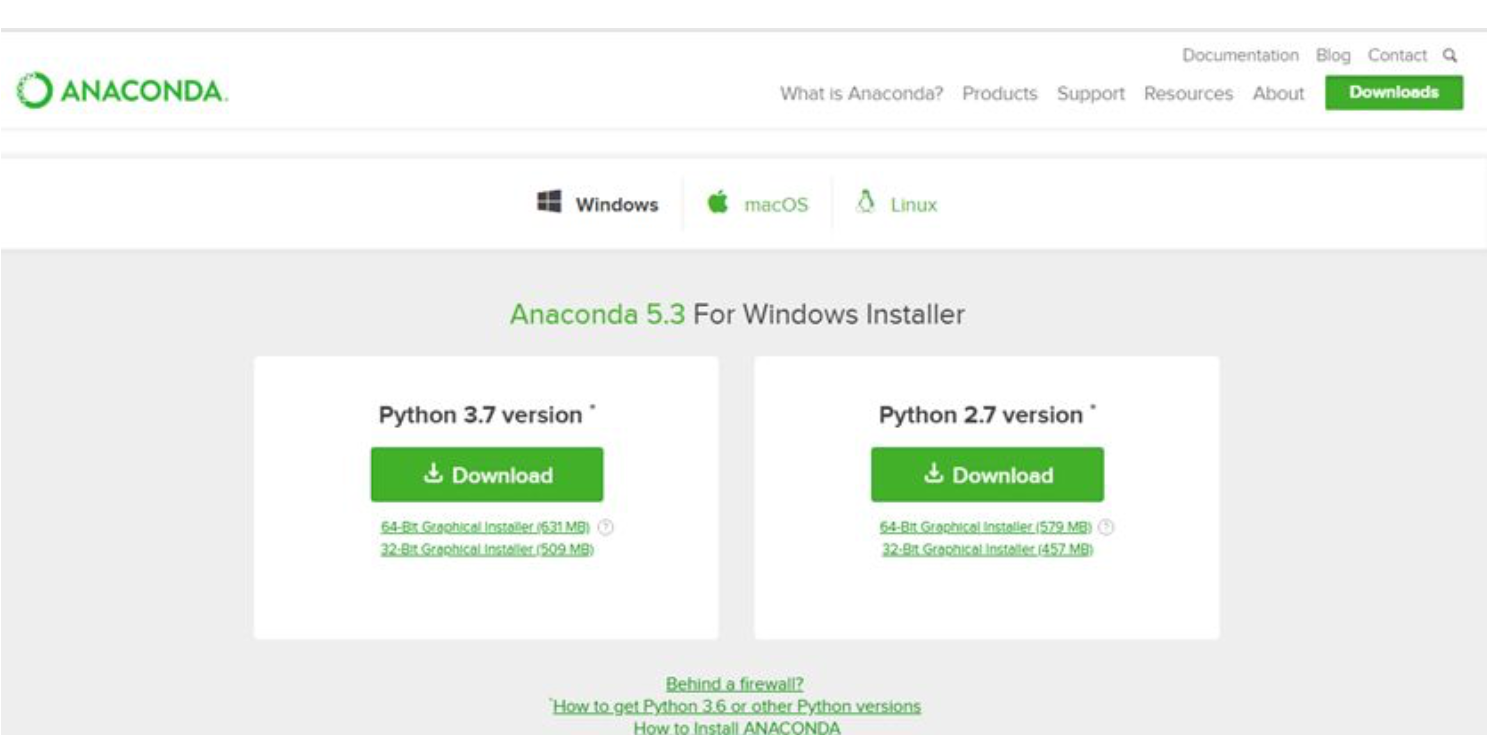
● In WINDOWS:  
 ● Step 1: Open Anaconda.com/downloads in a web browser.  
 ● Step 2: Download python 3.4 version for (32-bit graphic installer/64 -bigraphic installer)  
 ● Step 3: select installation type( all users)  
 ● Step 4: Select path(i.e. add anaconda to path & register anaconda as default python 3.4) next click install and next click finish  
 ● Step 5: Open jupyter notebook ( it opens in default browser)   
  Figure 5 : Anaconda download



Figure 6 : Jupyter notebook

**2.5 PYTHON VARIABLE TYPES:**

**●** Variables are nothing but reserved memory locations to store values. This means that when you create a variable you reserve some space in memory.  
**●** Variables are nothing but reserved memory locations to store values.  
**●** Based on the data type of a variable, the interpreter allocates memory and decides what can be stored in the reserved memory. **●** Python variables do not need explicit declaration to reserve memory space. The declaration happens automatically when you assign a value to a variable.  
**●** Python has various standard data types that are used to define the operations possible on them and the storage method for each of them. **●** Python has five standard data types –

1. Numbers
2. Strings
3. Lists
4. Tuples
5. Dictionary

**2.5.1 Python Numbers:** ● Number data types store numeric values. Number objects are created when you assign a value to them.  
● Python supports four different numerical types − int (signed integers) long (long integers, they can also be represented in octal and hexadecimal) float (floating point real values) complex (complex numbers).  
 **2.5.2 Python Strings:**● Strings in Python are identified as a contiguous set of characters represented in the quotation marks. ● Python allows for either pairs of single or double quotes.  
● Subsets of strings can be taken using the slice operator ([ ] and [:] ) with indexes starting at 0 in the beginning of the string and working their way from -1 at the end. ● The plus (+) sign is the string concatenation operator and the asterisk (\*) is the repetition operator.  
**2.5.3 Python Lists:**● Lists are the most versatile of Python's compound data types. ● A list contains items separated by commas and enclosed within square brackets ([]) ● To some extent, lists are similar to arrays in C. One difference between them is that all the items belonging to a list can be of different data type.  
● The values stored in a list can be accessed using the slice operator ([ ] and [:]) with indexes starting at 0 in the beginning of the list and working their way to end -1.

● The plus (+) sign is the list concatenation operator, and the asterisk (\*) is the repetition operator.

**2.5.4 Python Tuples:**● A tuple is another sequence data type that is similar to the list. ● A tuple consists of a number of values separated by commas. Unlike lists, however,tuples are enclosed within parentheses.  
● The main differences between lists and tuples are: Lists are enclosed in brackets ( [ ] ) and their elements and size can be changed, while tuples are enclosed in parentheses ( ( ) ) and cannot be updated. ● Tuples can be thought of as read-only lists.  
● For example − Tuples are fixed size in nature whereas lists are dynamic. In other words, a tuple is immutable whereas a list is mutable. You can't add elements to a tuple. Tuples have no append or extend method. You can't remove elements from a tuple. Tuples have no remove or pop method.  
**2.5.5 PythonDictionary:**● Python's dictionaries are a kind of hash table type. They work like associative arrays or hashes found in Perl and consist of key-value pairs. A dictionary key can be almost any Python type, but are usually numbers or strings. Values, on the other hand, can be any arbitrary Python object . ● Dictionaries are enclosed by curly braces ({ }) and values can be assigned and accessed using square braces ([]).  
● You can use numbers to "index" into a list, meaning you can use numbers to find out what's in lists. You should know this about lists by now, but make sure you understand that you can only use numbers to get items out of a list.  
● What a dict does is let you use anything, not just numbers. Yes, a dict associates one thing to another, no matter what it is.  
   
**2.6 PYTHON FUNCTION: 2.6.1 Defining a Function:**​You can define functions to provide the required functionality. Here are simple rules to define a function in Python. Function blocks begin with the keyword def followed by the function name and parentheses (i.e.()).  
​Any input parameters or arguments should be placed within these parentheses. You can also define parameters inside these parentheses  
The code block within every function starts with a colon (:) and is indented. The statement returns [expression] exits a function, optionally passing back an expression to the caller. A return statement with no arguments is the same as return None.

**2.6.2 Calling a Function:**

​Defining a function only gives it a name, specifies the parameters that are to be included in the function and structures the blocks of code. Once the basic structure of a function is finalized, you can execute it by calling it from another function or directly from the Python prompt.

**2.7 PYTHON USING OOPs CONCEPTS:**

**2.7.1 Class:** ● Class: A user-defined prototype for an object that defines a set of attributes that characterize any object of the class. The attributes are data members (class variables and instance variables) and methods, accessed via dot notation.  
● Class variable: A variable that is shared by all instances of a class. Class variables are defined within a class but outside any of the class's methods. Class variables are not used as frequently as instance variables are. ● Data member: A class variable or instance variable that holds data associated with a class and its objects.  
● Instance variable: A variable that is defined inside a method and belongs only to the current instance of a class. ● Defining a Class:  
 1. We define a class in a very similar way how we define a function.  
 2. Just like a function ,we use parentheses and a colon after the class name(i.e. ():) when we define a class. Similarly, the body of our class is indented like a functions body is.

Figure 7 : Defining a Class

**CHAPTER 3: CASE STUDY**

**3.1 PROBLEM STATEMENT:**

Banks, telephone service companies, Internet service providers, pay TV companies, insurance firms, and alarm monitoring services, often use customer attrition analysis and customer attrition rates as one of their key business metrics (along with cash flow, EBITDA, etc.) because the cost of retaining an existing customer is far less than acquiring a new one (Wikipedia).

According to this article, the probability of selling to a new customer is 60-70%, while the probability of selling to a new prospect is 5-20%. So knowing if a customer is at risk of leaving is one of the most important tasks a company has to perform in order to keep growing its business.

In this Notebook we will analyse a dataset containing information about customers of a telephone company. The data can be found here on kaggle public datasets. We will predict if a customer will churn based on his informations.

**3.2 DATA SET:**

The given data set consists of the following parameters:

State

Account length

Area code

International plan

Voice mail plan

Number vmail messages

Total day minutes

Total day calls

Total day charge

Total eve minutes

Total eve calls

Total eve charge

Total night minutes

Total night calls

Total night charge

Total intl minutes

Total intl calls

Total intl charge

Customer service calls

Churn

**3.3 OBJECTIVE OF THE CASE STUDY:**

"Predict behavior to retain customers. You can analyze all relevant customer data and develop focused customer retention programs."

**CHAPTER 4 :MODEL BUILDING**

**4.1 PREPROCESSING OF THE DATA:**

Preprocessing of the data actually involves the following steps:

**4.1.1 GETTING THE DATASET:**

We can get the data set from the database or we can get the data from the client.

**4.1.2 IMPORTING THE LIBRARIES:**

We have to import the libraries as per the requirement of the algorithm.

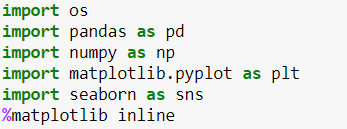


Figure 8 : Importing Libraries

**4.1.3 IMPORTING THE DATA-SET:**

​Pandas in python provide an interesting method read\_csv(). The read\_csv function reads the entire dataset from a comma separated values file and we can assign it to a DataFrame to which all the operations can be performed. It helps us to access each and every row as well as columns and each and every value can be accessed using the dataframe. Any missing value or NaN value has to be cleaned.

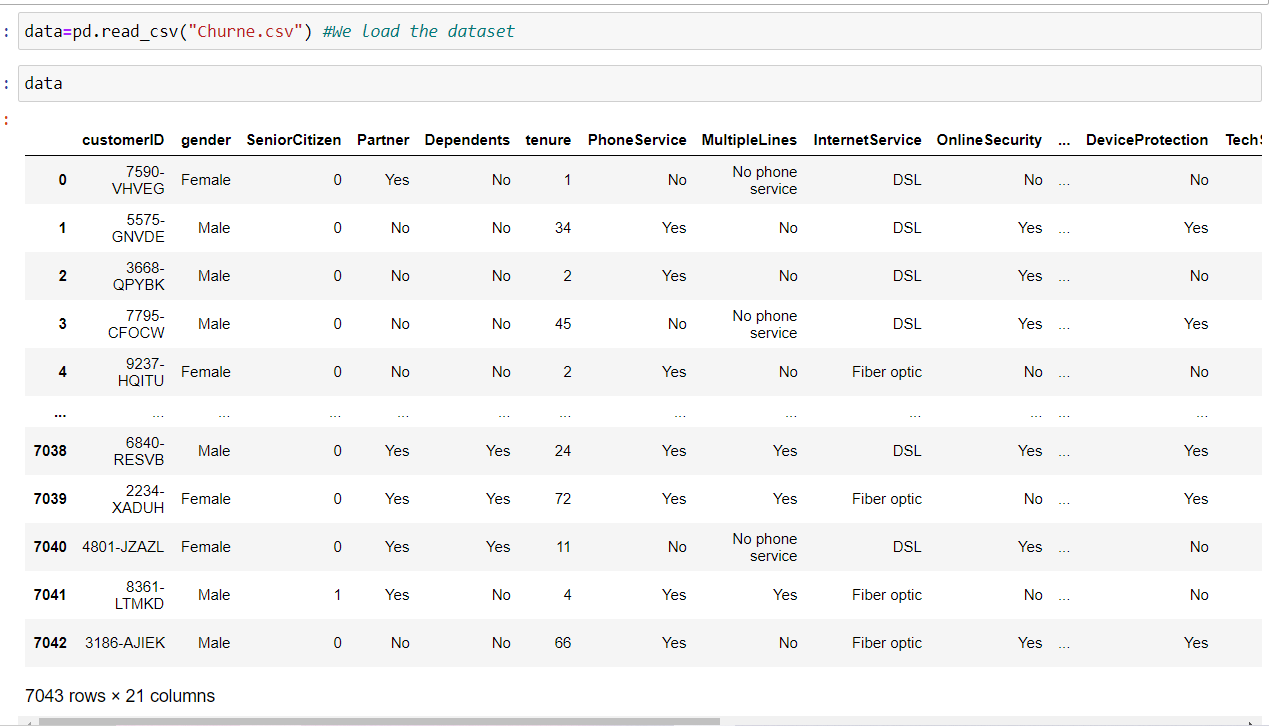


Figure 9 : Reading the dataset

**[[1]](#footnote-0)4.1.4 HANDLING MISSING VALUES:**

Missing values can be handled in many ways using some inbuilt methods:

1. dropna()
2. fillna()
3. interpolate()
4. mean imputation and median imputation

There are missing values in the TotalCharges column that are equal to ' ', that's why it's considered as an "object". Let's correct this



figure10:checking the missing values

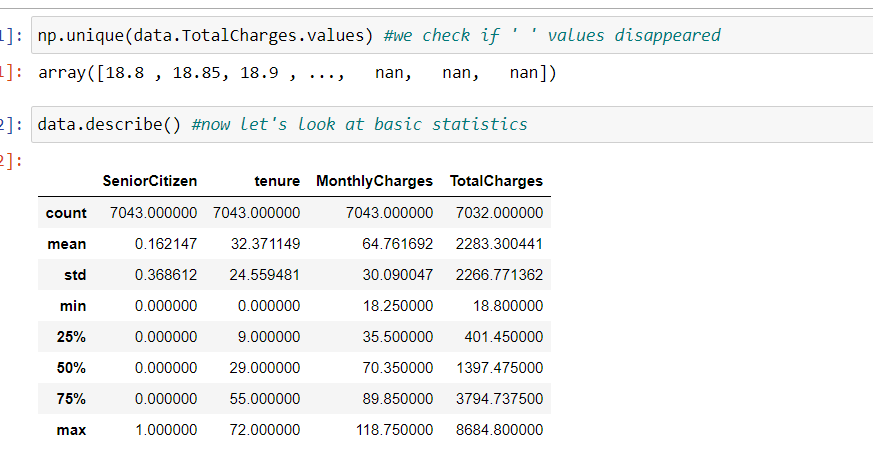
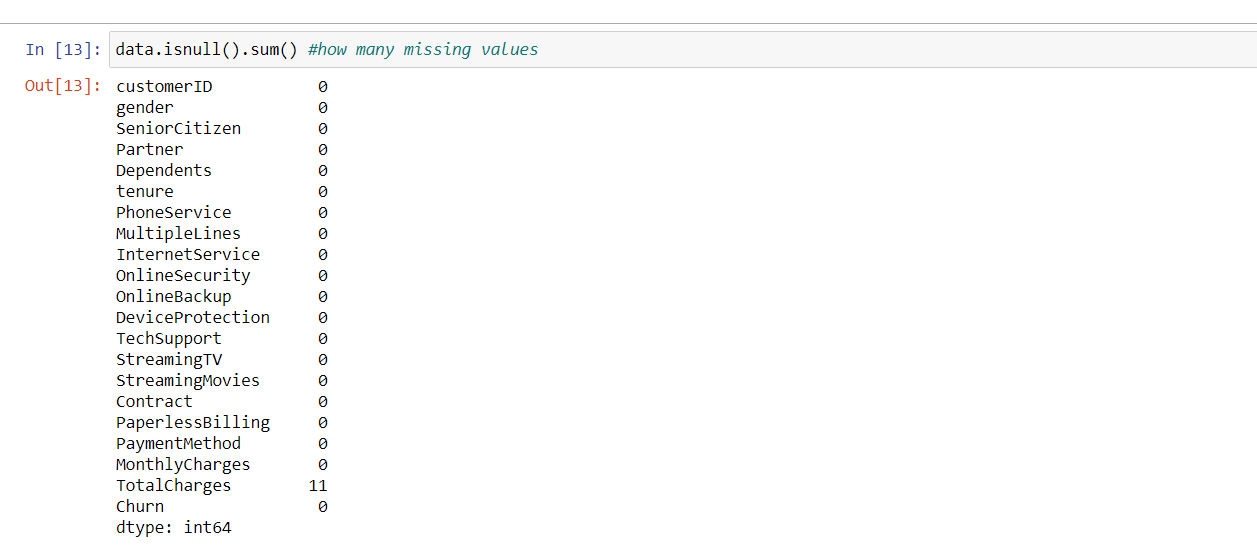


figure11:checking the missing values

This table show that there are 11 missing values in TotalCharges. On average people stay 1 month, but the dispersion of values is high with a standard deviation of 24.



**4.1.5 CATEGORICAL DATA:**

● Machine Learning models are based on equations, we need to replace the text by numbers. So that we can include the numbers in the equations.  
● Categorical Variables are of two types: Nominal and Ordinal  
● **Nominal**​: The categories do not have any numeric ordering in between them. They don't have any ordered relationship between each of them. Examples: Male or Female, any colour  
● **Ordinal**​: The categories have a numerical ordering in between them. Example: Graduate is less than Post Graduate, Post Graduate is less than Ph.D. customer satisfaction survey, high low medium

● Categorical data can be handled by using dummy variables, which are also called as indicator variables.

● Handling categorical data using dummies: ​In pandas library we have a method called get\_dummies() which creates dummy variables for those categorical data in the form of 0’s and 1’s.Once these dummies got created we have to concat this dummy set to our dataframe or we can add that dummy set to the dataframe.

**4.2. DATA PROCESSING :**

Let’s begin

Data as such can't be used. We'll transform the data so that we can feed it to a machine learning algorithm.

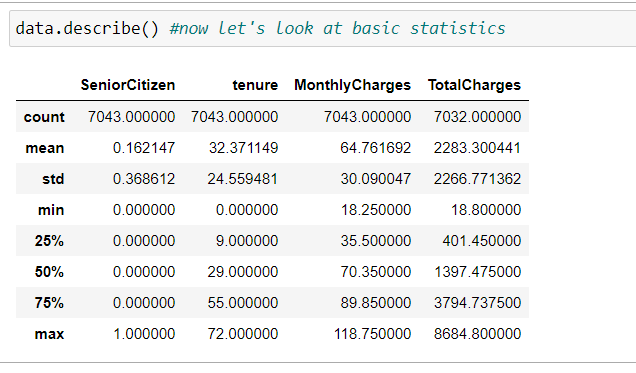


figure11:summary of the data frame

1.Now we will check the no of rows and columns in data frame

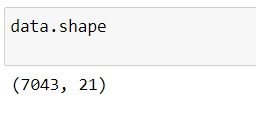


Figure12: checking the rows and columns

2. Checking the number of unique elements in the objects using nunique() function

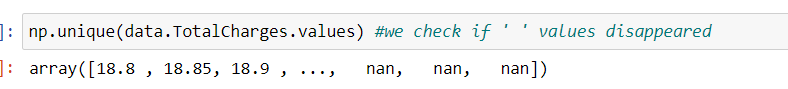


Figure15: checking the unique values

**4.3 EVALUATING THE CASE STUDY:**

So this model can be trained using the three different models i.e-

1.Logistic regression

2.KNN

3. Random forest classifier

**Logistic Regression:**

Logistic Regression is used when the dependent variable(target) is categorical.

Logistic Regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like all regression analyses, the logistic regression is a predictive analysis and it predicts the probability

Example: Yes or No, get a disease or not, pass or fail, defective or non-defective, etc.,

Also called a classification algorithm, because we are classifying the data. It predicts the probability associated with each dependent variable category.

**Z=b0+b1(x1)+b2(x2)+b3(x3)**

But, when we use the above equation to calculate probability, we would get values less than 0 as well as greater than 1. That doesn’t make any sense. So, we need to use such an equation which always gives values between 0 and 1, as we desire while calculating the probability.

**k-Nearest Neighbors(KNN):**

A supervised machine learning algorithm is one that relies on labeled input data to learn a function that produces an appropriate output when given new unlabeled data.

The k-nearest neighbors (KNN) algorithm is a simple, easy-to-implement supervised machine learning algorithm that can be used to solve both classification and regression problems. However, it is more widely used in classification problems in the industry.It is used for classification and regression of known data where usually the target variable is known beforehand.

K nearest neighbors is an algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions). It should also be noted that all three distance measures are only valid for continuous variables.

The 'k' stands for the number of nearest neighbors for the newly entered value.

The kNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other. kNN captures the idea of similarity (sometimes called distance, proximity, or closeness) with some mathematics we might have learned in our childhood— calculating the distance between points on a graph.

This works based on minimum distance from the query instance to the training samples to determine the k-nearest neighbors. After we gather these k-nearest neighbors, we take the simple majority of these k nearest neighbors to be the prediction of the query instance.

**Methods of calculating distance between points:**

The **first step** is to calculate the distance between the new point and each training point. There are various methods for calculating this distance, of which the most commonly known methods are – Euclidean, Manhattan and Hamming distance.

## **Euclidean Distance:** Euclidean distance is calculated as the square root of the sum of the squared differences between a new point (x) and an existing point (y).

## **Manhattan Distance/City Block Distance:** This is the distance between real vectors using the sum of their absolute difference.

## 

Figure35:distance functions

**Random forest classifier:**

Random forest classifier creates a set of decision trees from a randomly selected subset of training set. It then aggregates the votes from different decision trees to decide the final class of the test object.

In Layman's term,

Suppose training set is given as : [X1, X2, X3, X4] with corresponding labels as [L1, L2, L3, L4], random forest may create three decision trees taking input of subset for example,

1. [X1, X2, X3]
2. [X1, X2, X4]
3. [X2, X3, X4]

So finally, it predicts based on the majority of votes from each of the decision trees made.

● Importing the required libraries

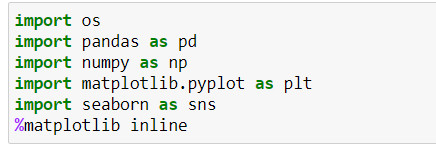
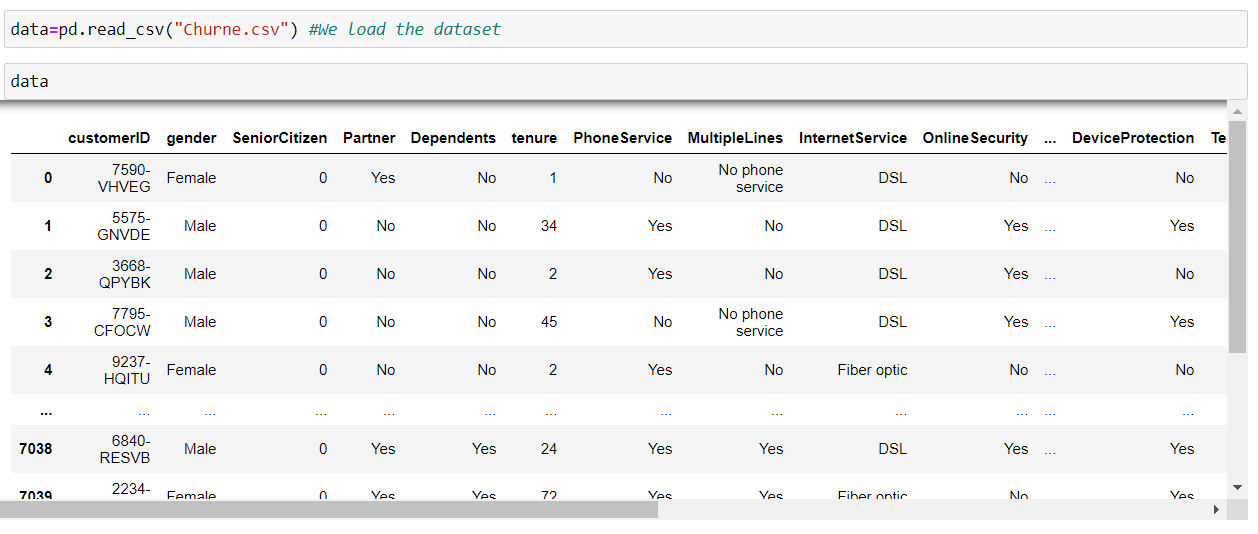


Figure36:importing the libraries

● Reading the Data-Set



● Handling the missing values

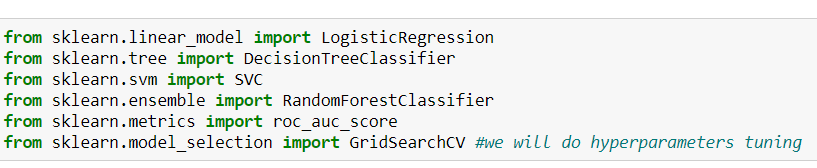
1. There is a method called isnull() which gives the number of missing values in each and every column.
2. Using fillna() method each and every missing value is replaced by 0.



**4.4 BUILDING THE MODEL:**

Here we build the model initially with three algorithms and then use best of them

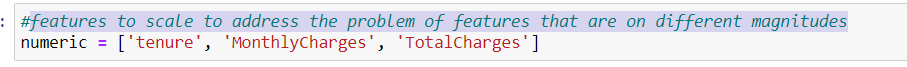
* importing the libraries



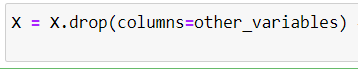
* we scale numeric features because they have different magnitudes which can impact the performance of our model

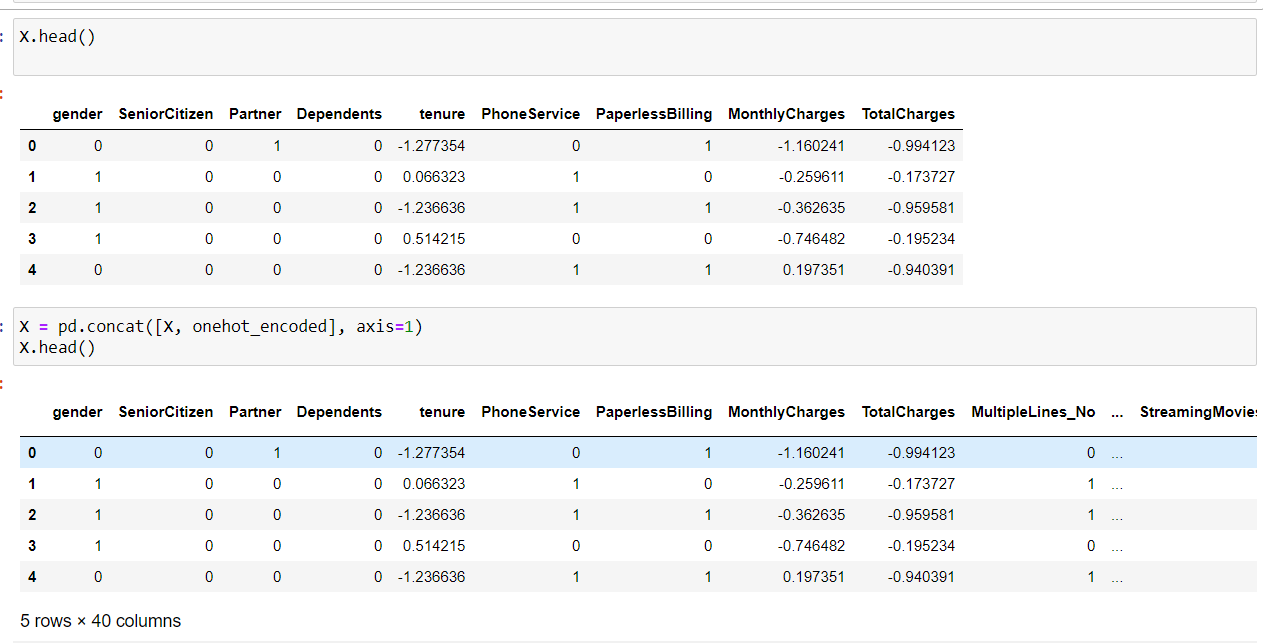


* features to scale to address the problem of features that are on different magnitudes

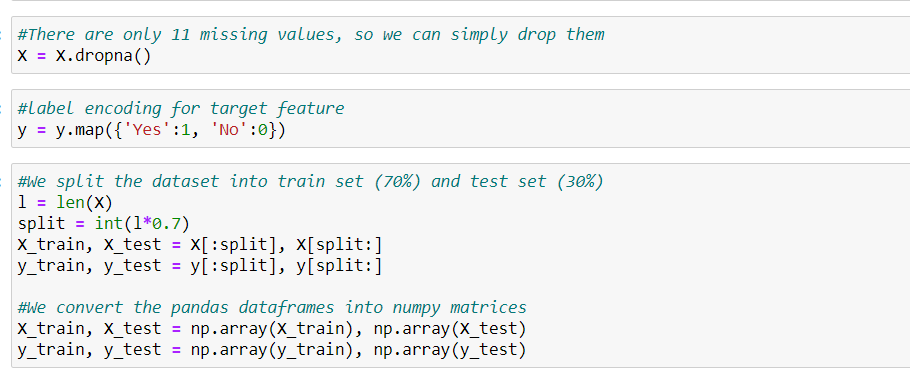


* we drop these variables, then concatenate the table with the one-hot encoded version





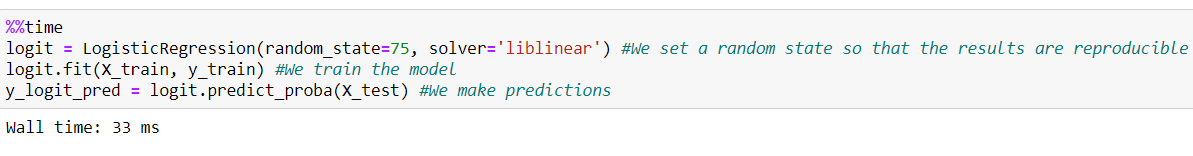
* After concatenation , we’ll check whether there’s any missing values or not. Then we’ll split the data . Import the train\_test\_split from model\_selection package from scikit learn library
* Then assigning the output to four different variables, before assigning we have to mention the train size or test size as a parameter to train\_test\_split. Then this method will split according to the size and assign it to four variables.



1. **Using logistic regression:**

Here we already imported logistic regression linear\_model package from scikit learn library

Then we build the model with it in the following way



* **Plotting the ROC-AUC curve:**

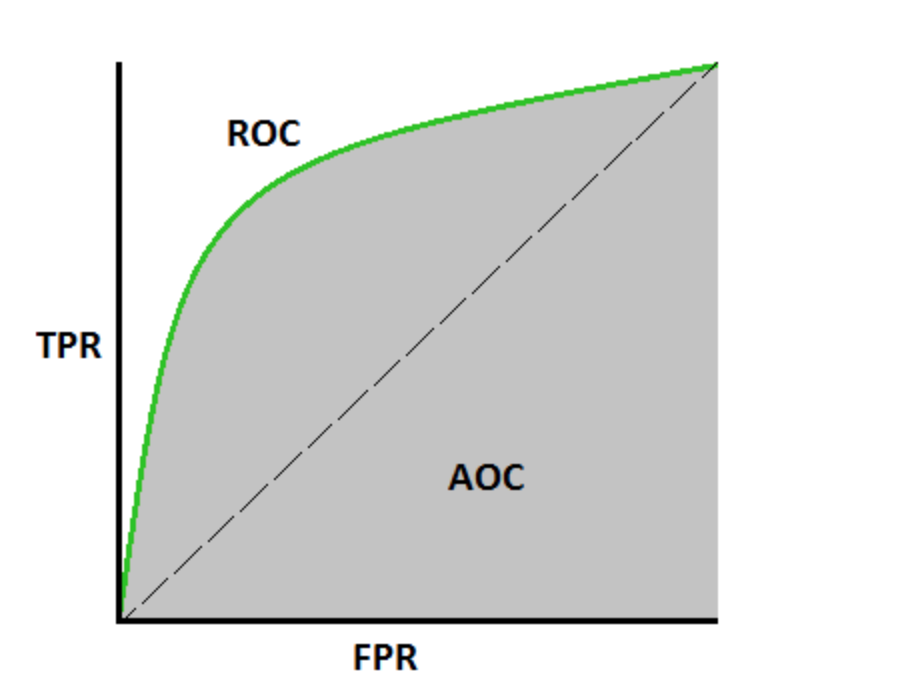
AUC - ROC curve is a performance measurement for classification problems at various thresholds settings. ROC is a probability curve and AUC represents degree or measure of separability. It tells how much a model is capable of distinguishing between classes.Higher the AUC, better the model is at predicting 0s as 0s and 1s as 1s. By analogy, Higher the AUC, better the model is at distinguishing between patients with disease and no disease.The ROC curve is plotted with TPR against the FPR where TPR is on the y-axis and FPR is on the x-axis.****

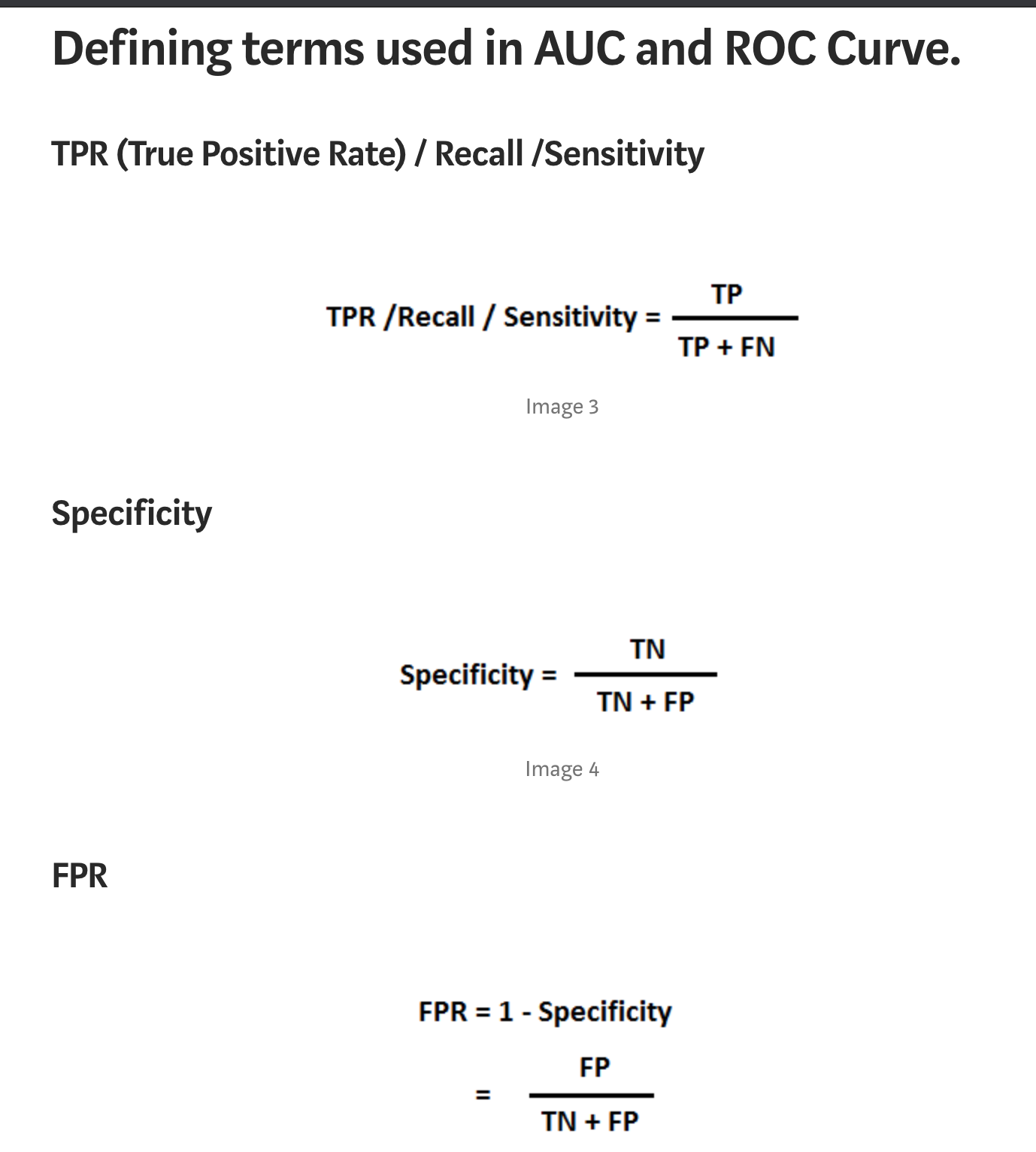
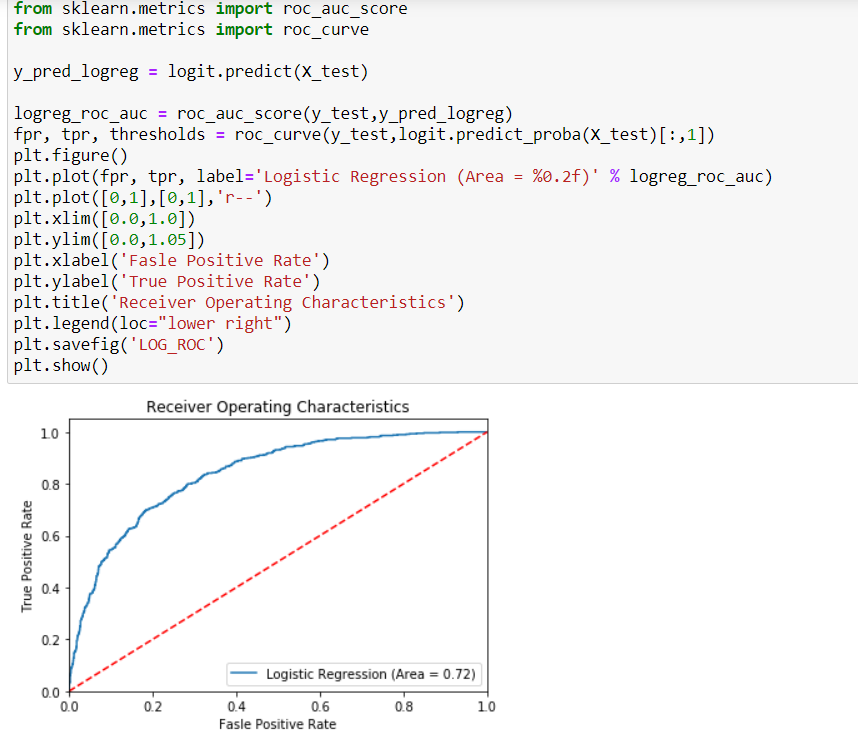
Figure 45: AUC - ROC Curve

Figure 46: defining the terms of AUC & ROC Curve



From the above plot we can observe that the area is equal to 0.72

* **Checking the metrics:**

We calculate the following metrics

1. Accuracy score
2. Precision Score
3. Recall Score
4. F1 Score
5. Confusion matrix

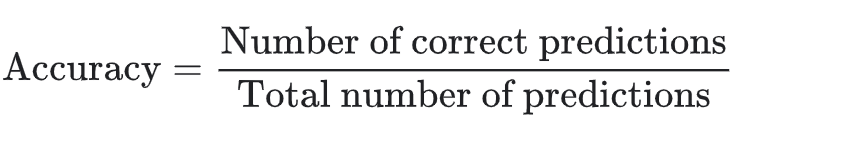
* **Accuracy score**:Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition:

Figure 48: accuracy formulae

For binary classification, accuracy can also be calculated in terms of positives and negatives as follows:

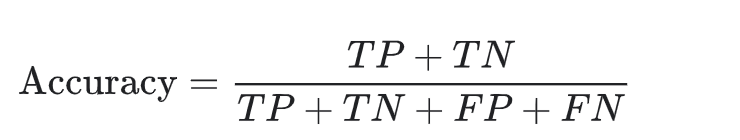


Figure 49: accuracy score in terms of positives and negatives

Where

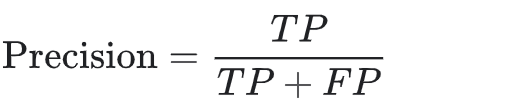
* *TP* = True Positives,
* *TN* = True Negatives,
* *FP* = False Positives, and
* *FN* = False Negatives.
* **Precision Score :**Precision, used in document retrievals, may be defined as the number of correct documents returned by our ML model. We can easily calculate it by confusion matrix with the help of following formula −

Figure 50: precision score in terms of positives and negatives

**Precision** attempts to answer the following question:

**What proportion of positive identifications was actually correct?**

* **Recall Score :**Recall may be defined as the number of positives returned by our ML model. We can easily calculate it by a confusion matrix with the help of the following formula.

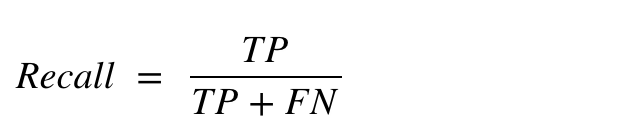


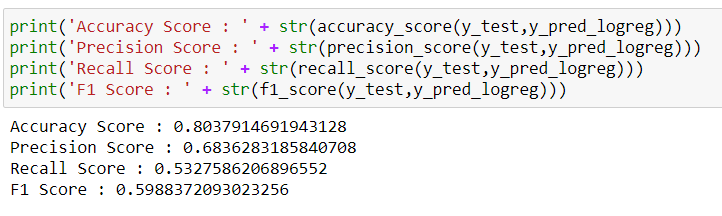
Figure 51: recall score in terms of positives and negatives

### **F1 Score:**This score will give us the harmonic mean of precision and recall. Mathematically, F1 score is the weighted average of the precision and recall. The best value of F1 would be 1 and worst would be 0. We can calculate F1 score with the help of following formula −

### F1=2∗(precision∗recall)/(precision+recall)

### F1 score is having equal relative contribution of precision and recall.

### We can use the classification\_report function of sklearn.metrics to get the classification report of our classification model.



* **Confusion matrix:**It is the easiest way to measure the performance of a classification problem where the output can be of two or more types of classes. A confusion matrix is nothing but a table with two dimensions viz. “Actual” and “Predicted” and furthermore, both the dimensions have “True Positives (TP)”, “True Negatives (TN)”, “False Positives (FP)”, “False Negatives (FN)” as shown below −

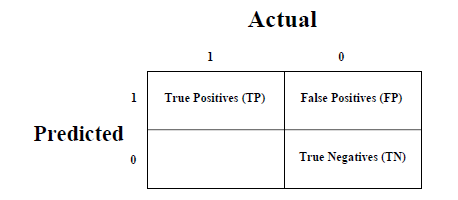
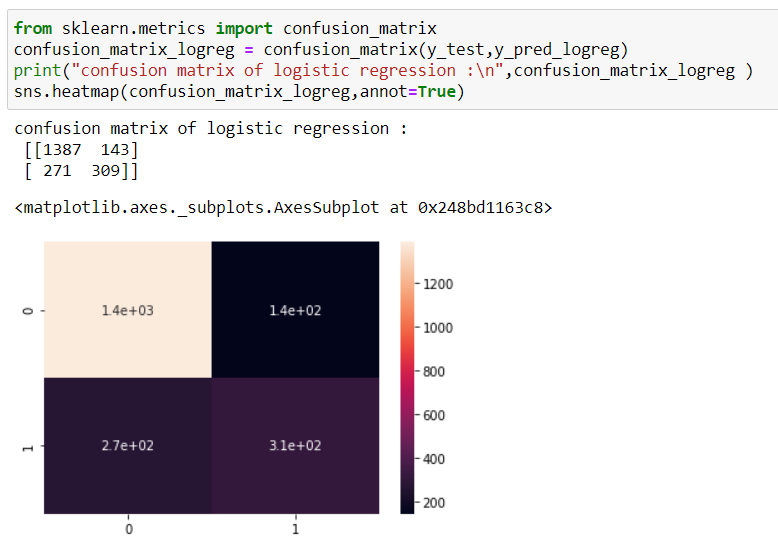


Figure 53: confusion matrix table

Explanation of the terms associated with confusion matrix are as follows −

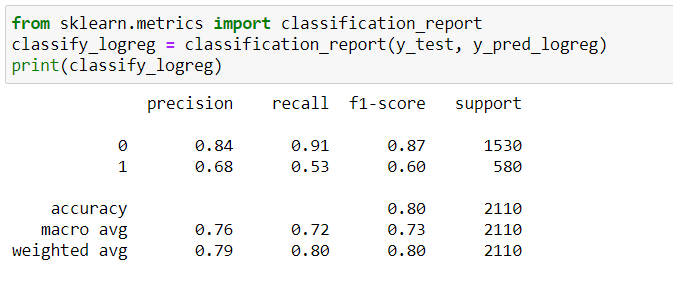
* True Positives (TP) − It is the case when both actual class & predicted class of data point is 1.
* True Negatives (TN) − It is the case when both actual class & predicted class of data point is 0.
* False Positives (FP) − It is the case when the actual class of data point is 0 & predicted class of data point is 1.
* False Negatives (FN) − It is the case when the actual class of data point is 1 & predicted class of data point is 0.

We can use the confusion\_matrix function of sklearn.metrics to compute Confusion Matrix of our classification model.



* **Classification report**: A Classification report is used to measure the quality of predictions from a classification algorithm. How many predictions are True and how many are False.

More specifically, True Positives, False Positives, True negatives and False Negatives are used to predict the metrics of a classification report as shown below.

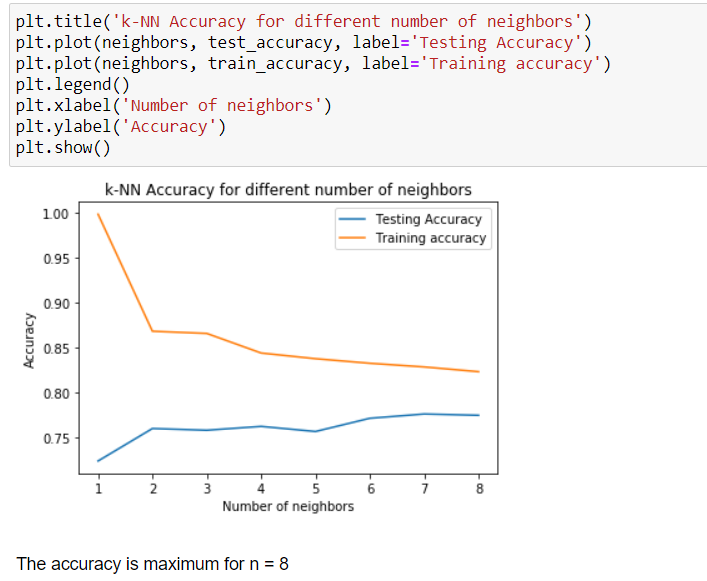


**2. Using KNN:**

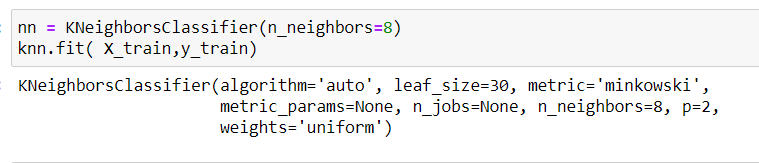
* accuracy for the testing and training accuracy



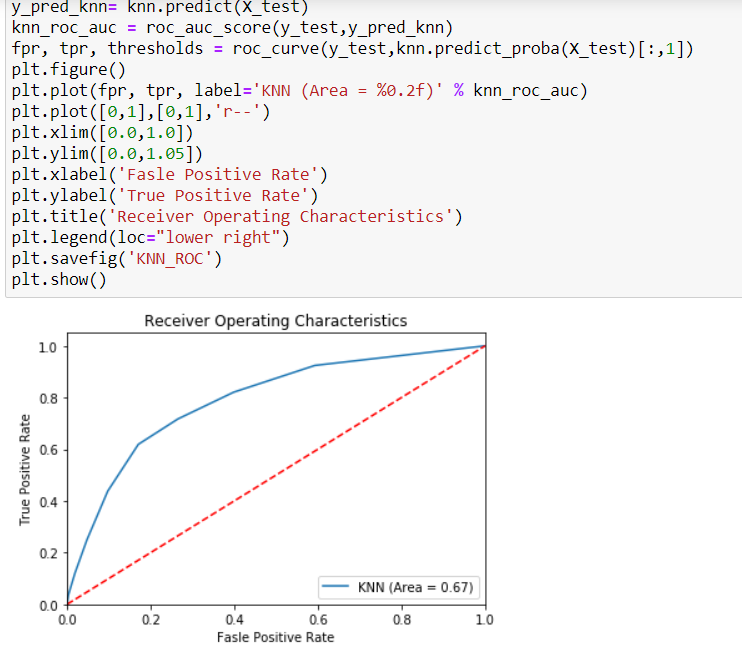
* **Plotting the testing and training accuracy:**

****From the above plot we understand that both test and train data accuracy is maximum at n=8

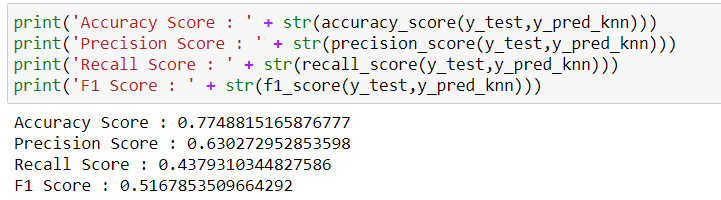
Let us fit the data using KNN classifier



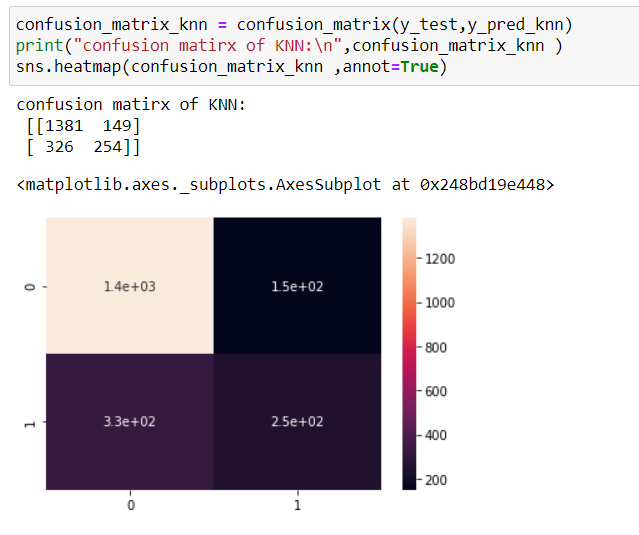
* **Plotting the ROC-AUC curve:**

****

* **Checking the metrics:**

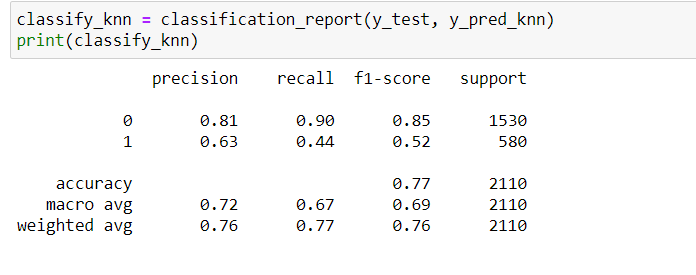
****

* confusion matrix



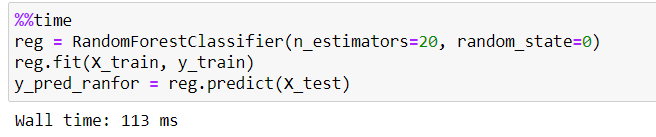
* **Classification report :**

Now let us check the overall classification report

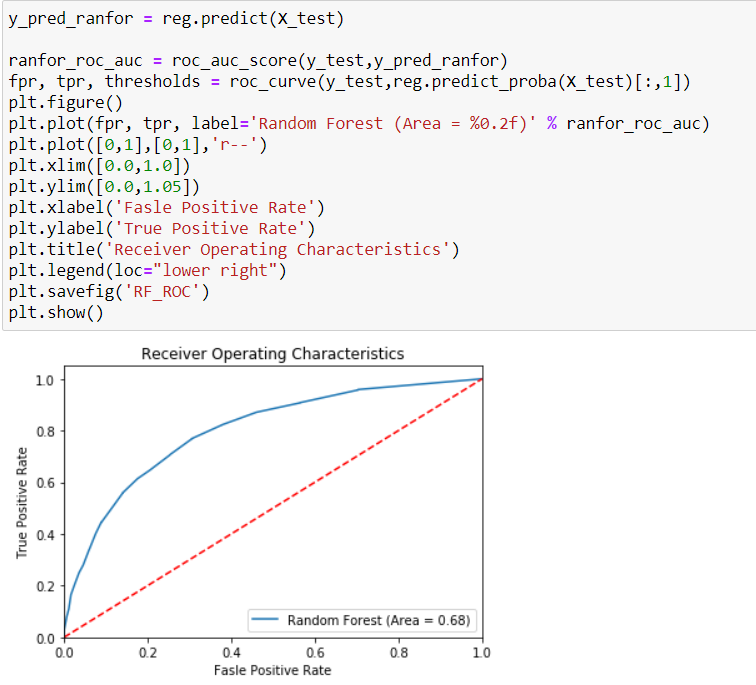


**3. Using the Random forest classifier:**

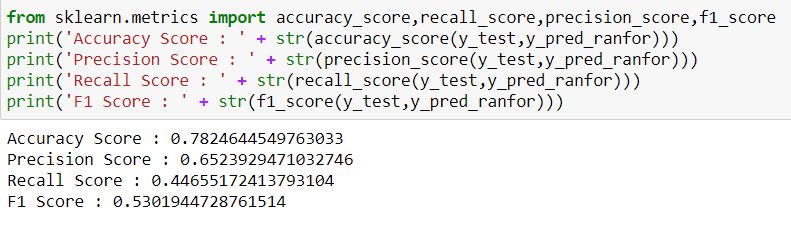
* fitting the data



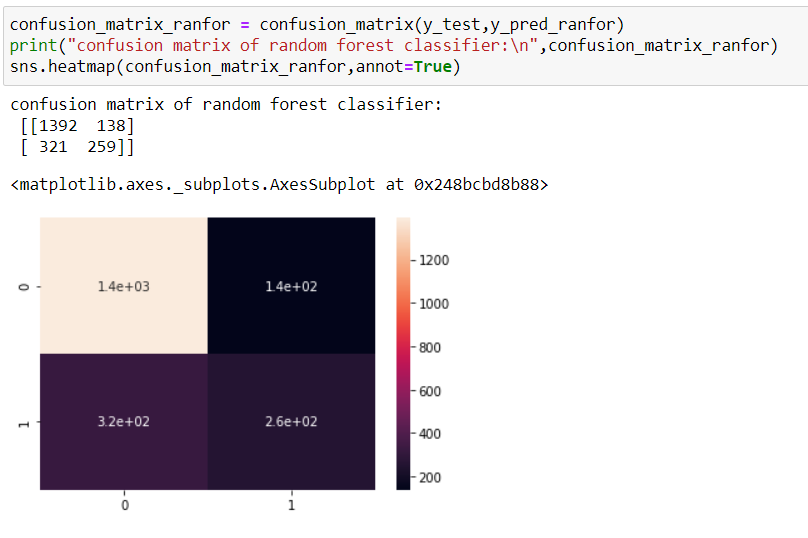
* **Plotting the ROC-AUC curve:**

****

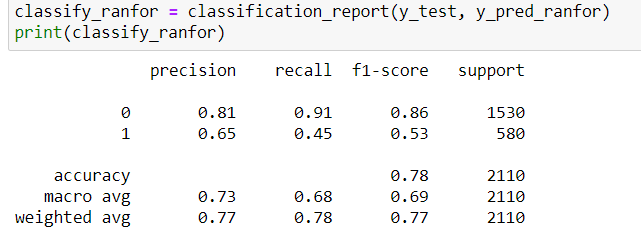
* **Checking the metrics:**

****

* confusion matrix of random forest classifier

****

* **Classification report:**

****

**Hyperparameter tuning using GridSearchCV:**

**4.5.1 What is Hyperparameter tuning ?**

* A hyperparameter is a parameter whose value is set before the learning process begins.
* Hyperparameter tuning is also tricky in the sense that there is no direct way to calculate how a change in the hyperparameter value will reduce the loss of your model, so we usually resort to experimentation.
* This starts with us specifying a range of possible values for all the hyperparameters.
* Now, this is where most get stuck, what values you are going to try, and to answer that question, you first need to understand what these hyperparameters mean and how changing a hyperparameter will affect your model architecture, thereby try to understand how your model performance might change.
* The next step after you define the range of values is to use a hyperparameter tuning method, there’s a bunch, the most common and expensive being Grid Search .

**4.5.2 What is GridSearchCV?**

* Grid search is a traditional way to perform hyperparameter optimization. It works by searching exhaustively through a specified subset of hyperparameters.
* Grid search is the process of performing hyper parameter tuning in order to determine the optimal values for a given model.
* This is significant as the performance of the entire model is based on the hyper parameter values specified.
* Using sklearn’s GridSearchCV, we first define our grid of parameters to search over and then run the grid search.

**Among all three algorithms logistic has more accuracy. So,lets optimize it by using hyperparameter optimization.**

****

**CONCLUSION:**

It is concluded after performing thorough Exploratory Data analysis which include Logistic Regression which is computed to get the accuracy and also Heat maps which are computed to get a clear understanding of the data set (which parameter has most abundant effect on the study case).

1. [↑](#footnote-ref-0)