Coursework id: AM41UD

Student id: 210274503

THANUJA MODIBOINA

Understanding data Assessment

Abstract

The customer churn problem is one of the main problems of telecom companies. It is a challenging task for every telecommunication company. Because of the competition between the industries, customers are churned. There are several factors affecting churn. Like bad customer service, price isn't good, offers to customers, communications with customers etc. Lu's communications which is a telecom company in the U.K. This company also facing the same issue. To lose customers, this company is trying to implement special programs for those who are likely to churn. So, here we are predicting customer churn based on previous historical data. In our methodology, there are six steps involved. They are data collection, data cleaning and preparation, EDA(exploratory data analysis), initial hypothesis, data pre-processing and building models. The data has been split into 80% and 20%. In our prediction process, 4 predictive models are applied. They are Logistic Regression, K-Nearest Neighbor, Decision Tree, and Random Forest Classifier. Finally, the models are built and trained then predicted churn information. Accuracy of Logistic Regression is 84.69%, K-Nearest Neighbor is 84.90%, Decision Tree is 85.24 and Random Forest Classifier is 85.73%. The highest accuracy is achieved by Random Forest Classifier.

Table of contents

Introduction:

Churn rate is the key parameter for company growth. The rate of consumer dissatisfaction with a good or service over time is known as the churn rate. It makes the integrity of the company clear, demonstrates client experience, and enables comparison with competing companies to determine an acceptable amount of attrition.[1]The churn rate and company growth are inversely proportional to each other. If the churn rate is increased, then company growth decreases or if the churn rate is decreased then company growth increases. The growth rate of the company represents the new customers(gained customers) and the churn rate monitors the loss of customers.

An excellent technique to develop proactive marketing initiatives that are directed at consumers that are going to churn is to predict customer churning. We need ways to lower the churn rate to increase customer happiness and profit because its impacts are obvious.[1] A company needs to know this information since it might be difficult and expensive to acquire new consumers compared to keeping existing ones. One technique to determine a company's churn rate is the fraction of the total number of active customers at the beginning of the period by the number of users who left during that time. Thus, the knowledge acquired through churn prediction enables them to concentrate more on the clients who are most likely to leave.[2] Customer churn is a global issue, and the average churn rate is increasing.

Objective:

Here, the main goal is to predict customer churn in Lu’s communications company using previous historical data. The telecommunications sector has become one of the leading industries so, we have to deal with business properly without losing customers.

About dataset:

Lu’s communication is a telecom company which is held in the UK. This company provide services to customers like the Internet, Landline and TV. Due to competition between industries, this company is facing a problem with churn. Lu’s communications decided to use a targeted approach to identify the customers in advance who are likely to churn, which takes a special program approach.

The goal of Lu's communications is to develop a focused strategy to predict which consumers are most likely to move on so that they may offer them bonuses or special deals.

There are steps for the prediction of customer churn. They are

1)Data collection

The main dataset has 7350 customers as observations and 12 features. First, we must read a CSV file in python. Here, we must predict customer attrition based on certain features.

In the given data, it contains outliers, missing values, and irrelevant data that is not related to the prediction of churn.

There are 12 features in this dataset including one indexed column. That column is removed then only 11 features are available. They are

Features	Description of feature	Type
1)customer_id	A unique ID which identifies every customer.	Categorical
2) gender	male or female	Categorical
3)location	Residence of customer	Categorical
4)partner	If the customer has a partner, then Yes or else No.	Numerical
5)dependents	If the customer has a dependent, then Yes or else No.	Categorical
6)senior	Is the customer a senior citizen or not? (1=Yes or 0=No)	Numerical
7)Tenure	No.of years the customer stayed in the company.	Numerical
8)monthly_cost	The amount spent by the customer per month.	Categorical
9)package	Packages which are offered by Lu's company.	Numerical
10)survey	Customer service score is given by customers. (0='poor', 10='Excellent')	Categorical
11)Class	Customer churned or not.	Categorical

Customer-based factors: gender, partner, dependents and senior.

Service-based factors: This captures the data about customers' use of the data. Such features as monthly_cost and survey.

2)Data cleaning and preparation

First, we must read a CSV file in python. The main dataset has 7350 customers as observations and 12 features including an index. After removing an index column, only 11 features are available. Here, we must build a machine learning model to predict customer attrition based on features.

For every dataset first, we must clean the dataset and transform the data into a certain format then be ready to apply algorithms and build models.

Steps to clean the data:

- Check the structure of the dataset i.e., how many observations and features are in the data.
- Identify the features which are not useful to predict churn attrition.
- Mention the type of features, which are numerical or categorical. Check the type if we want to change the type or not.

Load the dataset in python that is reading CSV file. Check the number of null values or nan values and their type. In our dataset, 7271 nan values in the 'monthly_cost' feature and 59 null values in the 'Class'.

A feature is referred to as numerical if it has an int or float data type, and categorical if it has an object datatype or has string values.

There are 7 categorical and 4 numerical features. 7 Categorical features are 'customer_id', 'gender', 'location', 'dependents', 'monthly_cost', 'survey', and 'Class'. 4 Numerical features are 'partner', 'senior', 'Tenure', and 'package'.

Here some features need to be formatting i.e., change the data types of the features. That is 'gender', 'monthly_cost', 'survey', and 'Class'. The 'monthly_cost' is not categorical. So, we need to change the datatype to numerical. The features such as 'gender', 'survey', and 'Class' are also better to change the datatype to numerical. Then it is easier to do further processing.

At present monthly_cost column is given with nan values. Before changing the format of those features, calculate the values of monthly_cost. From the package, which services are used by customers, based on that we calculate monthly_cost. This is the amount charged to the customer monthly. Then the total_cost is calculated by multiplying the 'monthly_cost' and 'Tenure'.

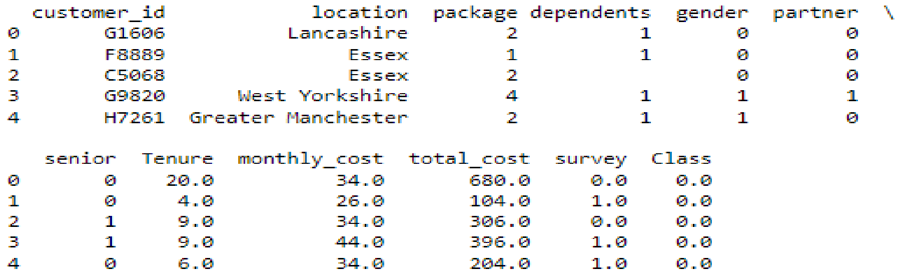
total_cost = monthly_cost * Tenure

This is the total_cost charged to the customer calculated to the end of the quarter(Tenure period). The 'Tenure' represents the number of years the customer has stayed in the company. In the given dataset, negative values are also given. But it is not appropriate. Tenure value should be in positive number which is customer stayed period. So, the negative values of the tenure column are removed. Now, the shape of the dataset is 7224 observations and 12 features.

The 'gender' and 'Class' are categorical types. The gender is whether the customer is a male or female. So, we can convert it into 1 and 0 (male = 1, female=0). Same as Class represents whether the customer wants to churn or not. So, it is converted to 0 or 1 (Yes=1, No =0).

The 'survey' is about customer service (0 = "poor", 10 = "Excellent"). This is the main factor in churn prediction. If the service is good, the customers will stay or else they will leave. There are some string characters in the survey feature column. These are replaced with 0. In the survey, if a service score is >5 taken as good service and <5 as bad services. Based on this, we can convert these values into 0's and 1's (good service = 1, bad service = 0). So, this feature is converting its type to numerical.

In the dependents column also, some string characters are removed. Finally, the Class feature has 76 null values that are removed. After removing all missing values, then remove irrelevant features from the dataset for the prediction of customer churn.



```
  customer_id             location  package dependents  gender  partner  \
0       G1606           Lancashire        2          1       0        0
1       F8889                Essex        1          1       0        0
2       C5068                Essex        2                  0        0
3       G9820       West Yorkshire        4          1       1        1
4       H7261   Greater Manchester        2          1       1        0

   senior  Tenure  monthly_cost  total_cost  survey  Class
0       0    20.0          34.0       680.0     0.0    0.0
1       0     4.0          26.0       104.0     1.0    0.0
2       1     9.0          34.0       306.0     0.0    0.0
3       1     9.0          44.0       396.0     1.0    0.0
4       0     6.0          34.0       204.0     1.0    0.0
```

Figure1: After cleaning the dataset

Now, the data is in the format ready to apply with the algorithms and modelling.

Here is the unstructured data into structured data so that our ML model can process the data and takes a further step and visualize the relation between features.

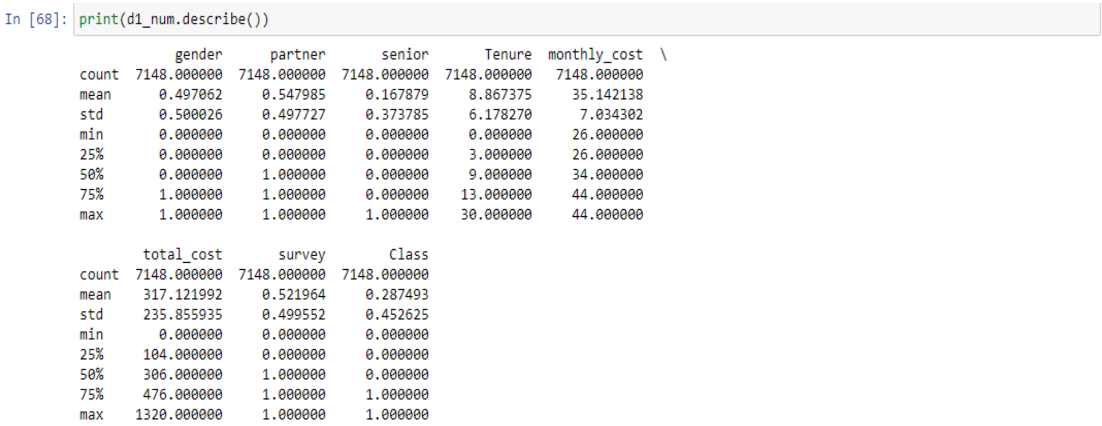
3)Exploratory data analysis

EDA is a crucial step in Machina learning. To create a valuable product using the data, you need to explore the data.

To explore and analyze the data, there are two methods are used in exploratory data analysis. They are descriptive statistical analysis and descriptive graphical analysis.

i. Descriptive statistical analysis

Descriptive statistical analysis, which gives information about statistical parameters of features. So, new datasets are created, that is d1_num which consists of numerical features and d1_cat consists of categorical features.



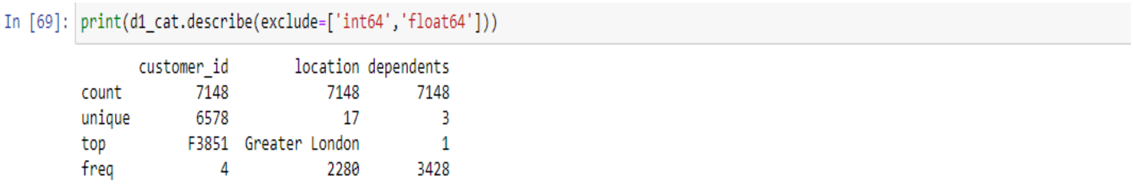
```
In [68]: print(d1_num.describe())

              gender      partner       senior       Tenure  monthly_cost  \
count  7148.000000  7148.000000  7148.000000  7148.000000   7148.000000
mean      0.497062     0.547985     0.167879     8.867375     35.142138
std       0.500026     0.497727     0.373785     6.178270      7.034302
min       0.000000     0.000000     0.000000     0.000000     26.000000
25%       0.000000     0.000000     0.000000     3.000000     26.000000
50%       0.000000     1.000000     0.000000     9.000000     34.000000
75%       1.000000     1.000000     0.000000    13.000000     44.000000
max       1.000000     1.000000     1.000000    30.000000     44.000000

        total_cost       survey        Class
count  7148.000000  7148.000000  7148.000000
mean    317.121992     0.521964     0.287493
std     235.855935     0.499552     0.452625
min       0.000000     0.000000     0.000000
25%     104.000000     0.000000     0.000000
50%     306.000000     1.000000     0.000000
75%     476.000000     1.000000     1.000000
max    1320.000000     1.000000     1.000000
```

Figure2: Descriptive statistical analysis of the dataset

The descriptive statistics of categorical features are



```
In [69]: print(d1_cat.describe(exclude=['int64','float64']))

       customer_id        location dependents
count         7148            7148       7148
unique        6578              17          3
top          F3851  Greater London          1
freq             4            2280       3428
```

Figure3: Descriptive graphical analysis of the dataset

ii. Descriptive graphical analysis

Descriptive graphical analysis means analyzing the data with the help of graphs.

1. Histogram of all numerical features

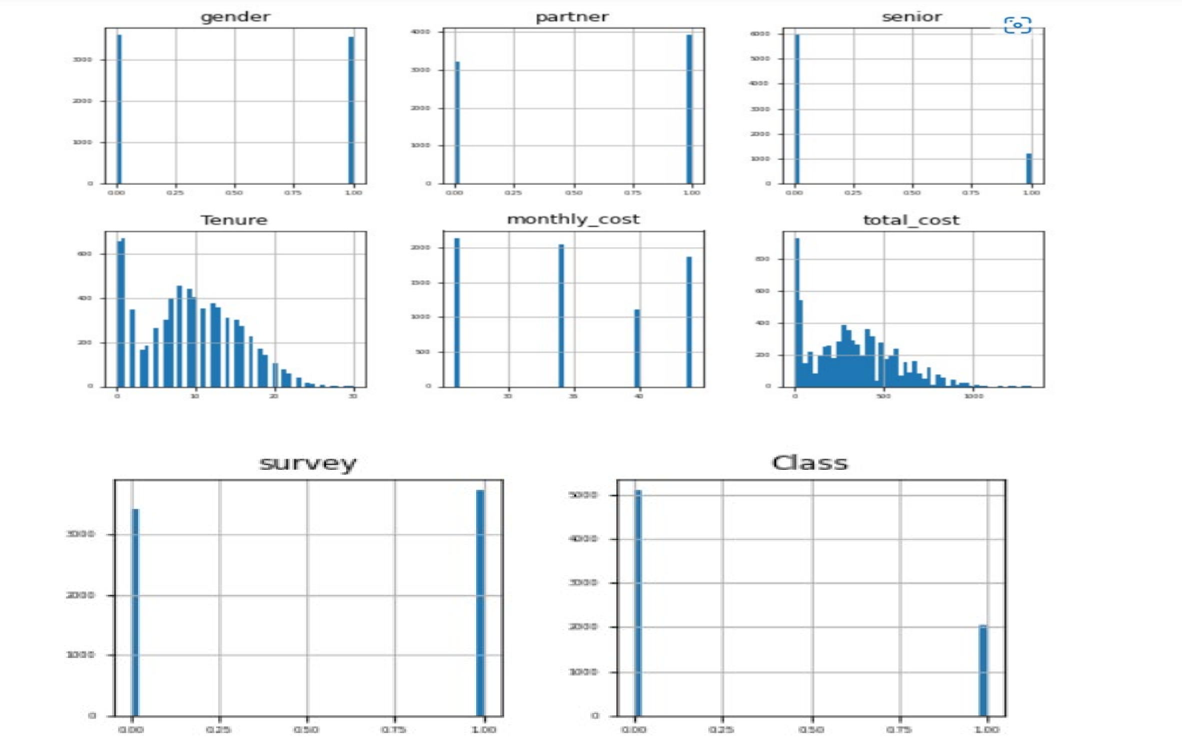


Figure4: Histogram of all numerical features

2.A bar plot for the Class column i.e., if the customer wants to churn(yes =1) or else (No = 0).

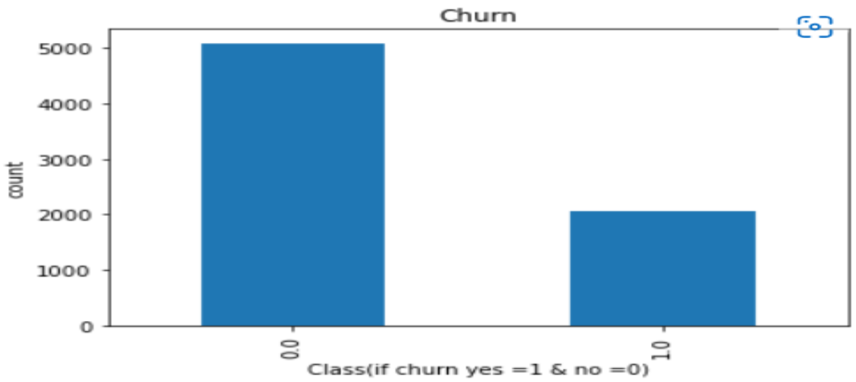


Figure5: plot for Churn

From the plot, we can say there are a lot of people who want to churn from the company than who want to stay in the company.

3. A plot which shows the relation between gender and customer churn(class). In the plot, male=1 and female=0.

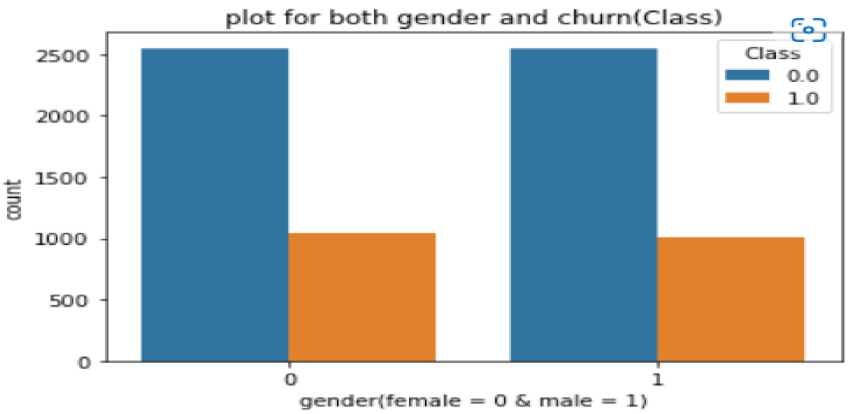


Figure6: Plot which shows the relation between gender and churn

4. Plot for churn(Class) based on the package(services used by the customer)

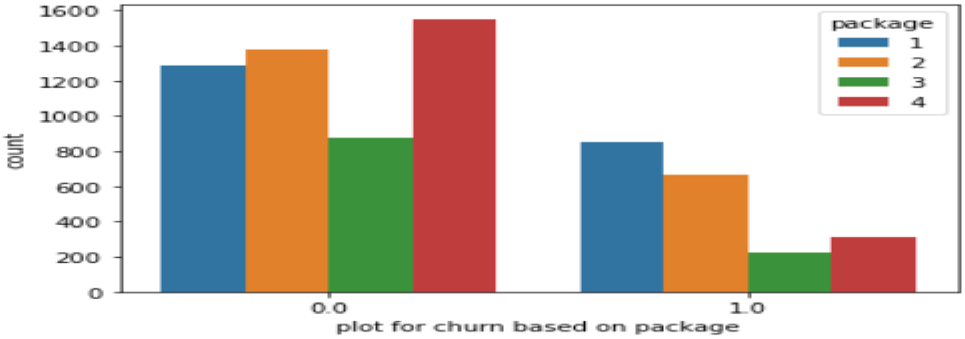
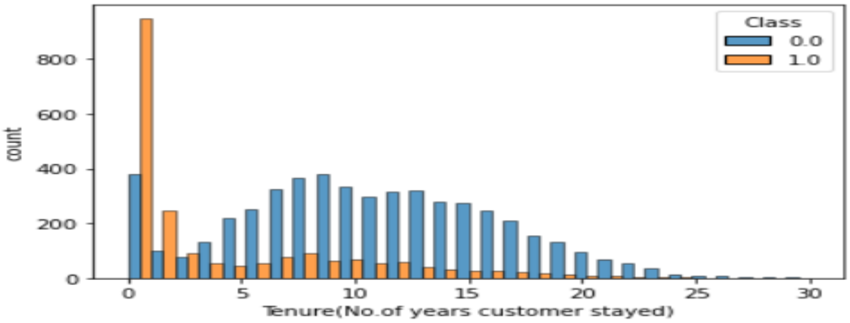


Figure7: Plot which shows the relation between churn and package

5. Histogram of Tenure based on Class(churn)



`

Figure8: Plot which shows the relation between churn and tenure

6. Histogram of monthly_cost based on Class(churn)

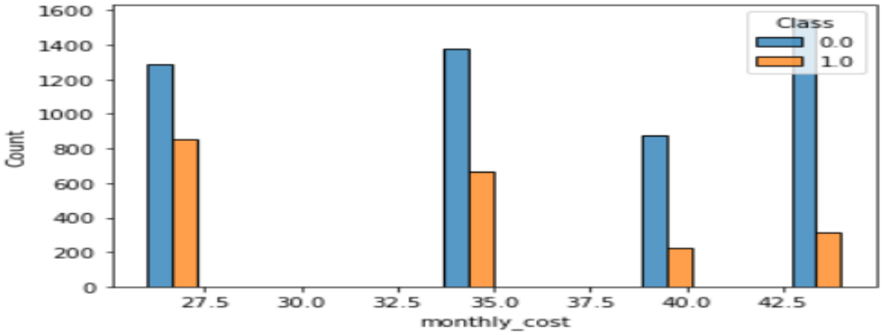


Figure8: Plot which shows the relation between churn and monthly_cost

7. Histogram of total_cost based on Class(churn)

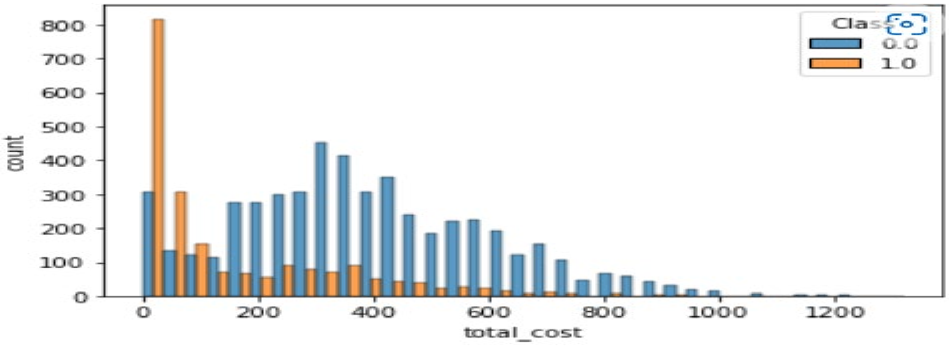


Figure9: Plot which shows the relation between churn and total_cost

4)Initial hypothesis

From the dataset, here we will predict churn attrition based on 6 features. They are 'partner', 'senior', 'Tenure', 'monthly_cost', 'total_cost', and 'survey'.

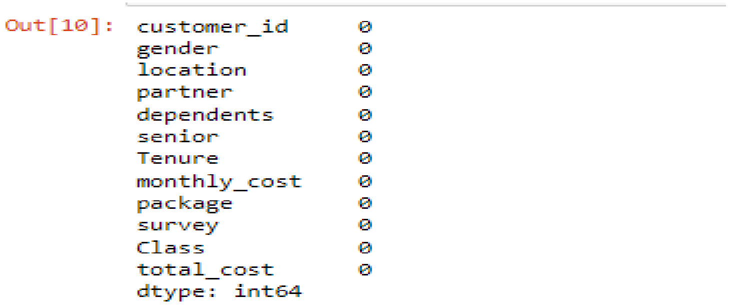
Several factors that are affect churn attrition like poor service, price isn’t right, offer incentives, and based on contracts etc.

5)Data pre-processing

I. **Dealing with outliers**
Regardless of how cautious you are while collecting data, dealing with outliers in machine learning may be difficult. Every data scientist has experienced this. A data point that sticks out among the rest is called an outlier. They highlight measurement mistakes, poor data collection, or simply show variables that were not considered during data gathering. In our dataset, outliers are present. These are dropped in our dataset.

II. **Dealing with missing values**
In the given dataset, missing values are present in Class and null values in monthly_cost. The missing values present in the Class feature are removed whereas null values in the

monthly_cost column are replaced with values based on the package. These details are mentioned above(Data cleaning). After dealing with missing values, there are no missing values in our dataset.



```
Out[10]: customer_id      0
         gender           0
         location         0
         partner          0
         dependents       0
         senior           0
         Tenure           0
         monthly_cost     0
         package          0
         survey           0
         Class            0
         total_cost       0
         dtype: int64
```

III. Dealing with different scales of data

Normalization:

Normalization is a technique which is applied to data often in machine learning for data preparation. After exploring and cleaning the dataset. We need to normalize the data because it can improve the accuracy of models without distorting the values of data that should be in the range of 0 and 1. Normalization aims to remove the data redundancy then the data will be more flexible.

Normalization of variable(y) = $(x-x_{min})/(x_{max}-x_{min})$

Here, we used the minmaxscaler() method to normalize the data. Then rescale the values of the data using the minmaxscaler method in python.

Sta

IV. Feature selection

Feature selection means selecting the subset of input features from the dataset. From the input features, we must select features which are used to predict the target feature. Here, 'gender', 'partner', 'senior', 'Tenure', 'monthly_cost', 'total_cost', and 'survey' these features are chosen to predict the customer churn.

6)Build and test models

i. Logistic regression

Logistic regression is an example of a supervised learning model, which is used for the prediction of the occurrence probability of a binary event. In logistic regression, the predicted outcome should be binary. Here, it is used to predict whether the customer is churned or not. It is also a kind of binary classification problem.

Splitting the dataset

After cleaning the data, we will apply models and algorithms to it. In case, if the dataset is split into 80:20 ratio then 80% of the data is to be trained and 20% of the data is to be tested. From the dataset, 5718 observations will be trained, and the remaining are to be tested. Normalize the data as mentioned above. Then evaluate the model. The accuracy of the logistic model is **84.69%.**

ii. K-Nearest Neighbor model

K-Nearest Neighbor is a supervised learning model. It is used for both classification and regression problems. But mostly used for classification problems. The KNN method simply saves the information during the training stage, and when it receives new data, it organizes it into a category that is quite like the new data. So, the model will be evaluated and then predict the churn information. The accuracy of KNN is **84.90%.**

iii. Decision Tree model

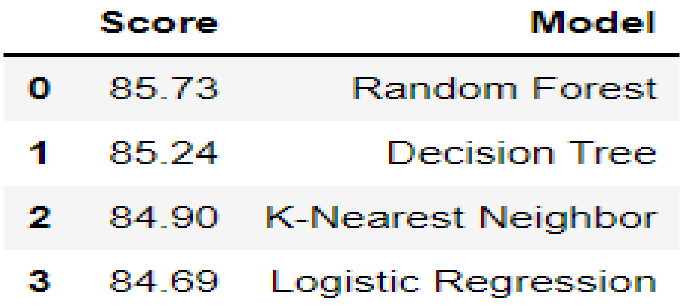
Decision Tree is a non-parametric supervised learning model, which is used for both classification and regression models. It looks like a tree. In our dataset, the decision tree is used to predict binary information i.e., churned or not. The accuracy of the decision tree model is **85.24%.**

iv. Random Forest Classifier

A random forest is a machine learning approach for tackling classification and regression issues. It makes use of supervised learning, a method for solving complicated issues by combining several classifiers. An algorithm called a random forest uses several decision trees. So, it follows the decision tree algorithm but more than one. The accuracy of the random forest classifier is **85.73%.** This model predicts the highest accuracy as compared to other models.

7)Conclusion

Here, we used four models to predict customer churn. They are Logistic regression, KNN, Decision tree, and Random Forest classifier. By splitting the dataset into training and testing. Models are trained and then predict the customer churn. Every model gave an accuracy of nearly 85%.



	Score	Model
0	85.73	Random Forest
1	85.24	Decision Tree
2	84.90	K-Nearest Neighbor
3	84.69	Logistic Regression

From the above table, we can tell Random Forest Classifier is the best ML model to predict customer churn. The accuracy is based on how you split your dataset, what features are you considering and how you are training your model. So, based on these parameters the accuracy of the model may vary and predict the churn information.

Comment on the initial hypothesis

As mentioned above about the initial hypothesis, here we got an accuracy of 85%. So, the initial hypothesis is correct. So, from this, we can conclude that customer churn depends on these parameters which are considered in the model.

References:

1. https://neptune.ai/blog/how-to-implement-customer-churn-prediction
2. https://towardsdatascience.com/predict-customer-churn-the-right-way-using-pycaret-8ba6541608ac#:~:text=One%20of%20the%20ways%20to,churn%20rate%20is%205%20percent.
3. Guide to Churn Prediction: Part 1 — Gather & Clean | by Jaanvi | Mage
4. Guide to Churn Prediction: Part 3 — Descriptive statistical analysis | Mage Blog
5. Guide to Churn Prediction: Part 4 — Graphical analysis | Mage Blog
6. https://www.analyticsvidhya.com/blog/2021/08/churn-prediction-commercial-use-of-data-science/

Appendix(Program):

```
#import libraries
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

#Load dataset into python
d1 =pd.read_csv('Group 1.csv')
print(d1.shape)
d1.head()
#remove unused index
d1 = pd.read_csv('Group 1.csv', index_col=[0])
d1.isna().sum()
print(d1.dtypes)
#data cleaning
package_s = {1:'26',2:'34',3:'40',4:'44'}
d1['monthly_cost'] = d1['package'].map(package_s)
d1['monthly_cost'] = d1['monthly_cost'].astype(float)
d1['total_cost'] = d1['monthly_cost']*d1['Tenure']
d1.drop(d1.index[d1['Tenure'] < 0], inplace=True)
print(d1.head(91))
d1.shape

d1["Class"] = d1["Class"].map({'Churn=No':0,'Churn=Yes':1})
d1["gender"] = d1["gender"].map({'Female':0,'Male':1})
d1['survey'] = d1['survey'].str.replace('No reply', '')
d1['dependents'] = d1['dependents'].str.replace('Unknown', '')
d1['survey']=np.where(d1['survey']>='5',1,0)
d1['survey'] = d1['survey'].astype(float)
print(d1.head(20))
d1.shape
```

```
d1= d1.dropna(subset=['Class'])
print(d1.head(20))
d1.shape

d1.isna().sum()
print(d1.dtypes)
#after cleaning the dataset
d1=d1.reindex(columns=['customer_id','location','package','dependents','gender','partner','senior', 'Tenure','monthly_cost', 'total_cost','survey', 'Class'])
print(d1.head(5))

d1_num = d1[['gender','partner','senior', 'Tenure','monthly_cost', 'total_cost','survey', 'Class']]
d1_cat = d1[['customer_id', 'location','package', 'dependents']]
d1_num.head(5)

print(d1_num.describe())

print(d1_cat.describe(exclude=['int64','float64']))

d1_num.hist(figsize=(10, 8), bins=50, xlabelsize=5, ylabelsize=5);

d1['Class'].value_counts().plot(kind = 'bar').set_title('Churn')
plt.xlabel('Class(if churn yes =1 & no =0)')
plt.ylabel('count')

sns.countplot(x='gender',hue='Class',data=d1)
plt.xlabel('gender(female = 0 & male = 1)')
plt.ylabel('count')
plt.title('plot for both gender and churn(Class)')
plt.show()

sns.countplot(x='Class',data=d1, hue='package')
plt.xlabel('plot for churn based on package')
tenure_plot = sns.histplot(x = 'Tenure', hue = 'Class', data = d1, multiple='dodge')
tenure_plot.set(xlabel="Tenure(No.of years customer stayed)", ylabel = "count")

tenure_plot = sns.histplot(x = 'total_cost', hue = 'Class', data = d1, multiple='dodge')
tenure_plot.set(xlabel="total_cost", ylabel = "count")
sns.histplot(x='monthly_cost',hue='Class',data=d1,multiple='dodge')
corr = d1.corr()
sns.heatmap(corr, xticklabels=corr.columns, yticklabels=corr.columns)
#feature selection
X = d1.iloc[:,5:11]
Y = d1.iloc[:,-1]
```

```
#splitting dataset
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size=0.2, random_state=0)
print(x_train.shape)
#normalizing the dataset
from sklearn.preprocessing import MinMaxScaler
mms = MinMaxScaler()
x_train_norm = mms.fit_transform(x_train)
x_test_norm = mms.transform(x_test)

from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.linear_model import LinearRegression
from sklearn import metrics
Regressor= LogisticRegression(random_state = 50)
Regressor.fit(x_train_norm,y_train)
loc_pred = Regressor.predict(x_test_norm)
loc_accuracy = round(metrics.accuracy_score(y_test, loc_pred)*100,2)

from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier

knn_model = KNeighborsClassifier(n_neighbors = 5,metric ='minkowski', p=2)
knn_model.fit(x_train_norm,y_train)
knn_pred = knn_model.predict(x_test_norm)
knn_accuracy = round(metrics.accuracy_score(y_test, knn_pred)*100,2)

decisiontree_model = DecisionTreeClassifier(criterion = 'gini', random_state = 50)
decisiontree_model.fit(x_train_norm,y_train)
decisiontree_model_pred = decisiontree_model.predict(x_test_norm)
decisiontree_model_accuracy=round(metrics.accuracy_score(y_test,decisiontree_model_pred)*100,2)

rf_model = RandomForestClassifier(n_estimators = 100, criterion = 'entropy', random_state = 0)
rf_model.fit(x_train_norm,y_train)
rf_model_pred = rf_model.predict(x_test_norm)
rf_mode_accuracy = round(metrics.accuracy_score(y_test, rf_model_pred)*100,2)

model_comparison=pd.DataFrame({'Model':['LogisticRegression','K-Nearest Neighbor','DecisionTree','RandomForest'],'Score':[loc_accuracy,knn_accuracy,decisiontree_model_accuracy,rf_mode_accuracy]})
model_comparison_d1 = model_comparison.sort_values(by = 'Score', ascending = False)
model_comparison_d1 = model_comparison_d1.set_index('Score')
```

```
model_comparison_d1.reset_index()
#confusion matrix
from sklearn.metrics import confusion_matrix
confus_rf = confusion_matrix(y_test,rf_model_pred)
confus_rf
```