

Telecom Customer Churn Prediction

*Project Report submitted in the partial fulfilment of the
requirements for the award of the degree*

BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE AND ENGINEERING

Submitted by

V. Sahendra Reddy	(19471A05P1)
V. Ankamma Rao	(20475A0502)
M. Chandrasekhara rao	(19471A05N0)

Under the esteemed guidance of
A. Thanuja, M.Tech.,(Ph.D)
Asst.Professor



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
NARASARAOPETA ENGINEERING COLLEGE: NARASARAOPET
(AUTONOMOUS)**

Accredited by NAAC with A+ Grade and NBA under (Tier -1)
NIRF rank in the band of 251-320 and an ISO 9001:2015 Certified
Approved by AICTE, New Delhi, Permanently Affiliated to JNTUK, Kakinada
KOTAPPAKONDA ROAD, YALAMANDA VILLAGE, NARASARAOPET-522601
2022-2023

NARASARAOPETA ENGINEERING COLLEGE: NARASARAOPET

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE



This is to certify that the project entitled “**Telecom Customer Churn Prediction**” is a bonafide work done by “**V. Sahendra Reddy (19471A05P1), V. Ankamma Rao (20475A0502), M. Chandrasekhara rao(19471A05N0)**” in partial fulfilment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY** in the Department of **COMPUTER SCIENCE AND ENGINEERING** during 2022-2023.

PROJECT GUIDE

A. Thanuja, M.Tech.,(Ph.D)
Asst.Professor

PROJECT CO-ORDINATOR

Dr. M. Sireesha, M.Tech.,Ph.D.
Assoc.Professor

HEAD OF THE DEPARTMENT

Dr. S. N. Tirumala Rao, M.Tech., Ph.D.
Professor

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

We wish to express my thanks to carious personalities who are responsible for the completion of the project. We are extremely thankful to our beloved chairman sri **M.V.Koteswara Rao, B.Sc** who took keen interest in us in every effort throughout this course. We owe out sincere gratitude to our beloved principal **Dr.M.Sreenivasa Kumar, M.Tech, Ph.D.(UK), MISTE., FIE(I)** for showing his kind attention and valuable guidance throughout the course.

We express our deep felt gratitude towards **Dr.S.N.Tirumala Rao, M.Tech.,Ph.D** HOD of CSE department and also to our guide **A. Thanuja, M.Tech.,(Ph.D)** Assistant Professor of CSE department whose valuable guidance and unstinting encouragement enable us to accomplish our project successfully in time.

We extend our sincere thanks towards **Dr. M. Sireesha, M.Tech., Ph.D** Associate professor & Project coordinator of the project for extending her encouragement. Their profound knowledge and willingness have been a constant source of inspiration for us throughout this project work.

We extend our sincere thanks to all other teaching and non-teaching staff of department for their cooperation and encouragement during our B.Tech degree.

We have no words to acknowledge the warm affection, constant inspiration and encouragement that we received from our parents.

We affectionately acknowledge the encouragement received from our friends and those who involved in giving valuable suggestions had clarifying out doubts which had really helped us in successfully completing our project.

By

V. Sahendra Reddy (19471A05P1)

V. Ankamma Rao(20475A0502)

ABSTRACT

Customer churn analysis and prediction in telecom sector is an issue now a days because it's very important for telecommunication industries to analyze behaviors of various customer to predict which customers are about to leave the subscription from telecom company. So data mining techniques and algorithm plays an important role for companies in today's commercial conditions because gaining a new customer's cost is more than retaining the existing ones. In this paper we can focuses on various machine learning techniques for predicting customer churn through which we can build the classification models such as Logistic Regression, SVM, Random Forest and Gradient boosted tree and also compare the performance of these models.



INSTITUTE VISION AND MISSION

INSTITUTION VISION

To emerge as a Centre of excellence in technical education with a blend of effective student centric teaching learning practices as well as research for the transformation of lives and community

INSTITUTION MISSION

M1: Provide the best class infra-structure to explore the field of engineering and research

M2: Build a passionate and a determined team of faculty with student centric teaching, imbibing experiential, innovative skills

M3: Imbibe lifelong learning skills, entrepreneurial skills and ethical values in students for addressing societal problems



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VISION OF THE DEPARTMENT

To become a centre of excellence in nurturing the quality Computer Science & Engineering professionals embedded with software knowledge, aptitude for research and ethical values to cater to the needs of industry and society.

MISSION OF THE DEPARTMENT

The department of Computer Science and Engineering is committed to

M1: Mould the students to become Software Professionals, Researchers and Entrepreneurs by providing advanced laboratories.

M2: Impart high quality professional training to get expertize in modern software tools and technologies to cater to the real time requirements of the Industry.

M3: Inculcate team work and lifelong learning among students with a sense of societal and ethical responsibilities.



Program Specific Outcomes (PSO's)

PSO1: Apply mathematical and scientific skills in numerous areas of Computer Science and Engineering to design and develop software-based systems.

PSO2: Acquaint module knowledge on emerging trends of the modern era in Computer Science and Engineering

PSO3: Promote novel applications that meet the needs of entrepreneur, environmental and social issues.



Program Educational Objectives (PEO's)

The graduates of the programme are able to:

PEO1: Apply the knowledge of Mathematics, Science and Engineering fundamentals to identify and solve Computer Science and Engineering problems.

PEO2: Use various software tools and technologies to solve problems related to academia, industry and society.

PEO3: Work with ethical and moral values in the multi-disciplinary teams and can communicate effectively among team members with continuous learning.

PEO4: Pursue higher studies and develop their career in software industry.

Program Outcomes

1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Project Course Outcomes (CO'S):

CO425.1: Analyse the System of Examinations and identify the problem.

CO425.2: Identify and classify the requirements.

CO425.3: Review the Related Literature

CO425.4: Design and Modularize the project

CO425.5: Construct, Integrate, Test and Implement the Project.

CO425.6: Prepare the project Documentation and present the Report using appropriate method.

Course Outcomes – Program Outcomes mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C425.1		✓											✓		
C425.2	✓		✓		✓								✓		
C425.3				✓		✓	✓	✓					✓		
C425.4			✓			✓	✓	✓					✓	✓	
C425.5					✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C425.6									✓	✓	✓		✓	✓	

Course Outcomes – Program Outcome correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C425.1	2	3											2		
C425.2			2		3								2		
C425.3				2		2	3	3					2		
C425.4			2			1	1	2					3	2	
C425.5					3	3	3	2	3	2	2	1	3	2	1
C425.6									3	2	1		2	3	

Note: The values in the above table represent the level of correlation between CO's and PO's:

1. Low level

2. Medium level

3. High level

Project mapping with various courses of Curriculum with Attained PO's:

Name of the course from which principles are applied in this project	Description of the device	Attained PO
C3.2.4, C3.2.5	Gathering the requirements and defining the problem, plan to develop a software for Predicting Customer Churn Prediction in Telecom Sector	PO1, PO3
CC4.2.5	Each and every requirement is critically analyzed, the process model is identified and divided into different modules	PO2, PO3
CC4.2.5	Logical design is done by using the unified modelling language which involves individual team work	PO3, PO5, PO9
CC4.2.5	Each and every module is tested, integrated, and evaluated in our project	PO1, PO5
CC4.2.5	Documentation is done by all our three members in the form of a group	PO10
CC4.2.5	Each and every phase of the work in group is presented periodically	PO10, PO11
CC4.2.5	Implementation is done and the project can be handled easily and in future updates in our project can be done by applying it in various industries	PO4, PO7
CC4.2.8 CC4.2.	The design includes software components like model and python application.	PO5, PO6

INDEX

S.NO	CONTENTS	PG.NO
	List of figures	XVI
1	Introduction	1
	1.1 Introduction	1
	1.2 Existing system	1
	1.3 Proposed system	2
	1.4 System requirements	2
	1.4.1 Hardware requirements	2
	1.4.2 Software requirements	3
2	Literature survey	4
	2.1 Machine learning	4
	2.2 Some Machine learning methods	6
	2.3 Applications of Machine learning	7
3	System analysis	8
	3.1 Implementation	8
	3.2 Algorithms	8
	3.3 Code	9
4	Output screens	13

5	Conclusion	23
6	Future scope	24
7	Bibliography	25

LIST OF FIGURES

S.NO	LIST OF FIGURES	PAGE NO
1	Fig.4.1 import packages	13
2	Fig.4.2 Read dataset	13
3	Fig.4.3 Display dataset 1	14
4	Fig.4.4 Display dataset 2	14
5	Fig.4.5 dataset features correlation graph	15
6	Fig 4.6 Splitting dataset	15
7	Fig.4.7 AUC	16
8	Fig 4.8 Train Logistic regression	16
9	Fig 4.9 Train SVM	17
10	Fig 4.10 Train Random forest	17
11	Fig 4.11 Train Gradient boosting	18
12	Fig.4.12 Algorithms performance table	18
13	Fig.4.13 Prediction 1	19
14	Fig.4.14 Prediction 2	19
15	Fig 4.15 Churn Prediction Test Data	20
16	Fig.4.16 Prediction Screen	20
17	Fig.4.17 Prediction result 1	21

18	Fig.4.18 Prediction result 2	21
19	Fig.4.19 Prediction	22

1. INTRODUCTION

1.1 Introduction

In today's technological conditions, new data are being produced by different sources in many sectors. However, it is not possible to extract the useful information hidden in these data sets, unless they are processed properly. In order to find out these hidden information, various analyses should be performed using data mining, which consists of numerous methods.[6] The Churn Analysis [4] aims to predict customers who are going to stop using a product or service among the customers. And, the customer churn analysis is a data mining based work that will extract these possibilities. Today's competitive conditions led to numerous companies selling the same product at quite a similar service and product quality. With the Churn Analysis[7], it is possible to precisely predict the customers who are going to stop using services or products by assigning a probability to each customer. This analysis can be performed according to customer segments and amount of loss (monetary equivalent). Following these analyses, communication with the customers can be improved in order to persuade the customers and increase customer loyalty. Effective marketing campaigns for target customers can be created by calculating the churn rate or customer attrition. In this way, profitability can be increased significantly or the possible damage due to customer loss can be reduced at the same rate . For example, if a service provider which has a total of 2 million subscribers, gains 750.000 new subscribers and lost 275.000 customers; churn rate is calculated as 10%. The customer churn rate has a significant effect on the financial market value of the company. So most of the companies keep an eye on the value of the customer at monthly or quarterly periods.

1.2 Existing system

From the problems obligatory through market saturation and value implications, there has been associate identification of a desire for a laptop based mostly churn prediction methodology that's capable of accurately distinctive a loss of client ahead, so proactive retention ways is deployed during a bid to retain the client. The churn prediction should be correct as a result of retention ways is pricey. A limitation of current analysis is that

alternative studies have focused virtually solely on churn capture, neglecting the problem of misclassification of non-churn as churn. Retention campaigns usually embrace creating service based mostly offers to customers during a bid to retain them. These offers is pricey, thus providing them to customers World Health Organization don't shall churn will have a substantial impact on the whole price of a retention strategy. an extra limitation of current analysis is that it's typically supported one output within the kind of zero for non-churn and one for churn. This has been recognized as a limitation as a result of it restricts analysis prospects.

Disadvantages:

1. In telecom industry if customer not satisfy with services then he will leave the subscription which may cause huge loss to Telecom Company

1.3 Proposed system

In this paper, we proposed different machine learning algorithms to analyze customer churn analysis. Through which we can multiple different models are employed to accurately predict those churn customers in the data set. These models are Logistic Regression, Support Vector Machine, Random Forest, Gradient Boosting Trees.

Advantages:

1. By using this algorithms company can know about customer behaviour and based on behaviour they may improve telecom service performance

1.4 System requirements

1.4.1 Hardware requirements

System	:	intel®core™i7-7500UCPU@2.70gh
Cache memory	:	4MB
RAM	:	12GB
Hard disc	:	8GB

1.4.2 Software requirements

Operating system	:	Windows 10, 64 bit
Coding language	:	python
Python distribution	:	jupyter, flask

2. LITERATURE SURVEY

2.1 Machine learning

Before we take a look at the details of various machine learning methods, let's start by looking at what machine learning is, and what it isn't. Machine learning is often categorized as a subfield of artificial intelligence, but I find that categorization can often be misleading at first brush. The study of machine learning certainly arose from research in this context, but in the data science application of machine learning methods, it's more helpful to think of machine learning as a means of building models of data.

Fundamentally, machine learning involves building mathematical models to help understand data. "Learning" enters the fray when we give these models *tunable parameters* that can be adapted to observed data; in this way the program can be considered to be "learning" from the data. Once these models have been fit to previously seen data, they can be used to predict and understand aspects of newly observed data. I'll leave to the reader the more philosophical digression regarding the extent to which this type of mathematical, model-based "learning" is similar to the "learning" exhibited by the human brain. Understanding the problem setting in machine learning is essential to using these tools effectively, and so we will start with some broad categorizations of the types of approaches we'll discuss here.

Categories Of Machine Learning :-

At the most fundamental level, machine learning can be categorized into two main types: supervised learning and unsupervised learning.

Supervised learning involves somehow modeling the relationship between measured features of data and some label associated with the data; once this model is determined, it can be used to apply labels to new, unknown data. This is further subdivided into *classification* tasks and *regression* tasks: in classification, the labels are discrete categories, while in regression,

the labels are continuous quantities. We will see examples of both types of supervised learning in the following section.

Unsupervised learning involves modeling the features of a dataset without reference to any label, and is often described as "letting the dataset speak for itself." These models include tasks such as *clustering* and *dimensionality reduction*. Clustering algorithms identify distinct groups of data, while dimensionality reduction algorithms search for more succinct representations of the data. We will see examples of both types of unsupervised learning in the following section.

Need for Machine Learning

Human beings, at this moment, are the most intelligent and advanced species on earth because they can think, evaluate and solve complex problems. On the other side, AI is still in its initial stage and haven't surpassed human intelligence in many aspects. Then the question is that what is the need to make machine learn? The most suitable reason for doing this is, "to make decisions, based on data, with efficiency and scale".

Lately, organizations are investing heavily in newer technologies like Artificial Intelligence, Machine Learning and Deep Learning to get the key information from data to perform several real-world tasks and solve problems. We can call it data-driven decisions taken by machines, particularly to automate the process. These data-driven decisions can be used, instead of using programming logic, in the problems that cannot be programmed inherently. The fact is that we can't do without human intelligence, but other aspect is that we all need to solve real-world problems with efficiency at a huge scale. That is why the need for machine learning arises.

Challenges in Machines Learning :-

While Machine Learning is rapidly evolving, making significant strides with cybersecurity and autonomous cars, this segment of AI as whole still has a long way to go. The reason behind is that ML has not been able to overcome number of challenges. The challenges that ML is facing currently are –

Quality of data Having good-quality data for ML algorithms is one of the biggest challenges. Use of low-quality data leads to the problems related to data preprocessing and feature extraction.

Time-Consuming task – Another challenge faced by ML models is the consumption of time especially for data acquisition, feature extraction and retrieval.

Lack of specialist persons – As ML technology is still in its infancy stage, availability of expert resources is a tough job.

No clear objective for formulating business problems – Having no clear objective and well-defined goal for business problems is another key challenge for ML because this technology is not that mature yet.

Issue of overfitting & underfitting – If the model is overfitting or underfitting, it cannot be represented well for the problem.

Curse of dimensionality – Another challenge ML model faces is too many features of data points. This can be a real hindrance.

Difficulty in deployment – Complexity of the ML model makes it quite difficult to be deployed in real life.

2.2 Some Machine learning methods

- **Supervised Learning** – This involves learning from a training dataset with labeled data using classification and regression models. This learning process continues until the required level of performance is achieved.
- **Unsupervised Learning** – This involves using unlabelled data and then finding the underlying structure in the data in order to learn more and more about the data itself using factor and cluster analysis models.
- **Semi-supervised Learning** – This involves using unlabelled data like Unsupervised Learning with a small amount of labeled data. Using labeled data vastly increases the learning accuracy and is also more cost-effective than Supervised Learning.

- **Reinforcement Learning** – This involves learning optimal actions through trial and error. So the next action is decided by learning behaviors that are based on the current state and that will maximize the reward in the future.

2.3 Applications of Machine learning

Machine Learning is the most rapidly growing technology and according to researchers we are in the golden year of AI and ML. It is used to solve many real-world complex problems which cannot be solved with traditional approach. Following are some real-world applications of ML –

- Emotion analysis
- Sentiment analysis
- Error detection and prevention
- Weather forecasting and prediction
- Stock market analysis and forecasting
- Speech synthesis
- Speech recognition
- Customer segmentation
- Object recognition
- Fraud detection
- Fraud prevention
- Recommendation of products to customer in online shopping

3. SYSTEM ANALYSIS

3.1 Implementation

In telecom industry if customer not satisfy with services then he will leave the subscription which may cause huge loss to Telecom Company so Telecom Company always in search of customer behaviour to know whether customer will stay or leave subscription. In propose paper we are employing and evaluating performance of various machine learning algorithms such as Logistic Regression, SVM, Random Forest and Gradient Boosting to predict customer churn. By using this algorithms company can know about customer behaviour and based on behaviour they may improve telecom service performance.

To train all algorithms we have used same dataset given by you and then trained and compare all algorithms performance in terms of accuracy and AUC and in all algorithms Random Forest is giving best performance.

We have coded training part with JUPYTER notebook and implemented prediction part with FLASK web interface where user will enter churn data and then algorithm will predict user behaviour of churn as Yes or No where Yes means customer will leave subscription and NO means will stay.

3.2 Algorithms

Logistic Regression:

SVM: Support Vector Machine(SVM) is a supervised machine learning algorithm used for both classification and regression. Though we say regression problems as well its best suited for classification. The objective of SVM algorithm is to find a hyperplane in an N-dimensional space that distinctly classifies the data points. SVM works relatively well when there is a clear margin of separation between classes. SVM is more effective in high dimensional spaces and is relatively memory efficient. SVM is effective in cases where the dimensions are greater than the number of samples.

Random Forest:

Random forest is a commonly-used machine learning algorithm trademarked by Leo Breiman and Adele Cutler, which combines the output of multiple decision trees to reach a single result. Its ease of use and flexibility have fueled its adoption, as it handles both classification and regression problems. Random forest is used on the job by data scientists in many industries including banking, stock trading, medicine, and e-commerce. It's used to predict the things which help these industries run efficiently, such as customer activity, patient history, and safety.

Gradient Boosting:

Gradient boosting is a machine learning technique used in regression and classification tasks, among others. It gives a prediction model in the form of an ensemble of weak prediction models, which are typically decision trees. In Machine Learning, we use gradient boosting to solve classification and regression problems. It is a sequential ensemble learning technique where the performance of the model improves over iterations. This method creates the model in a stage-wise fashion.

3.3 Code

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import plotly.graph_objects as go
from plotly.subplots import make_subplots
import warnings
warnings.filterwarnings('ignore')

from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import StandardScaler

from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier

from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import LabelEncoder
```

```

from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.neural_network import MLPClassifier
from sklearn.ensemble import AdaBoostClassifier
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.ensemble import ExtraTreesClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from xgboost import XGBClassifier
from catboost import CatBoostClassifier
from sklearn import metrics
from sklearn.metrics import roc_curve
from sklearn.metrics import recall_score, confusion_matrix, precision_score, f1_score,
accuracy_score, classification_report

```

```

from sklearn.ensemble import VotingClassifier
from sklearn.metrics import confusion_matrix, accuracy_score
from sklearn.metrics import f1_score, precision_score, recall_score, fbeta_score
from statsmodels.stats.outliers_influence import variance_inflation_factor
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import ShuffleSplit
from sklearn.model_selection import KFold
from sklearn import feature_selection
from sklearn import model_selection
from sklearn import metrics
from sklearn.metrics import classification_report, precision_recall_curve
from sklearn.metrics import auc, roc_auc_score, roc_curve
from sklearn.metrics import make_scorer, recall_score, log_loss
from sklearn.metrics import average_precision_score
#Standard libraries for data visualization:
data = pd.read_csv("data.csv")
data.head()
import missingno as msno
msno.matrix(data)
data = data.drop(["customerID"], axis = 1)
data.head()
data['TotalCharges'] = pd.to_numeric(data.TotalCharges, errors='coerce')
data.isnull().sum()
data['TotalCharges'] = pd.to_numeric(data.TotalCharges, errors='coerce')
data.isnull().sum()

```

```

type_ = ["No", "yes"]
fig = make_subplots(rows=1, cols=1)

fig.add_trace(go.Pie(labels=type_, values=data['Churn'].value_counts(), name="Churn"))

# Use `hole` to create a donut-like pie chart
fig.update_traces(hole=.4, hoverinfo="label+percent+name", textfont_size=16)

fig.update_layout(
    title_text="Churn Distributions",
    # Add annotations in the center of the donut pies.
    annotations=[dict(text='Churn', x=0.5, y=0.5, font_size=20, showarrow=False)])
fig.show()
plt.figure(figsize=(6, 6))
labels = ["Churn: Yes", "Churn:No"]
values = [1869, 5163]
labels_gender = ["F", "M", "F", "M"]
sizes_gender = [939, 930, 2544, 2619]
colors = ['#ff6666', '#66b3ff']
colors_gender = ['#c2c2f0', '#ffb3e6', '#c2c2f0', '#ffb3e6']
explode = (0.3, 0.3)
explode_gender = (0.1, 0.1, 0.1, 0.1)
textprops = {"fontsize": 15}
#Plot
plt.pie(values, labels=labels, autopct='% 1.1f%%', pctdistance=1.08,
        labeldistance=0.8, colors=colors, startangle=90, frame=True, explode=explode, radius=10,
        textprops=textprops, counterclock=True, )
plt.pie(sizes_gender, labels=labels_gender, colors=colors_gender, startangle=90,
        explode=explode_gender, radius=7, textprops=textprops, counterclock=True, )
#Draw circle
centre_circle = plt.Circle((0,0),5,color='black', fc='white',linewidth=0)
fig = plt.gcf()
fig.gca().add_artist(centre_circle)

plt.title('Churn Distribution w.r.t Gender: Male(M), Female(F)', fontsize=15, y=1.1)

# show plot

plt.axis('equal')
plt.tight_layout()
plt.show()
fig = px.histogram(data, x="Churn", color = "Contract", barmode = "group", title =
"<b>Customer contract distribution<b>")
fig.update_layout(width=700, height=500, bargap=0.2)
fig.show()
labels = data['PaymentMethod'].unique()

```

```

values = data['PaymentMethod'].value_counts()

fig = go.Figure(data=[go.Pie(labels=labels, values=values, hole=.3)])
fig.update_layout(title_text="<b>Payment Method Distribution</b>")
fig.show()

fig = px.histogram(data, x="Churn", color="PaymentMethod", title="<b>Customer Payment
Method distribution w.r.t. Churn</b>")
fig.update_layout(width=700, height=500, bargap=0.1)
fig.show()
fig = go.Figure()

fig.add_trace(go.Bar(
    x = [['Churn:No', 'Churn:No', 'Churn:Yes', 'Churn:Yes'],
        ["Female", "Male", "Female", "Male"]],
    y = [965, 992, 219, 240],
    name = 'DSL',
))

fig.add_trace(go.Bar(
    x = [['Churn:No', 'Churn:No', 'Churn:Yes', 'Churn:Yes'],
        ["Female", "Male", "Female", "Male"]],
    y = [889, 910, 664, 633],
    name = 'Fiber optic',
))

fig.add_trace(go.Bar(
    x = [['Churn:No', 'Churn:No', 'Churn:Yes', 'Churn:Yes'],
        ["Female", "Male", "Female", "Male"]],
    y = [690, 717, 56, 57],
    name = 'No Internet',
))

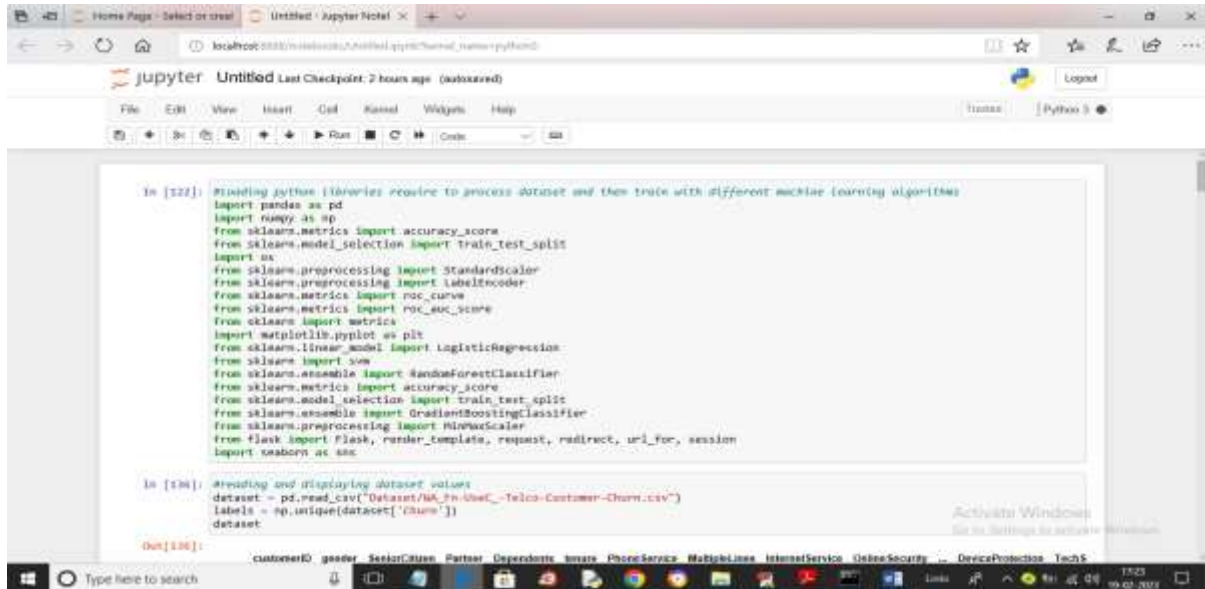
fig.update_layout(title_text="<b>Churn Distribution w.r.t. Internet Service and Gender</b>")

fig.show()

```

4. OUTPUT SCREENS

Below are the JUPYTER code screen with output and blue colour comments



```
In [122]: #Reading python libraries require to process dataset and then train with different machine learning algorithms
import pandas as pd
import numpy as np
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split
import os
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import roc_curve
from sklearn.metrics import roc_auc_score
from sklearn import metrics
import matplotlib.pyplot as plt
from sklearn.linear_model import LogisticRegression
from sklearn import svm
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.preprocessing importMinMaxScaler
from flask import Flask, render_template, request, redirect, url_for, session
import random as rnd
```

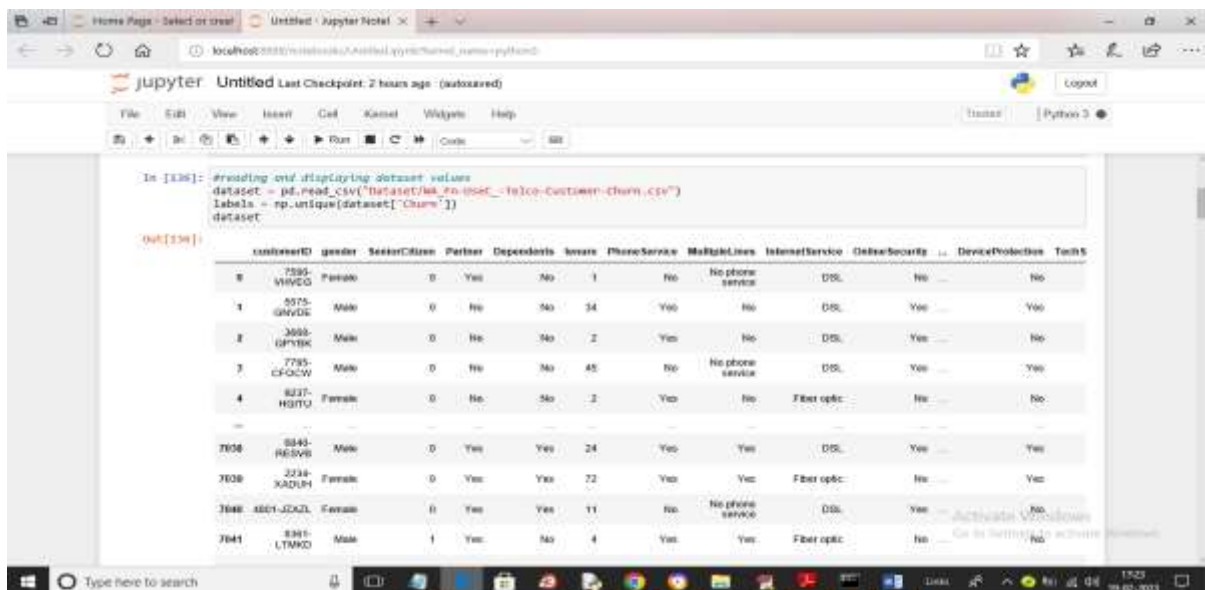
```
In [126]: #reading and displaying dataset values
dataset = pd.read_csv("Dataset/WA_Fn-UseC_-Telco-Customer-Churn.csv")
labels = np.unique(dataset['Churn'])
dataset
```

Out[126]:

customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	DeviceProtection	Tech5
5759-VIWOQ	Female	0	Yes	No	1	No	No phone service	DSL	No	...	No
5575-GHVDG	Male	0	No	No	34	Yes	No	DSL	Yes	...	Yes
3065-GHVDG	Male	0	No	No	2	Yes	No	DSL	Yes	...	No
7795-CFOCW	Male	0	No	No	45	No	No phone service	DSL	Yes	...	Yes
8227-HGTYU	Female	0	No	No	2	Yes	No	Fiber optic	No	...	No
...
7038-RESDG	Male	0	Yes	Yes	24	Yes	Yes	DSL	Yes	...	Yes
7038-XADUH	Female	0	Yes	Yes	72	Yes	Yes	Fiber optic	No	...	Yes
7688-ABOI-JZADL	Female	0	Yes	Yes	11	No	No phone service	DSL	Yes	...	No
7641-LTKMD	Male	1	Yes	No	4	Yes	Yes	Fiber optic	No	...	No

Fig 4.1 import packages

In above screen importing all require python packages



```
In [126]: #reading and displaying dataset values
dataset = pd.read_csv("Dataset/WA_Fn-UseC_-Telco-Customer-Churn.csv")
labels = np.unique(dataset['Churn'])
dataset
```

Out[126]:

customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	DeviceProtection	Tech5
5759-VIWOQ	Female	0	Yes	No	1	No	No phone service	DSL	No	...	No
5575-GHVDG	Male	0	No	No	34	Yes	No	DSL	Yes	...	Yes
3065-GHVDG	Male	0	No	No	2	Yes	No	DSL	Yes	...	No
7795-CFOCW	Male	0	No	No	45	No	No phone service	DSL	Yes	...	Yes
8227-HGTYU	Female	0	No	No	2	Yes	No	Fiber optic	No	...	No
...
7038-RESDG	Male	0	Yes	Yes	24	Yes	Yes	DSL	Yes	...	Yes
7038-XADUH	Female	0	Yes	Yes	72	Yes	Yes	Fiber optic	No	...	Yes
7688-ABOI-JZADL	Female	0	Yes	Yes	11	No	No phone service	DSL	Yes	...	No
7641-LTKMD	Male	1	Yes	No	4	Yes	Yes	Fiber optic	No	...	No

Fig 4.2 Read dataset

In above screen reading and displaying dataset values and in above dataset we can see there are numeric and non-numeric values but ML algorithms will accept only numeric values so we need to employ label encoder class to convert all non-numeric data to numeric values.

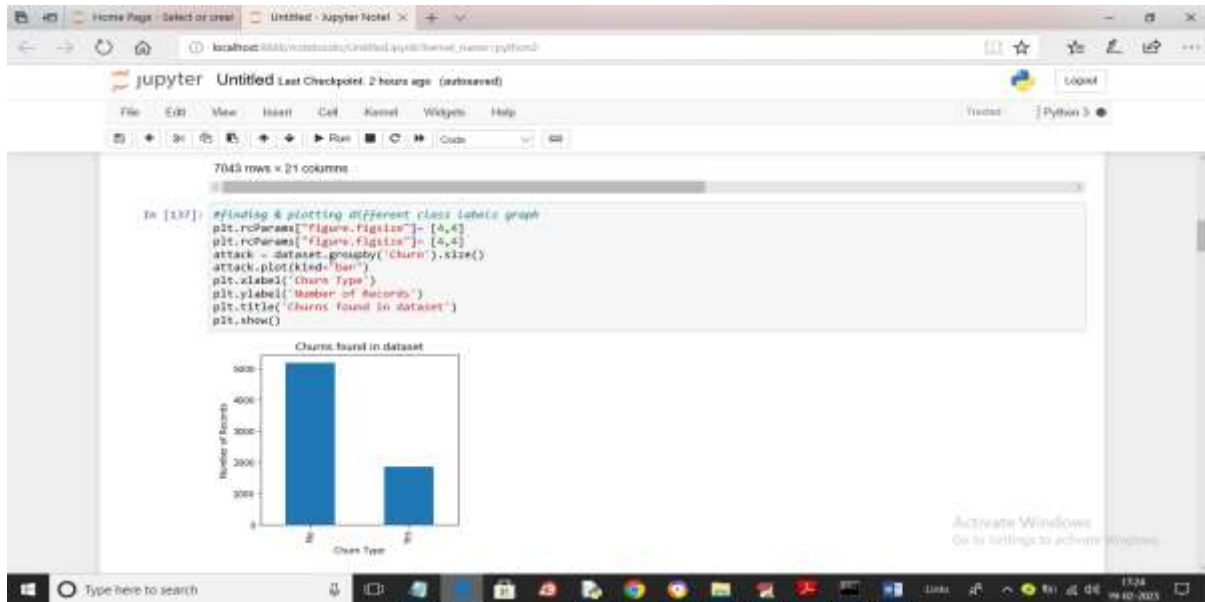


Fig 4.3 Display dataset 1

In above screen finding and plotting class labels from dataset where x-axis represents CHURN type as Yes or No and y-axis represents counts of record.

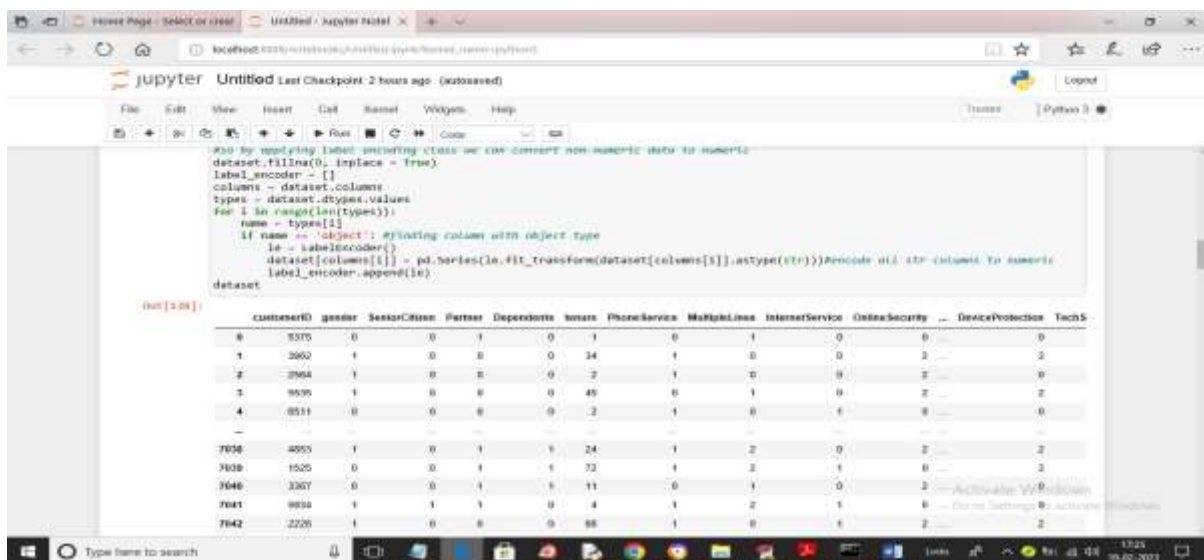


Fig 4.4 Display dataset 2

In above screen employed label encoder class to convert all non-numeric data to numeric values and in above screen we can see all values are numeric only

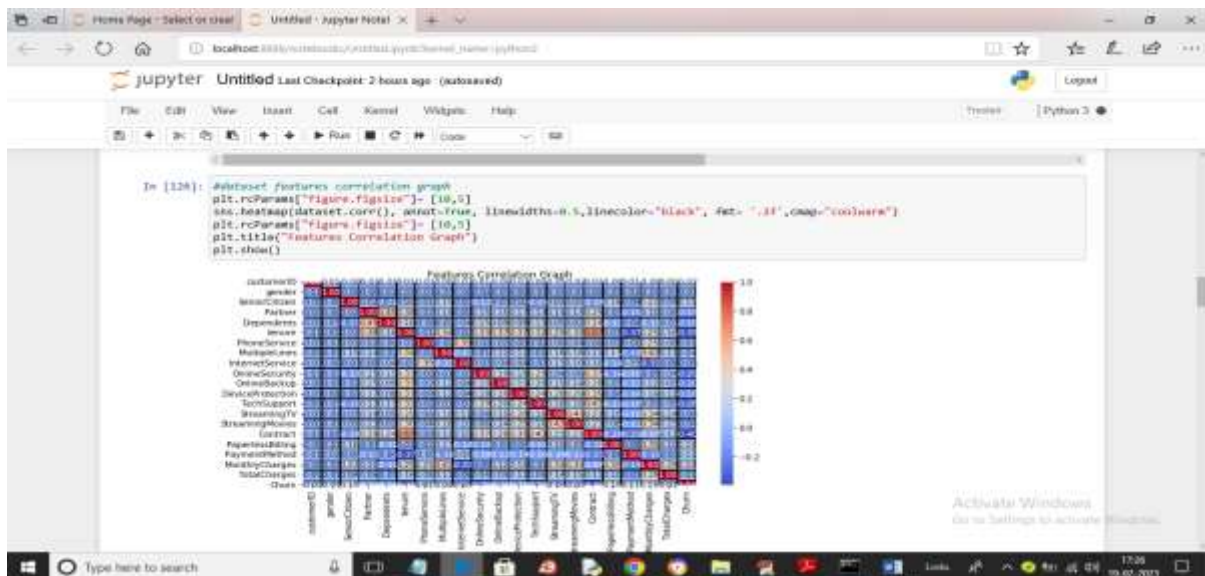


Fig 4.5 dataset features correlation graph

In above graph we are plotting dataset features correlation graph and in red box we can see all features got value 1 so all features are important

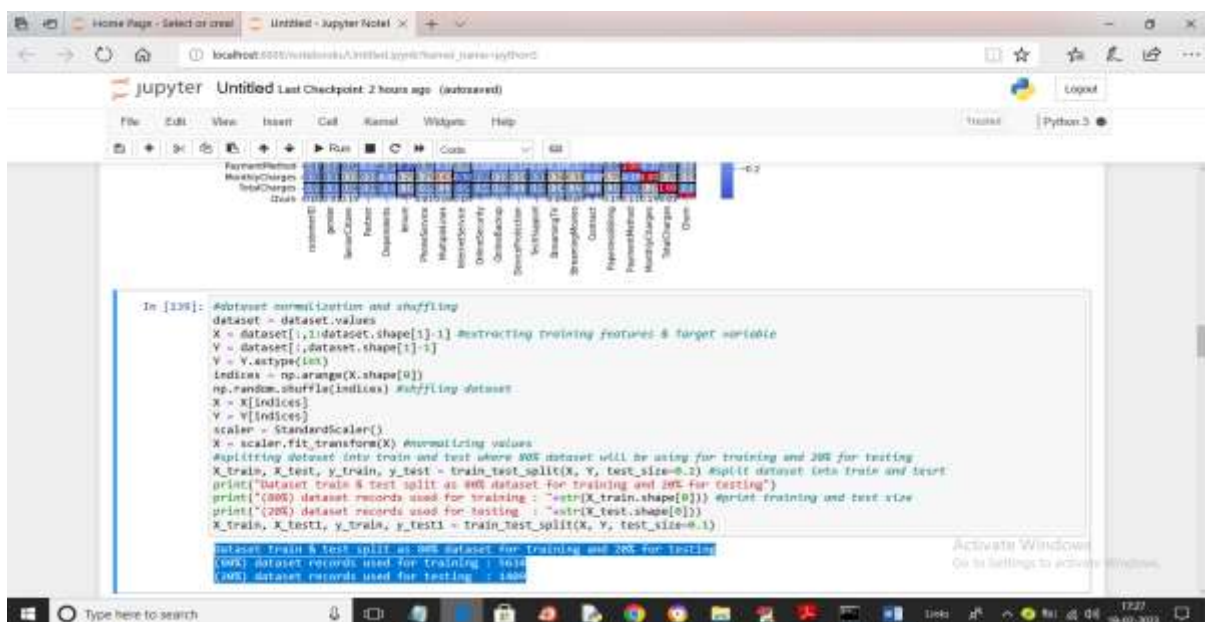
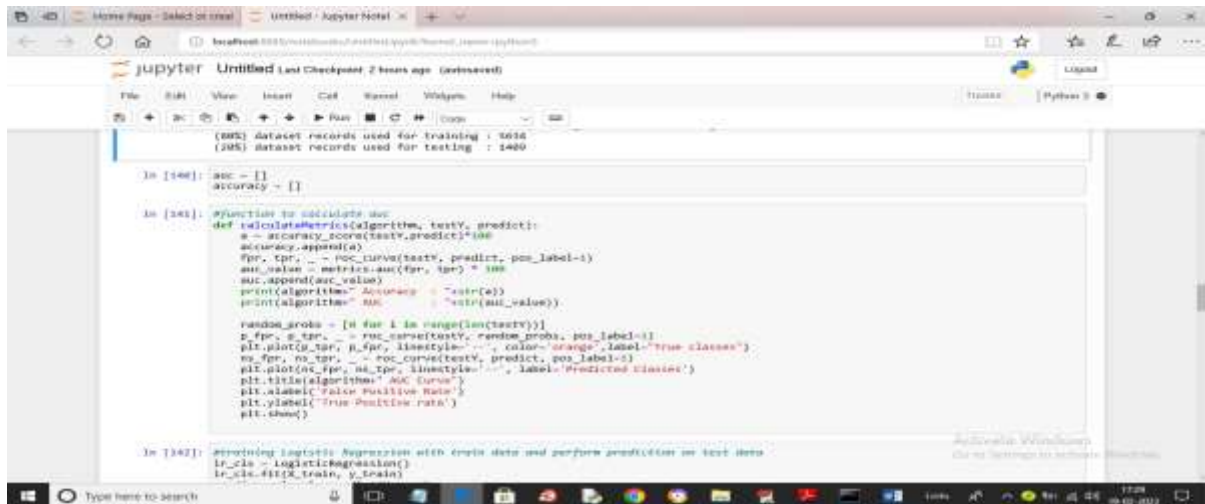


Fig 4.6 Splitting dataset

In above screen we are extracting X training features and Y target variable and then normalizing and shuffling dataset and then splitting dataset into train and test where application using 80% dataset for training and 20% for testing and in blue colour we can see total values used for training and testing



```

(885) dataset records used for training : 5614
(205) dataset records used for testing : 1400

In [146]: acc = []
          accuracy = []

In [147]: #function to calculate auc
          def calculateMetrics(algorithm, testy, predict):
              a = accuracy_score(testy, predict)*100
              accuracy.append(a)
              fpr, tpr, _ = roc_curve(testy, predict, pos_label=1)
              auc_value = metrics.auc(fpr, tpr) * 100
              auc.append(auc_value)
              print(algorithm+" Accuracy : "+str(a))
              print(algorithm+" AUC : "+str(auc_value))

              random_probs = [i for i in range(len(testy))]
              p_fpr, p_tpr, _ = roc_curve(testy, random_probs, pos_label=1)
              plt.plot(p_fpr, p_tpr, linestyle='--', color='orange', label='true classes')
              m_fpr, m_tpr, _ = roc_curve(testy, predict, pos_label=1)
              plt.plot(m_fpr, m_tpr, linestyle='--', label='Predicted classes')
              plt.title(algorithm+" ROC Curve")
              plt.xlabel('false positive rate')
              plt.ylabel('true positive rate')
              plt.show()

In [147]: #training logistic Regression with train data and perform prediction on test data
          lr_cls = LogisticRegression()
          lr_cls.fit(X_train, y_train)
  
```

Fig 4.7 AUC

In above screen defining function to calculate accuracy and AUC values

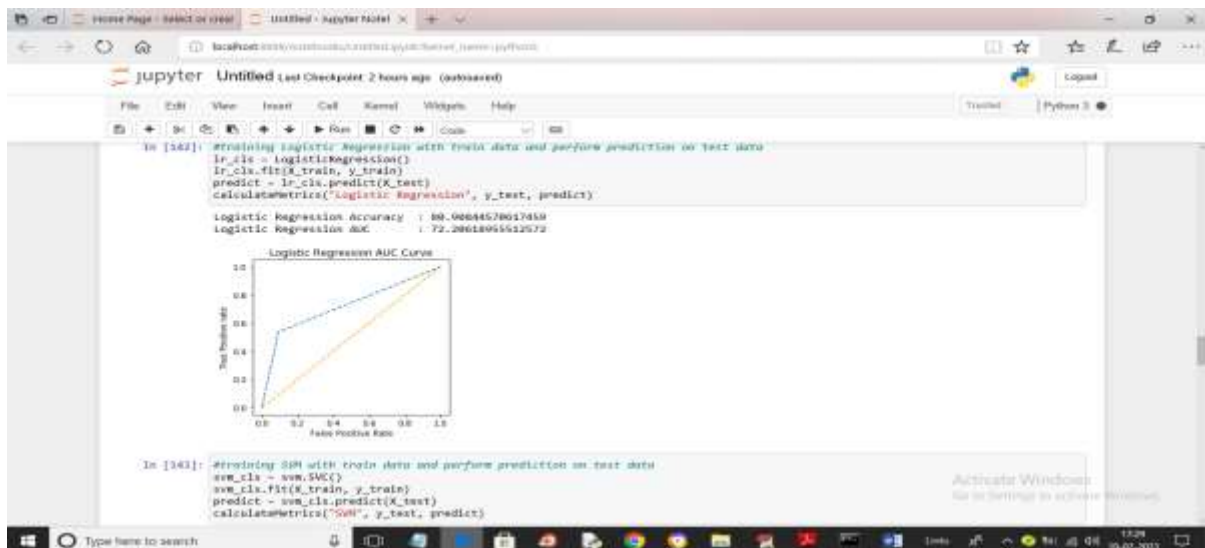


Fig 4.8 Train Logistic regression

In above screen we are training Logistic Regression algorithm and we got its accuracy as 80% and AUC as 72% and in AUC graph x-axis represents False Positive Rate and y-axis represents True Positive Rate and if blue lines comes on top of orange line then all predictions are correct and if goes below orange line then all predictions are incorrect or false

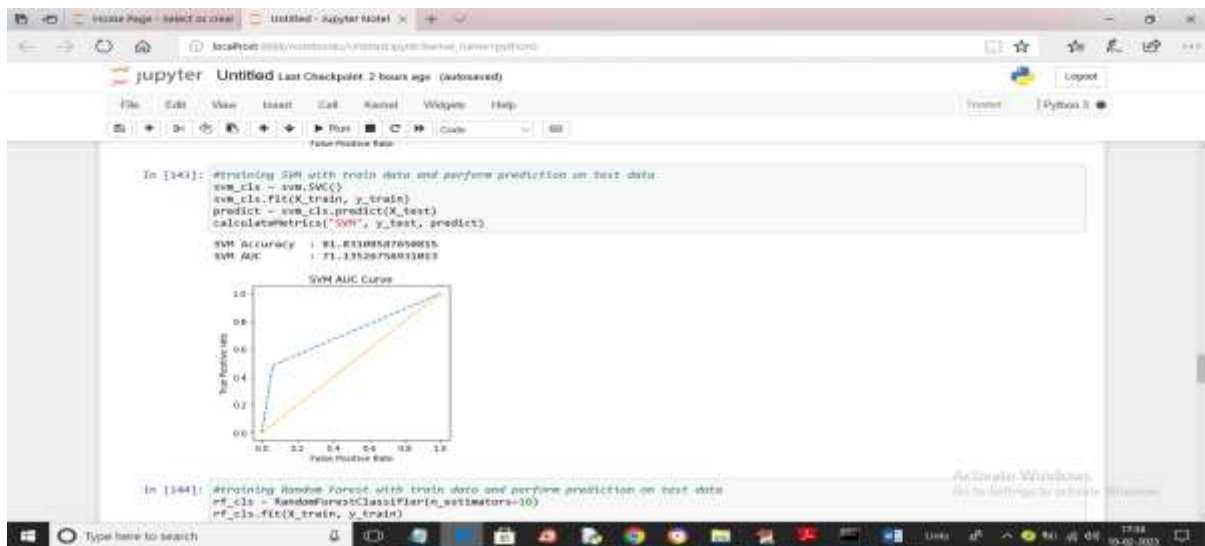


Fig 4.9 Train SVM

In above screen training SVM and got its accuracy as 81%

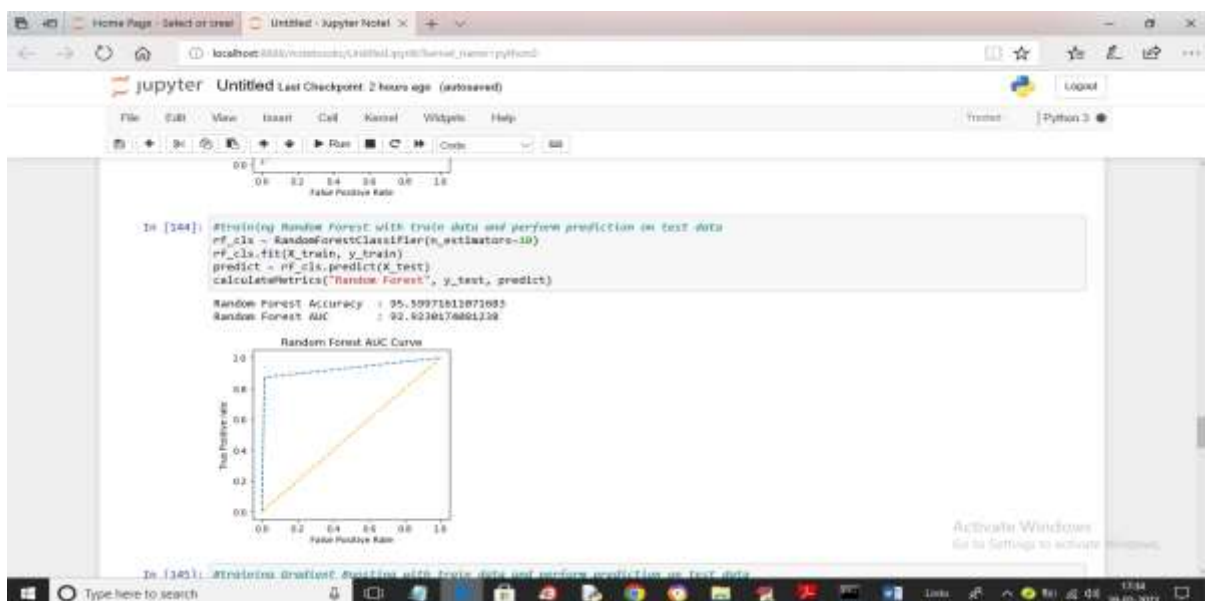


Fig 4.10 Train Random forest

In above screen training Random Forest and got its accuracy as 95%

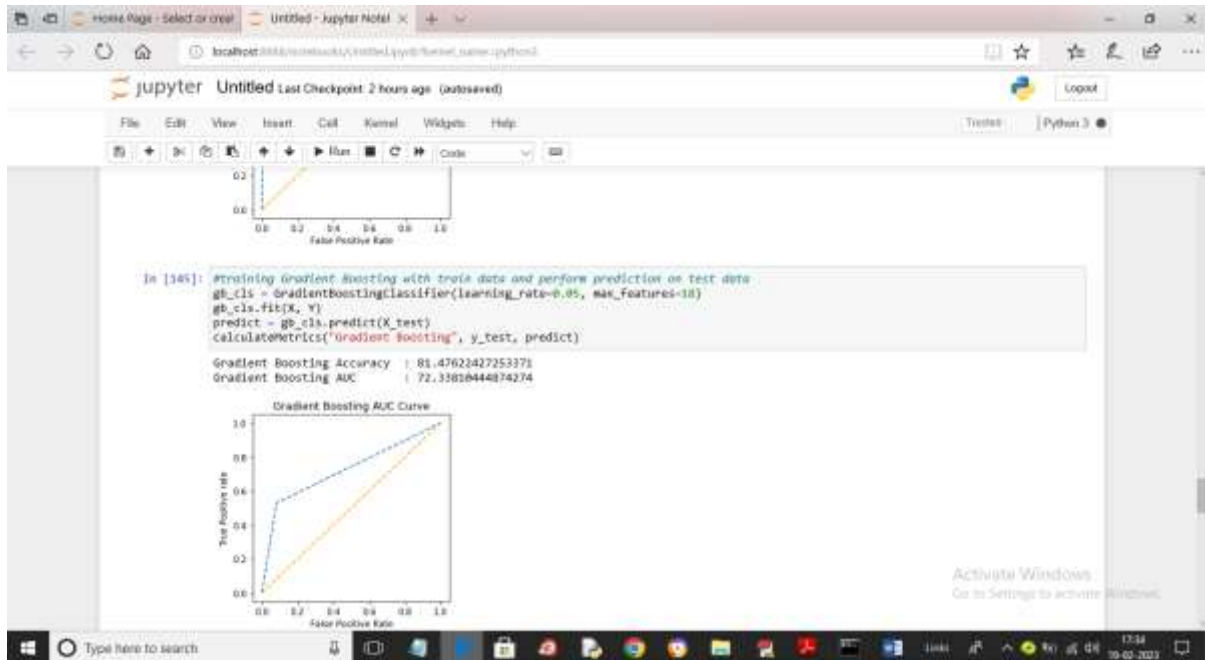


Fig 4.11 Train Gradient boosting

In above screen training Gradient Boosting and got its accuracy as 81%

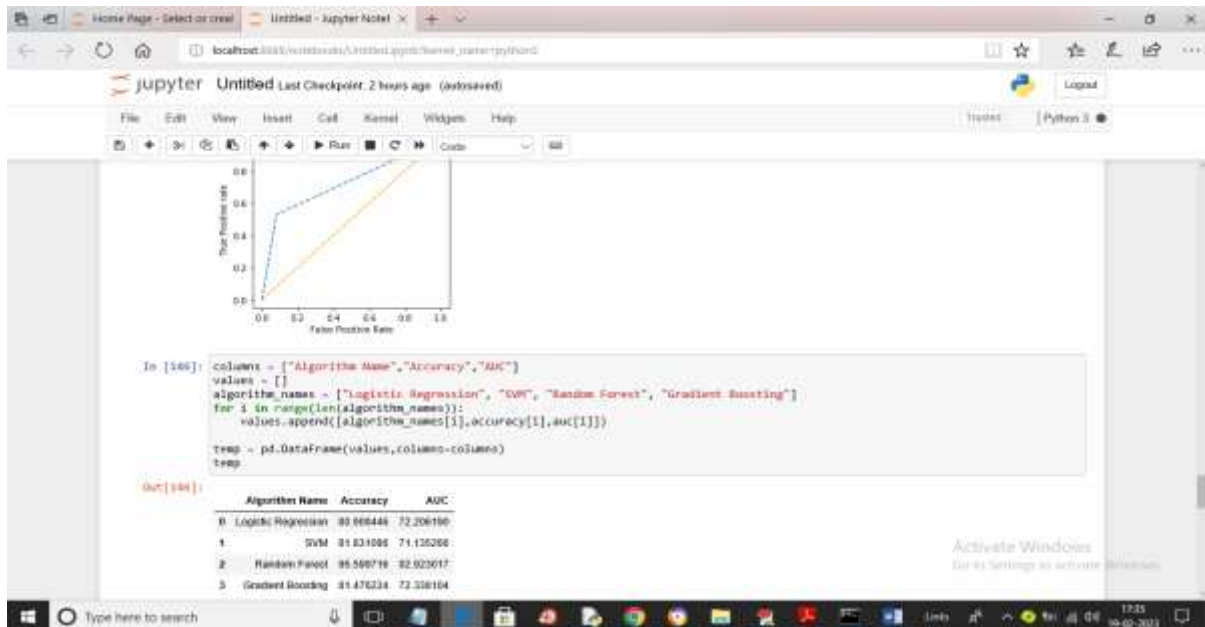



Fig 4.12 Algorithms performance table



The screenshot shows a Jupyter Notebook interface with the following code:

```
In [*]: from flask import Flask, render_template, request, redirect, url_for, session
from werkzeug.utils import secure_filename

app = Flask(__name__)
app.secret_key = 'churn'

@app.route('/index', methods=['GET', 'POST'])
def index():
    return render_template('index.html', msg='')

@app.route('/ChurnPrediction', methods=['GET', 'POST'])
def ChurnPrediction():
    return render_template('ChurnPrediction.html', msg='')

@app.route('/ChurnPredictionaction', methods=['GET', 'POST'])
def ChurnPredictionaction():
    if request.method == 'POST':
        column_details = 'customerID,gender,SeniorCitizen,Partner,Dependents,tenure,PhoneService,MultipleLines,InternetService,Or
        data = request.form['t1']
        arr = data.split(',')
        col_arr = column_details.split(',')
        values = {}
        for i in range(len(arr)):
            if i == 2:
                values.append(int(arr[i].strip()))
            elif i == 5:
                values.append(int(arr[i].strip()))
            else:
                i = 10
```

The screenshot displays a Jupyter Notebook environment with the following code and output:

```

testData['MonthlyCharges'] = testData['MonthlyCharges'].astype(float)
index = 0
columns = testData.columns
types = testData.dtypes.values
for i in range(len(types)):
    name = types[i]
    if name == 'object': #finding column with object type
        testData[columns[i]] = pd.Series(label_encoder[index].transform(testData[columns[i]].astype(str)).encode('utf-8'))
        index = index + 1
testData = testData.values
test = testData[:,1:testData.shape[1]]
test = scaler.transform(test)
predict = rf.predict(test)
output = "Text Data "+str(testData[0])+"====> Predicted As "+labels[predict]
return render_template("churnPrediction.html", msg=output)

if __name__ == "__main__":
    app.run()

```

The output section shows the following messages:

- * Serving flask app "__main__" (lazy loading)
- * Environment: production
- WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
- * Debug mode: off
- * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)

The terminal output shows a GET request to the index endpoint:

```

127.0.0.1 ~ - [15/Feb/2023 17:38:58] "GET /index HTTP/1.1" 200 -

```

19

In above screen defining FLASK code to perform prediction and run this block to start Flask server and get below output

In above screen Flask server started and now open browser and enter URL as 'http://127.0.0.1:5000/index' and press enter key to get below page



Fig 4.15 Churn Prediction Test Data

In above screen click on 'Churn Prediction Test Data' link to get below screen



Fig 4.16 Prediction Screen

Customer Churn Prediction

127.0.0.1:5000/C churnPredictionFunction

Machine Learning for Telecom Customers Churn Prediction

COLIFSETO

project network

4.6 (29)

User Login Screen

Text Data [5.656e+03 0.000e+00 1.800e+00 0.900e+00 0.500e+00 4.300e+01 1.000e+00 2.000e+00 1.000e+00 0.000e+00 3.000e+00 0.000e+00 0.800e+00 2.000e+00 0.000e+00 0.000e+00 1.000e+00 2.000e+00 9.025e+01 3.285e+03] ==> Predicted As No

Enter Churn Data:

Activate Windows
Go to Settings to activate Windows.

In above screen in blue colour text inside square bracket we can see normalized test values and after arrow symbol => we can see predicted churn value as 'No' and similarly you can copy one from 'Dataset file or testData.csv' file and paste in text field to get churn prediction. Below screen showing another record



21

In above screen for another record we got predicted value as 'Yes' after arrow symbol.

Below screen showing testData.csv file content

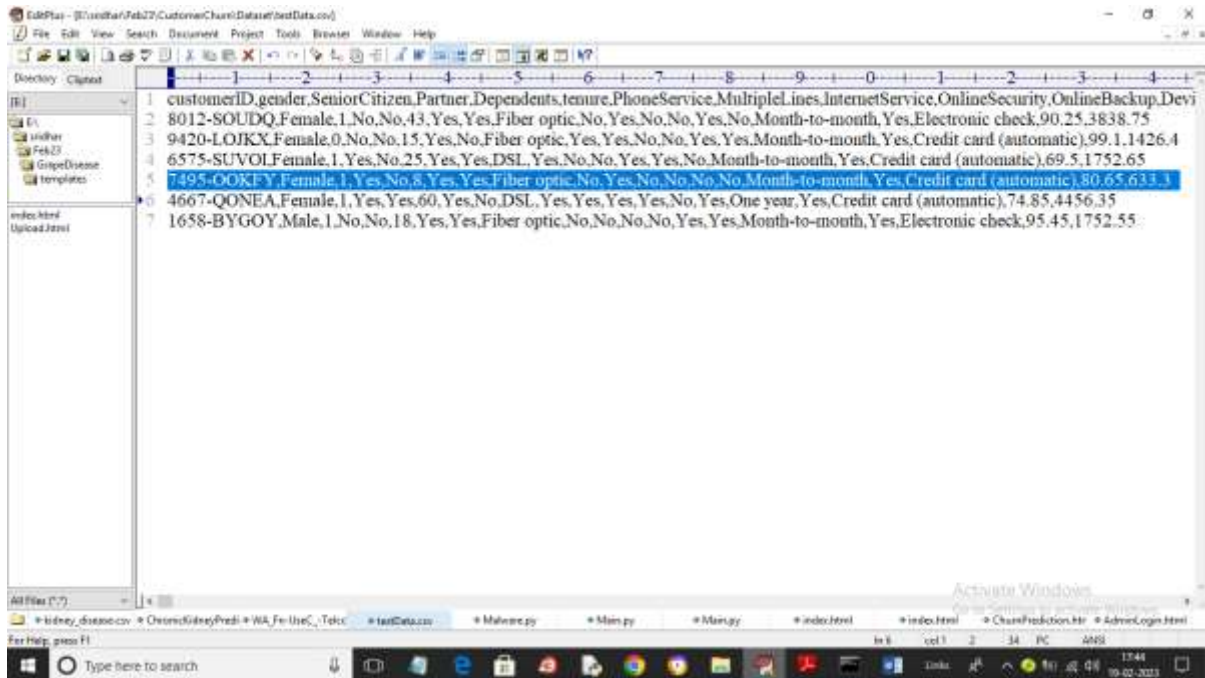


Fig 4.19 Prediction

From above test data you can copy one row and paste in application to get predicted output

5. CONCLUSION

In order to retain existing customers, Telecom providers need to know the reasons of churn, which can be realized through the knowledge extracted from Telecom data. In this paper, we train four machine learning models which is Logistic Regression, SVM, Random Forest and Gradient boosted tree and we can say that Random forest is best in among four models.

6. FUTURE SCOPE

The future scope of this paper will use hybrid classification techniques to point out existing association between churn prediction and customer lifetime value. The retention policies need to be considered by selecting appropriate variables from the dataset. The passive and the dynamic nature of the industry ensure that data mining has become increasingly significant aspect in the telecommunication industry prospect.

7. BIBILOGRAPHY

- [01] Peng Li 1, 2, Siben Li 2, Tingting Bi 2, Yang Liu 2, " Telecom Customer Churn Prediction Method Based on Cluster Stratified Sampling Logistic Regression" in IEEE.
- [02] Chuanqi Wang, Ruiqi Li, Peng Wang, Zonghai Chen, "Partition cost-sensitive CART based on customer value for Telecom customer churn prediction" in Proceedings of the 36th Chinese Control Conference 2017 IEEE.
- [03] Guo-en Xia, Hui Wang, Yilin Jiang, "Application of Customer Churn Prediction Based on Weighted Selective Ensembles" in IEEE 2016.
- [04] Rahul J. Jadhav, Usharani T. Pawar, "Churn Prediction in Telecommunication Using Data Mining Technology", in (IJACSA), Vol. 2, No.2, February 2011
- [05] Kiran Dahiya, Surbhi Bhatia, "Customer Churn Analysis in Telecom Industry" in IEEE 2015, 978-1-4673-7231- 2/15
- [06] N.Kamalraj, A.Malathi' " A Survey on Churn Prediction Techniques in Communication Sector" in IJCA Volume 64– No.5, February 2013
- [07] Kiran Dahiya,KanikaTalwar, "Customer Churn Prediction in Telecommunication Industries using Data Mining Techniques- A Review" in IJARCSSE, Volume 5, Issue 4, 2015.
- [08] R Data: <http://cran.r-project.org/>
- [09] Data Mining in the Telecommunications Industry, Gary M. Weiss, Fordham University, USA.
- [10] Manjit Kaur et al., 2013.Data Mining as a tool to Predict the Churn Behaviour among Indian bank customers, IJRITCC, Volume: 1 Issue: 9
- [11] R. Khare, D. Kaloya, C. K. Choudhary, and G. Gupta, "Employee attrition risk assessment using logistic regression analysis,".

- [12] Praveen et al., Churn Prediction in Telecom Industry Using R, in (IJETR) ISSN: 2321-0869, Volume-3, Issue- 5, May 2015
- [13] J. Burez and D. Van den Poel, “Handling class imbalance in customer churn prediction,” *Expert Systems with Applications*, vol. 36, no. 3, 2009.
- [14] W. Verbeke, K. Dejaeger, D. Martens, J. Hur, and B. Baesens, “New insights into churn prediction in the telecommunication sector: A profit driven data mining approach,” *European Journal of Operational Research*, vol. 218, no. 1, pp. 211–229, 2012.