Telecom Services Customer Churn

Final Project Report

Group 68

Achala Shenoy
Sakshi Gujarathi

857 263 1750 (Tel: Student 1)
857 222 1850 (Tel: Student 2)

shenoy.ac@northeastern.edu
gujarathi.sa@northeastern.edu

Percentage of Effort Contributed by Student1: 50%
Percentage of Effort Contributed by Student2: 50%

Signature of Student 1:

Signature of Student 2:

Submission Date: 04/21/2023

Problem Setting:

With the rise in technology and internet connectivity, there has been a decrease in the demand for landline and cable connections. To track this shift from telephones to mobile phones and from cable television to streaming services, we have chosen this sample data module which monitors the customer attrition of a hypothetical telecom company based on several variables. These variables or columns include information about each customer's service preferences, gender, dependents, and monthly costs. When a company's customers discontinue doing business with it, this is referred to as customer churn. This information is recorded in the churn column in case the client has unsubscribed to the services during the past month. Because it is much less expensive to retain an existing customer than to acquire a new one, businesses place a high priority on analyzing turnover and provide customized deals to retain their customers.

Problem Definition:

The intention of this analysis is to identify the best classification model and predictors for correctly predicting customer churn instances, which could eventually be used in tracking the changing demand of consumers at all stages and thereby changing business strategies accordingly. Good customer service and merchandise can help you retain customers. But getting to know a client well is the best approach for a business to stop customer attrition. Churn prediction models can be created using the enormous amounts of customer data that have been collected. Knowing which customers are most likely to leave allows a business to concentrate its marketing efforts on that segment of its clientele. We aim to segment the customers based on their personas, age groups etc. to identify and fill in the business gaps at all stages.

Data Source:

This telecom customer dataset has been taken from Kaggle, an open source data repository.
Dataset: Telco customer churn: IBM dataset

Data Description:

The dataset consists of 7043 observations (rows) and 33 attributes (columns).
Few attributes that we are working with:

- **CustomerID -** A unique entity to identify each customer. (Numerical)
- **Count**: A value used to sum up total number of customers.(Numerical)
- **State**: The state customer is from.
- **City**: The city customer is from.
- **Zip Code**: The zip code of the customer's primary address. (Numerical)

- **Gender**: The gender of customer (M/F) (Categorical : Male, Female)
- **Age**: Age group of the people (Y/N) (Categorical : Children, Adults, Senior Citizens)
- **Multiple Lines:** To check if the customer is subscribed to multiple lines with the company:(Y/N)
- **Streaming TV:** To check if the customer uses their Internet service to stream television programing from a third party provider: (Y/N)(Categorical : Movies, Series, Music Videos)
- **Churn Score:** A value from 0-100 that is calculated using predictive models. The higher the score, the more likely the customer will churn. (Numerical : Score on scale of 100)

We have values which can predicted using various prediction models. A predictive model predicts a specific future result based on trends discovered through historical data. In this case, we will train the data model that will recognize specific patterns to predict outcomes, such as future sales, customer loss, and few other values by gathering and analyzing historical data. We will use correlation matrix to show the differences in the data. We will use the prediction module to deduce the *Churn_Score*. We will create a training data set to test the imbalance between the current data and the previous data.

Explatory Data set:

Index	Data	Description	Values
1	Customer ID	Unique ID for every customer	Number
2	Gender	Whether the customer is a male or a female	String
3	Senior Citizen	Whether the customer is a senior citizen or not	Number
4	Partner	Whether the customer has a partner or not	Yes / No
5	Dependents	Whether the customer has dependents or not	Yes / No
6	Tenure	Number of months the customer has stayed with the company	Number
7	Phone Service	Whether the customer has a phone service or not	Yes / No
8	Multiple Lines	Whether the customer has multiple lines or not	Yes / No
9	Online Security	Customer’s internet service provider	Yes / No
10	Online Backup	Whether the customer has online security or not	Yes / No
11	Device Protection	Whether the customer has device protection or not	Yes / No
12	Tech Support	Whether the customer has tech support or not	Yes / No
13	Streaming TV	Whether the customer has streaming TV or not	Yes / No
14	Streaming Movies	Whether the customer has streaming movies or not	Yes / No
15	Contract	The contract term of the customer	Month / One-Year / Two-Year
16	Paperless Billing	Whether the customer has paperless billing or not	Yes / No
17	Payment Method	The customer’s payment method	String
18	Monthly Charges	The amount charged to the customer monthly	Numbers
19	Total Charges	The total amount charged to the customer	Numbers

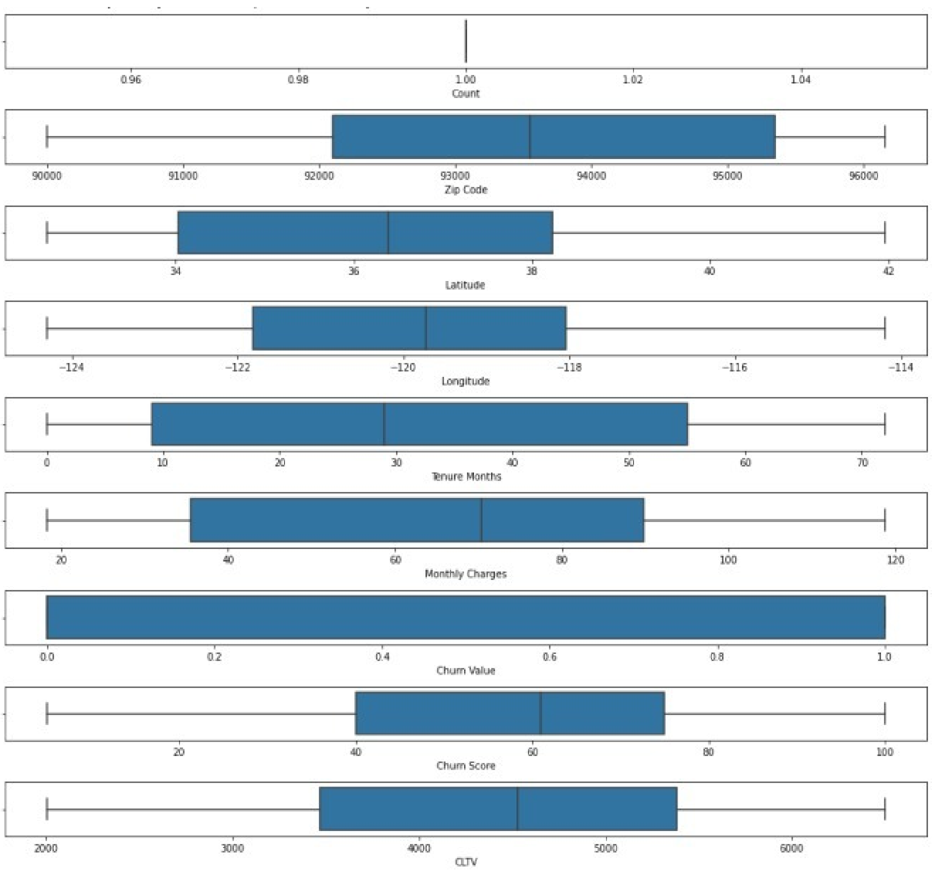
Data Mining Tasks

1. Data Understanding: Initially we have to determine the target variable, in this situation the target variable is whether or not a client would churn, which is the variable we want to predict. By examining previous churn data or speaking with company stakeholders, we will be able to determine the target variable. Later we have to recognize the sources of the data.Telecom company has access to a wealth of information on their customers, including demographics, usage trends, call detail records, billing information, and network performance information. We should make sure we are using trustworthy and accurate data, we should check the data sources and it's quality. After exploring the data, we got the features that are important for predicting churn. Customer requirement for TV channel or movies, usage trends, and information on network performance are examples of pertinent aspects. Missing data can significantly affect the predictive model's accuracy. We searched for the missing data and properly managed it. The accuracy of the model is also be impacted by inconsistent data, hence data cleaning is done to make sure it is clean and consistent. We have **7043** observations with **33** variables. We had **1869** nan values, which we cleaned, to avoid inconsistency of the data.

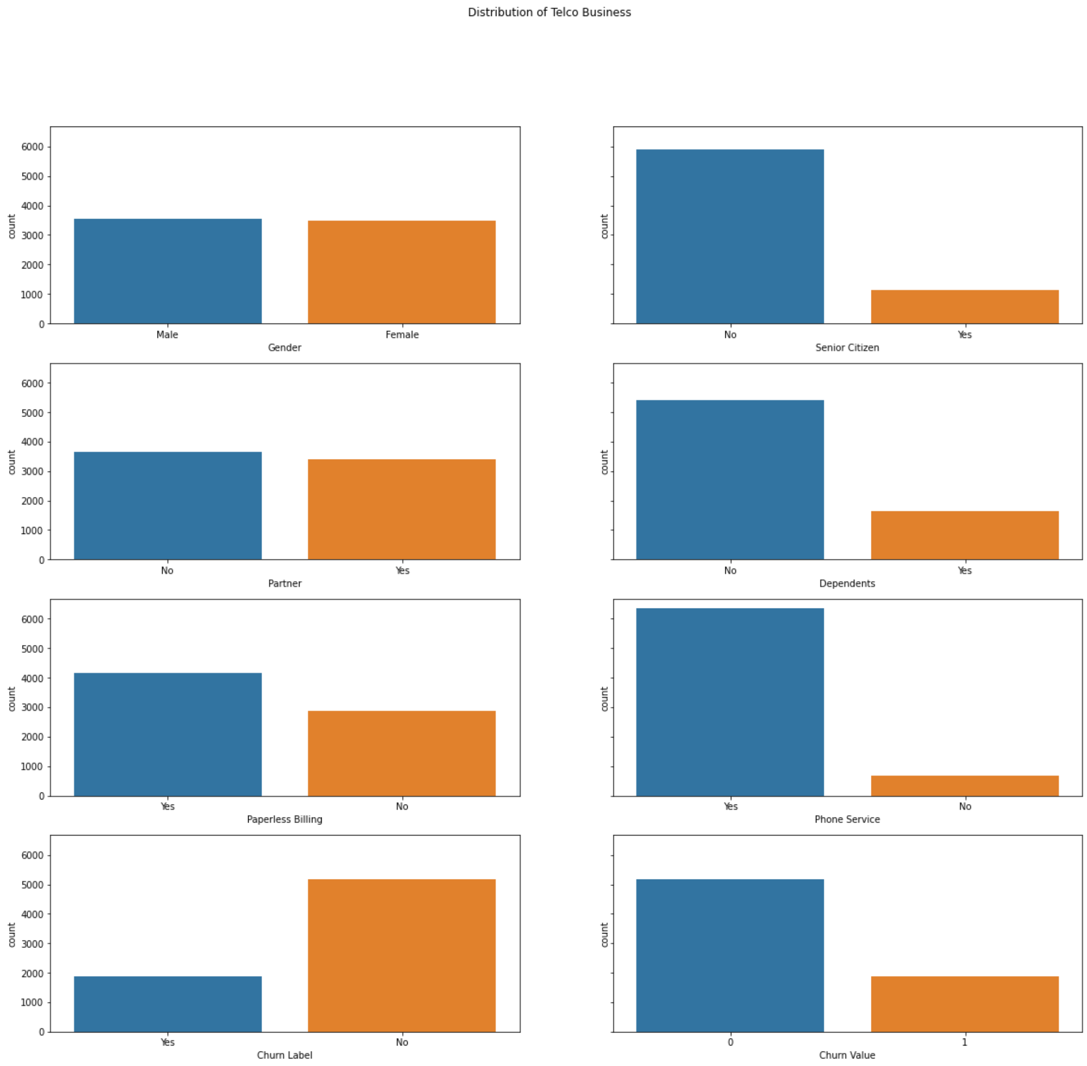
2. Data Pre-processing: Cleaning the data, dealing with missing values, outliers, and inconsistent data is the first step in data pre-processing. We used techniques like mean imputation, median imputation basis the skewness of data. We also plotted box plots and scatter plots to find outliers, which can then be eliminated or modified but didn't find any in our case. We created the below box plot to check for the potential outliers. In order to be used in the model, categorical variables such as customer type or subscription plan and the other services offered that consisted of yes and no binary values were encoded to numerical binary values using label encoding.

Outliers can be troublesome for this telecom churn dataset because they might have a disproportionate impact on statistical estimations like means and variances. The average monthly prices for the entire dataset, for instance, may be artificially inflated if a tiny percentage of consumers have excessively high monthly expenses. Due to their tendency to provide inaccurate summaries and visualizations, outliers can also make it challenging to identify patterns or links in the data.

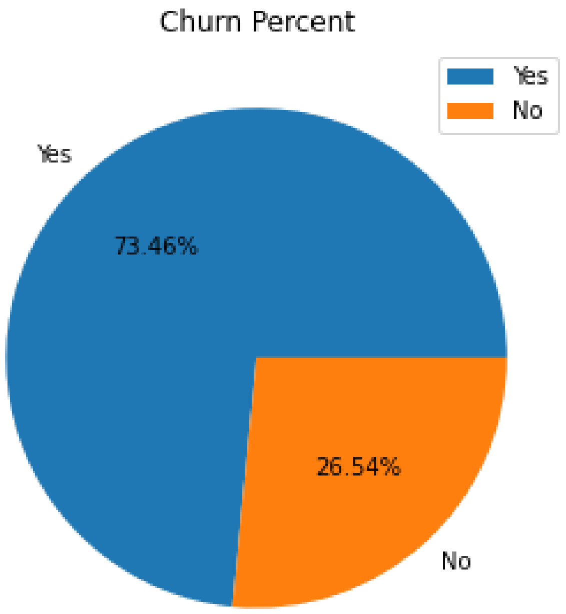
Among the numerous methods for finding outliers in a dataset and handling them to see how the data are distributed and spot any observations that are significantly outside of the expected range, here we have used boxplot to show our visualization.



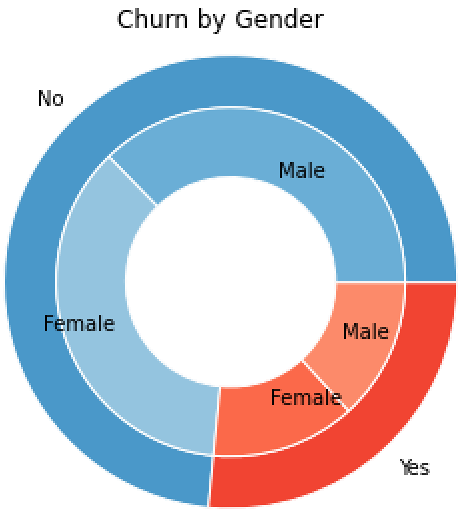
Categorical variables can be any variables that indicate non-numeric values in a telecom churn dataset, including customer type, subscription plan, device type, and region. Examples of categorical variables you could find in a telecom churn dataset are shown below: Customer type: This parameter identifies if a client is a new or returning client. This variable, which can be either prepaid or postpaid, describes the kind of subscription plan a customer has. Device type: This variable identifies the kind of device a user is utilizing, such as a tablet or a smartphone. Region: The location, such as a city or a state, where a customer resides is indicated by this variable. Considering that our dataset consists of many categorical binary variables, we also tried to reduce dimensions by pivoting the columns and finding correlation between various columns to consider elimination of certain columns. After which we plotted certain categorical variables to map the differences between their counts.



From the given visualization we can see the categorical values like Gender, Age group, Churn Value and few other elements which can be plotted to find the individual count of each binary category.



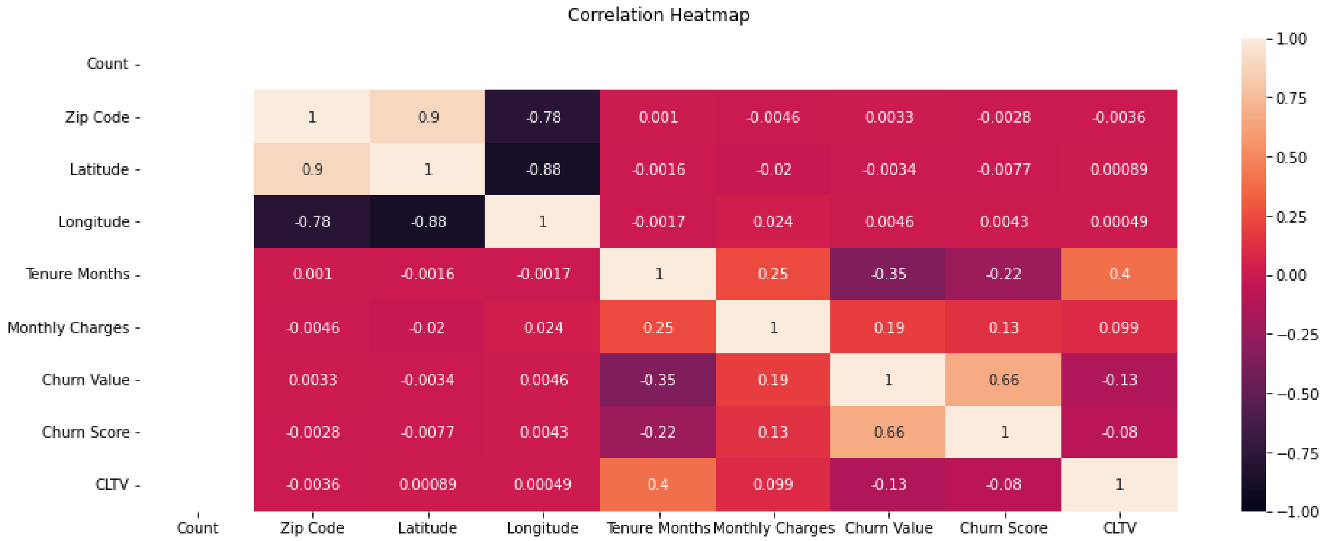
We have a pie chart to show the percentage of people who are part of churn and who aren’t. From the given pie chart we can say that the churn percent is greater and there are many people who are switching. This gives us an idea of what percentage of people are switching the services. The given visualization gives us the split of churn values based on gender.

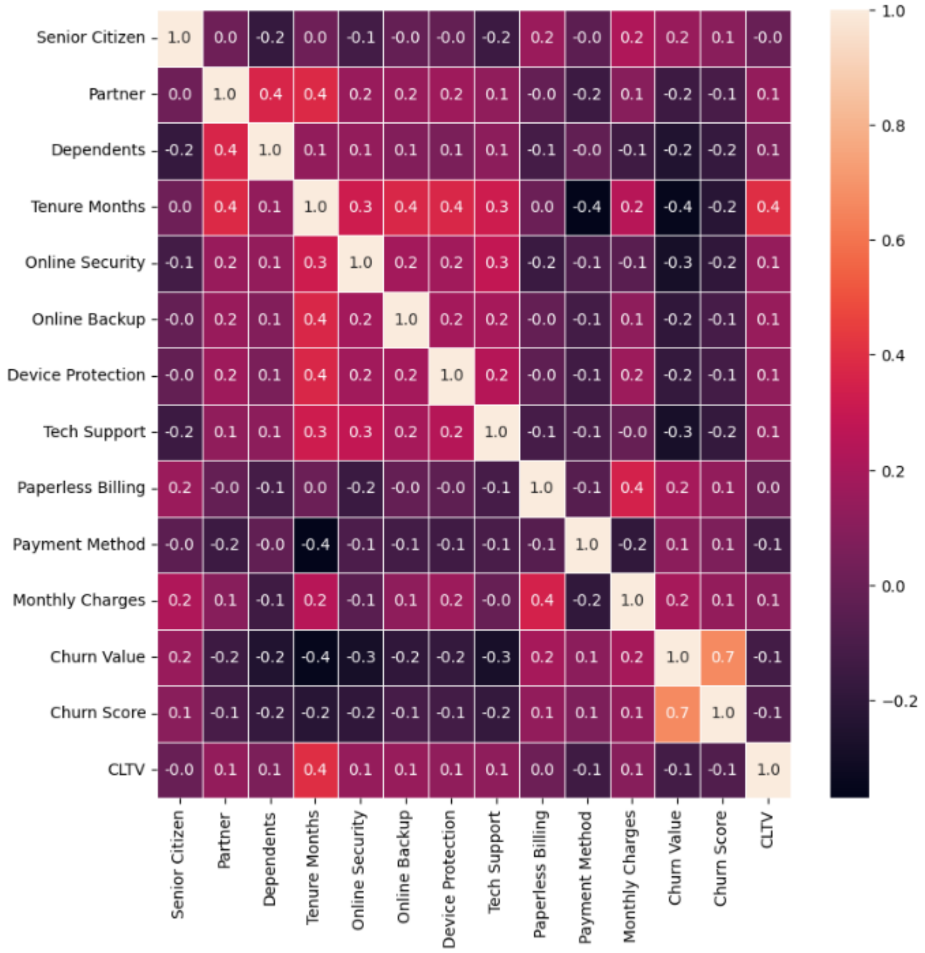


Numerical Variables:

Correlation Analysis

For correlation analysis, we first load the telecom churn dataset. We then computed the correlation matrix using the **corr** function from pandas, which computes the pairwise correlation coefficients between all pairs of features in the data. The resulting plot shows the correlation coefficients between pairs of features, with positive correlations in red and negative correlations in purple. Features with high positive correlations (close to 1) or high negative correlations (close to -1) may be indicative of collinearity issues and may need to be removed or combined in some way before building a predictive model. Additionally, features with high correlations with the target variable (Churn) may be good predictors of the target and should be included in the model. Here, we have used a correlation heatmap to visualize our findings.

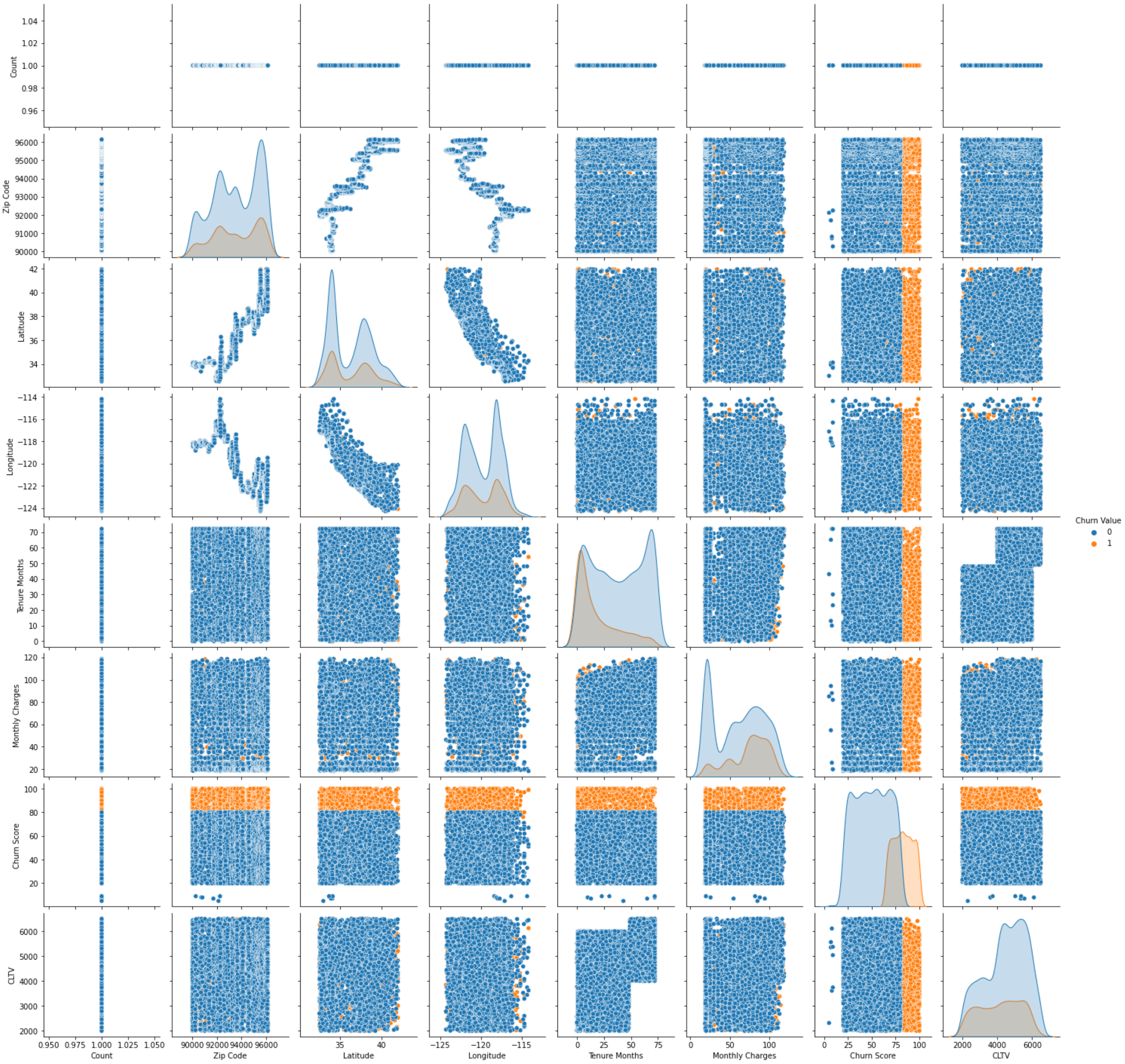




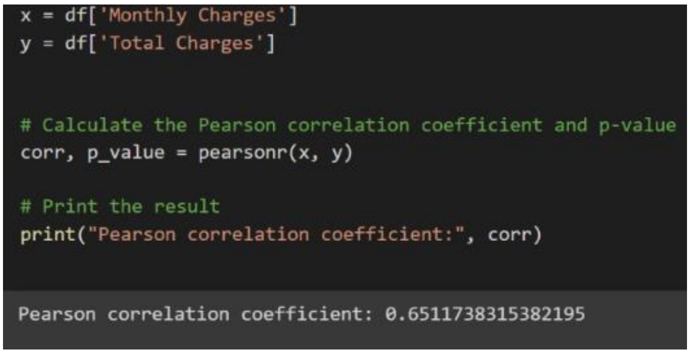
There are many other visualization through which we could show relation of our data. Few of the visualization are shown below:

Pair Plot:

The pairwise correlations between various features in a dataset are displayed via pair plots, sometimes referred to as scatter plot matrix. A histogram of the distribution of each feature is displayed on the diagonal of a pair plot, which compares each pair of features to one another in a scatter plot. A pair plot can be an effective tool for examining the correlations between various parameters and their possible predictive value for the target variable in the context of a telecom churn dataset (i.e., whether a customer will churn or not). For instance, we might search for connections between consumption patterns (such as monthly fees or overall expenses) and churn or between client demographics (such as age or gender) and churn. Features in a pair plot will typically have a linear relationship in the scatter plot if they are substantially associated with one another. Uncorrelated characteristics will manifest as a haphazard spread of points. Also, we were able to spot groups or trends in the data that are connected to certain churn rates.



The degree and direction of the linear link between two pairs of numerical variables can be determined using the Pearson's correlation coefficient for the telecom churn dataset. The generated correlation matrix can be used to distinguish between variables that have strong and weak correlations. Following this, we conducted Pearson analysis to get the correlation coefficient between ‘Monthly Charges’ and ‘Total Charges’ to check if we can drop any of the two columns.

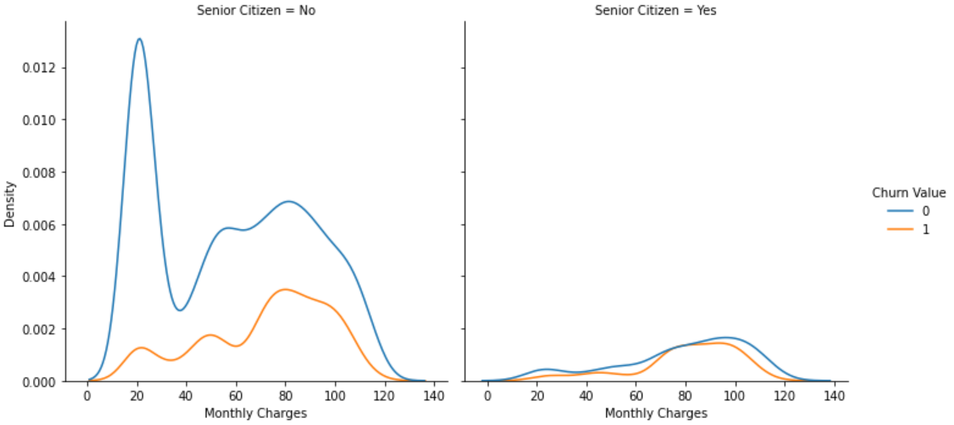


```
x = df['Monthly Charges']
y = df['Total Charges']


# Calculate the Pearson correlation coefficient and p-value
corr, p_value = pearsonr(x, y)

# Print the result
print("Pearson correlation coefficient:", corr)
```

```
Pearson correlation coefficient: 0.6511738315382195
```

DisPlot:

A displot is a sort of visualization that displays how a single variable is distributed over a dataset. It combines a kernel density estimate (KDE) graphic and a histogram. In this displot, the histogram displays the variable's frequency distribution, while the KDE plot displays the variable's estimated probability density function. A displot can be an effective tool for examining the distribution of a particular feature and its relationship to churn in the context of a telecom churn dataset. To show how the distribution of monthly charges differs between customers who churn and those who don't, you may, for instance, make a display of the Monthly Charges feature.

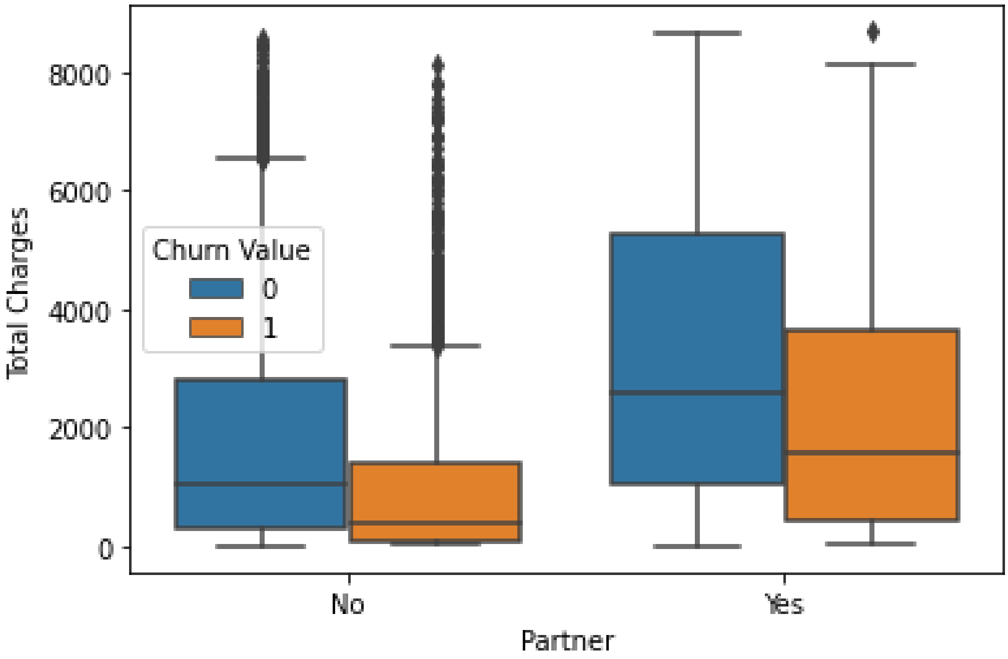


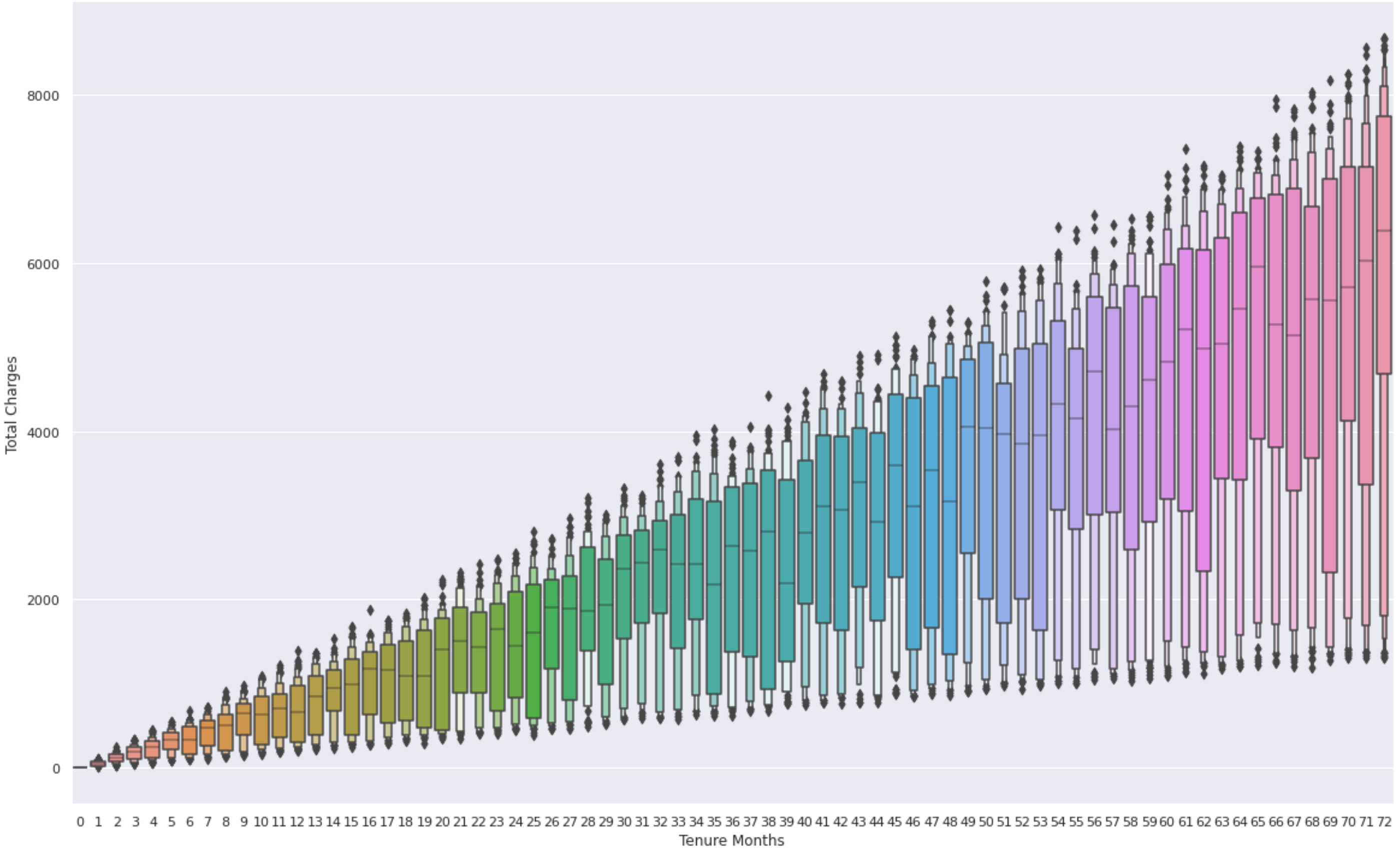
The resulting plot shows the distribution of monthly charges, with the histogram and KDE plot colored by the level of churn. From this we can concur that the monthly charges heavily depend on the age demographic of the customers. This can be a useful way to visualize the relationship between a single feature and the target variable.

Box Plot:

A box plot is a type of visualization that shows the distribution of a numeric variable in a dataset. In a box plot, the box represents the interquartile range (IQR) of the data, with the median represented by a horizontal line inside the box. Points outside this range are plotted as individual dots, representing "outliers". Here, for the telecom churn dataset, a box plot can be a useful way to explore the distribution of a numeric feature and how it relates to churn. For example, we have created a box plot of the Total

Charges feature to see how the distribution of total charges varies between customers who churn and those who don't.





This final plot provided a box and whisker plot of the total charges, with distinct boxes for churning and non-churning customers. This can be a practical technique to compare the distribution of a numerical feature across various dataset groupings.

We have a better knowledge of the telecom churn dataset now that it has been preprocessed, and it is ready for analysis. Following are some significant findings and inferences from the preprocessing procedures:

The dataset includes 30 features, including both numerical and categorical variables, and 7,043 observations of telecom subscribers.

The dataset contained some missing values, which we managed by either removing the missing values or by imputed them with suitable values.

In order to prepare the categorical variables for analysis, we also used label encoding yo convert them into numerical ones.

To learn more about the dataset, we developed visualizations like histograms, pair plots, and correlation matrices.

Overall, the telecom churn dataset has been successfully preprocessed in order to prepare it for further data mining models. The subsequent phases can include developing a prediction model with methods like logistic regression, decision trees, or neural networks to pinpoint clients at danger of leaving.

Data Preparation:

Here we used sklearn library's standardization method MinMaxScalar() function to bring columns like Tenure Months, Monthly Charges and Churn Values in a normal distribution so that they don't dominate due to their large values over other features and are centred around zero and have the same order of variance. This will help the estimators to correctly learn from other predictors/columns.

Data Partitioning:

Predictor variables are denoted by 'X' and our target variable 'Churn Value' is represented by 'Y'. We split the dataset into 80% for training and 20% for testing.

Data Mining Models:

Customer churn is a common problem faced by telecommunications companies, and there are several types of predictive models that can be used to address this problem. Some of the most used models that we have implemented include Logistic Regression, XGB Classifier, Random Forest, Neural Networks and Decision Tree

For prediction of our target column, 'Churn Value' we used 5 algorithms/models. The models we tested and trained our data are elaborated below:

1. Logistic regression: This is a popular model used for predicting binary outcomes, such as whether a customer will churn or not. Logistic regression is easy to interpret and can provide insights into the factors that are most strongly associated with churn. Logistic regression is a statistical model used for classification using the probabilities of output variable belonging to each class and then deciding basis the threshold or the cut-off value. Goal of logistic regression is to estimate the probability of a customer churning based on one or more input variables, such as customer demographics, usage patterns, and service plan information.

Advantages:

a. It is computationally fast and hence cheap even for a large dataset
b. t is a simple statistical model which is easy to interpret the nature of predictors

Disadvantages:

a. It is sensitive to outliers which in turn affects the coefficients and hence the predictions.
b. Can’t be used for continuous outcomes
c. Requires a large sample size to gain stable and results

2. XGB Classifier

It is an ML classification algorithm that uses boosting for regression, classification and ranking tasks. The XGB Classifier consists of a collection of decision trees that work together to make predictions. Each decision tree is trained to predict whether a customer will churn or not based on a subset of the features available in the dataset. During training, the algorithm learns to adjust the weights of each decision tree to minimize the prediction error. The final prediction is made by combining the predictions of all the decision trees in the ensemble.

Advantages:

a. It is fast and efficient. This speed makes it ideal for real time datasets that keep on growing
b. It had good accuracy and great predictive power
c. Can handle missing data
d. Its built-in feature can help to find the best features that can be considered for prediction

Disadvantages:

a. It is susceptible to overfitting though can be avoided using cross validation techniques
b. Can be memory intensive

3. Random Forest:

A group of decision trees are trained using various subsets of the training data and a random subset of the feature sets make up the Random Forest algorithm. It does this by constructing many trees and selecting the predicted class basis the majority outputs of various trees. In order to reduce the prediction error, the algorithm learns to modify the weights of each decision tree. The forecasts of each decision tree in the ensemble are combined to get the final prediction.

Advantages:

a. Doesn’t overfit with a lot of features
b. Generally, has a high accuracy
c. Also gives an estimate of important variables

Disadvantages:

a. It has noisy data
b. It has several hyperparameters that need to be tuned to optimize its performance
c. If the training data has imbalanced classes, it may produce biased output

4. Decision Tree:

The branches of a decision tree represent various outcomes that could result from the decision at each node, which is based on an input variable. The model aims to identify consumers who are likely to churn from those who are not by dividing the data into homogeneous categories.

Advantages:

a. It is easy to interpret as it provides a visual representation of the decision-making process,
b. It can handle both categorical and continuous variables
c. It can handle missing data when dealing with incomplete datasets.
d. It can provide insights into feature importance based on the frequency of use in the tree.

Disadvantages:

1. It tends to overfit the data which leads to poor performance on new data.
2. It can be unstable and hard to replicate.
3. It can be biased towards variables with many categories even if those splits are not statistically significant.
4. It might not capture complex interactions which may not capture complex interactions between variables that affect churn.

5. Neural networks:

A type of machine learning methods known as neural networks is particularly helpful for difficult, non-linear tasks like predicting teleco turnover. Layers of connected nodes, commonly referred to as neurons, make up neural networks, which are capable of learning to represent intricate connections between input and output data.

Advantages:

a. It consists of non-linear relationships which makes them particularly suitable for complex classification problems like teleco churn prediction.
b. It is robust to noise and missing data, and it can effectively handle input data that is not well-defined or contains errors.
c. It performs feature engineering to learn relevant features from raw input data
d. It is versatile and can be used for a wide range of classification tasks

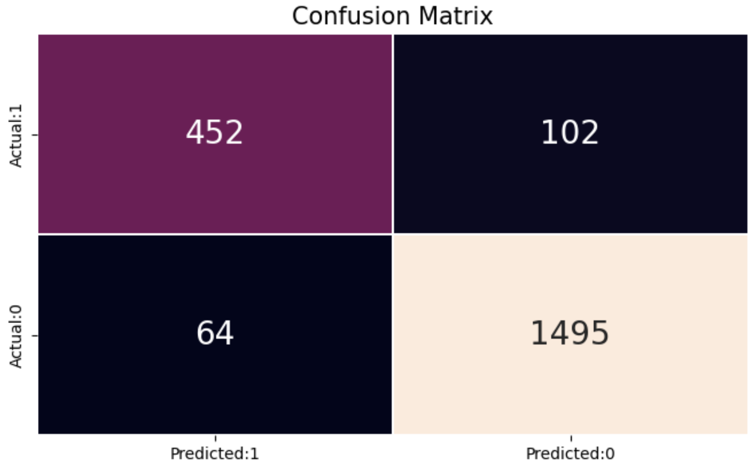
Disadvantages:

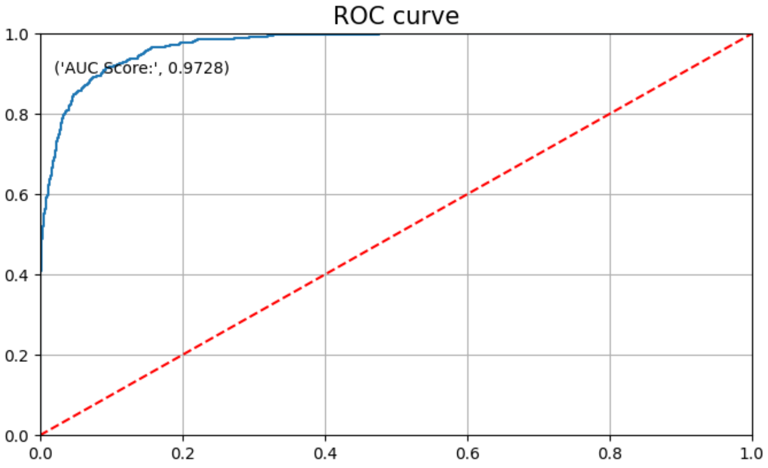
a. It can be computationally intensive and require a lot of resources.
b. It tends to be overfitting when the number of layers and neurons in the network is large.
c. It can be difficult to interpret and understand, especially when they are large and complex.
d. They have many hyperparameters that need to be optimized, which can make it difficult to find the best set of hyperparameters.

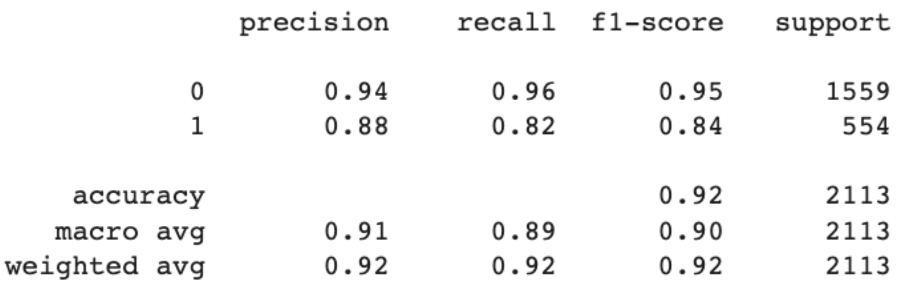
Performance Evaluation:

1) Logistic Regression:

Testing Data Coefficient Matrix:







```
              precision    recall  f1-score   support

           0       0.94      0.96      0.95      1559
           1       0.88      0.82      0.84       554

    accuracy                           0.92      2113
   macro avg       0.91      0.89      0.90      2113
weighted avg       0.92      0.92      0.92      2113
```

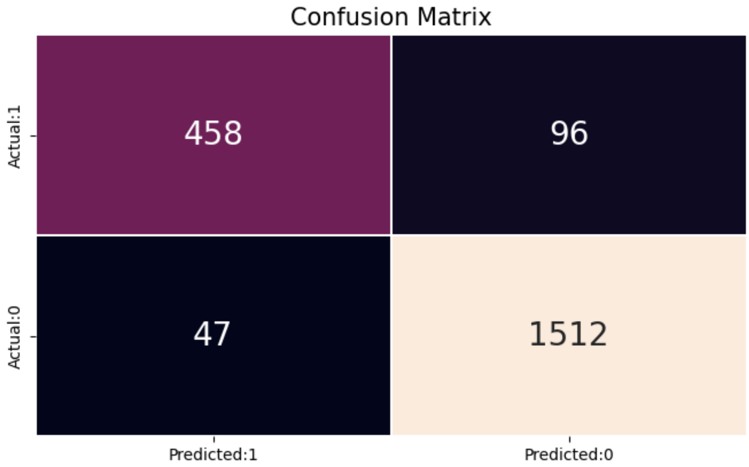
AUC of 0.9728 indicates that the model has good discriminatory power in distinguishing between positive and negative samples.

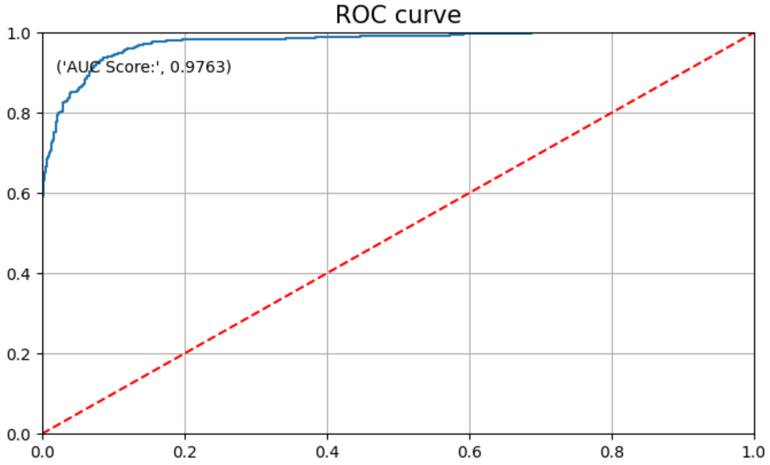
Typically, AUC values range from 0.5 to 1, where 0.5 represents random chance (no better than guessing) and 1 represents a perfect classifier. Therefore, an AUC of 0.9728 suggests that the model is performing better than random chance, but it may not be very accurate or precise.

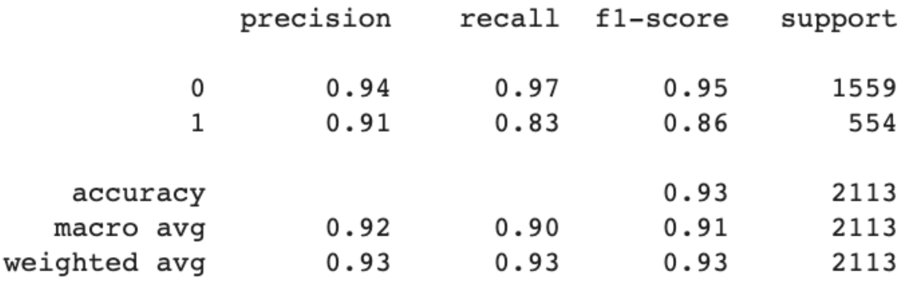
- The precision for Retain that is customer churn is relatively high, which means that when the model predicts a customer will be retained, it is correct 94% of the time. However, the precision for Churn is lower, meaning that when the model predicts a customer will churn, it is correct only 88% of the time.

- The recall for Retain is also relatively high, meaning that the model is able to correctly identify 96% of customers who will retain. The recall for Churn is much higher, meaning that the model is able to correctly identify 88% of customers who will churn.
- The F1-scores for both Retain and Churn are moderate, indicating that the model is able to achieve a balance between precision and recall for both classes.
- The weighted average of precision, recall, and F1-score is also provided, which takes into account the imbalance in the class distribution. The weighted average of precision is 0.82, indicating that the model is correct on 92% of its predictions on average. The weighted average of recall is 0.92, meaning that the model is able to identify 92% of customers who will actually churn or retain. The weighted average of F1-score is 0.92, which indicates that the model is able to achieve an overall balance between precision and recall.

2) Random Forest:







```
              precision    recall  f1-score   support

           0       0.94      0.97      0.95      1559
           1       0.91      0.83      0.86       554

    accuracy                           0.93      2113
   macro avg       0.92      0.90      0.91      2113
weighted avg       0.93      0.93      0.93      2113
```

AUC of 0.9763 indicates that the model has good discriminatory power in distinguishing between positive and negative samples.

This confusion matrix and the associated metrics are evaluating the performance of a Random Forest classifier in predicting whether customers will churn or not.

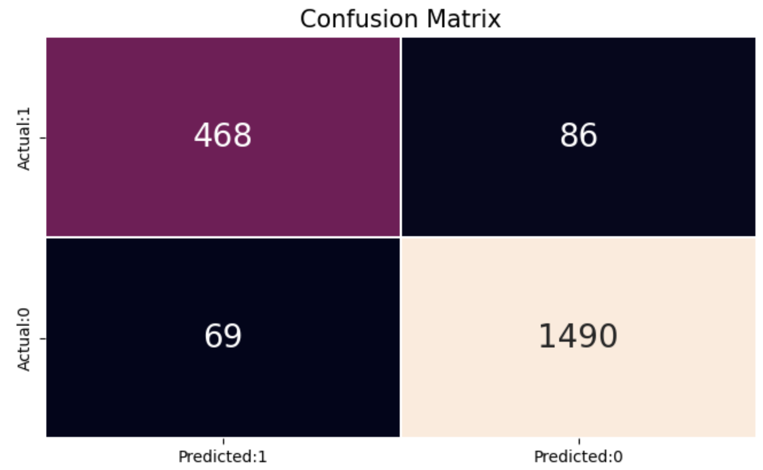
The confusion matrix shows that out of 2113 customers, 458 were correctly predicted to be retained, while 47 were predicted to be retained but actually churned. Similarly, 1512 were correctly predicted to churn, while 96 were predicted to churn but actually retained.

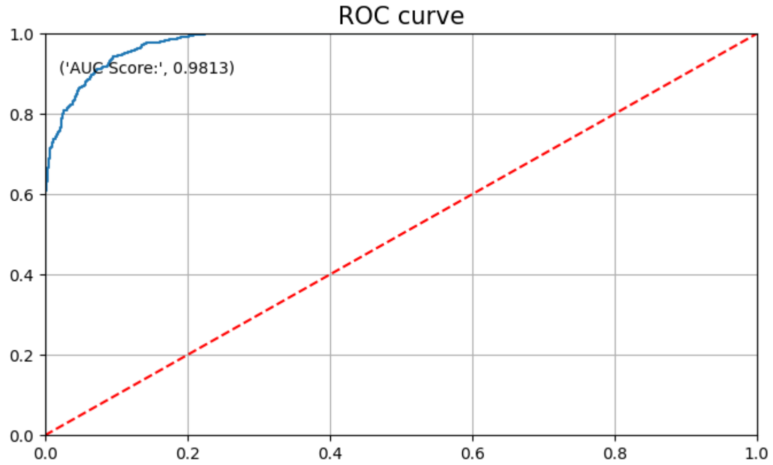
The precision for predicting customers who will churn is 0.51, which means that when the model predicts a customer will churn, it is correct 91% of the time. The recall for predicting customers who will churn is 0.83, which means that the model correctly identifies 83% of customers who actually churned. The precision for predicting customers who will be retained is 0.94, which means that when the model predicts a customer will be retained, it is correct 94% of the time. The recall for predicting customers who will be retained is 0.97, which means that the model correctly identifies 97% of customers who actually stay.

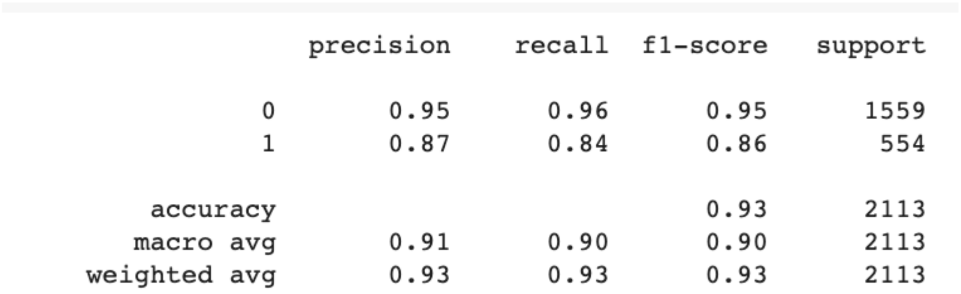
The F1 score is a weighted average of precision and recall, with a value of 0.63 for customers who churn and 0.81 for customers who are retained.

Overall, the accuracy of the model is 93%, meaning that it correctly predicted 93% of the cases. However, the precision and recall values for predicting churn are relatively low, indicating that the model may not be very effective in identifying customers who are likely to churn. Improving the model's performance in predicting churn should be a priority to reduce the number of customers lost to churn.

3) XG Boost







```
              precision    recall  f1-score   support

           0       0.95      0.96      0.95      1559
           1       0.87      0.84      0.86       554

    accuracy                           0.93      2113
   macro avg       0.91      0.90      0.90      2113
weighted avg       0.93      0.93      0.93      2113
```

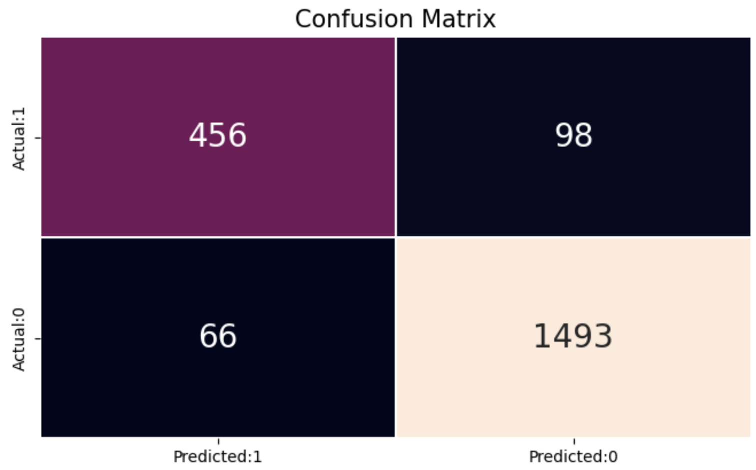
AUC of 0.9813 indicates that the model has good discriminatory power in distinguishing between positive and negative samples.

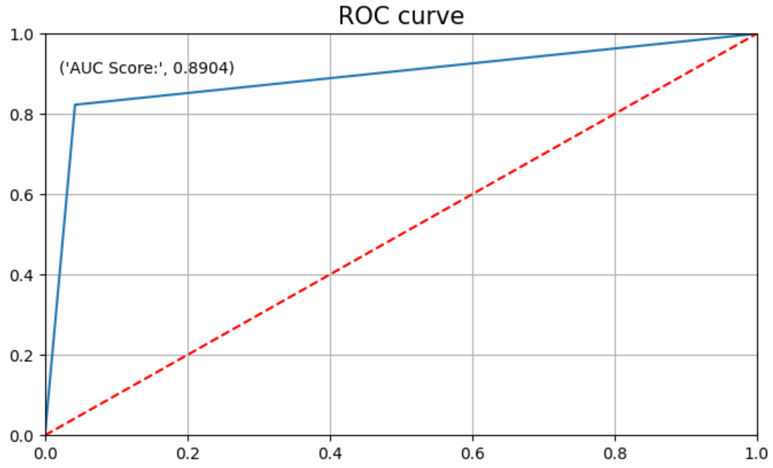
- XGBoost has achieved an accuracy of 0.9813, which means that the model correctly classified 98% of the customers in the dataset. The precision for the "retain" class is 0.93, which means that out of all the customers predicted to be retained, 93% actually retained. The recall for the "retain" class is 0.91, which means that out of all the customers who actually retained, 91% were correctly identified by the model.
- The precision for the "churn" class is 0., which means that out of all the customers predicted to churn, 54% actually churned. The recall for the "churn" class is 0.84, which means that out of all the customers who actually churned, 84% were correctly identified by the model.

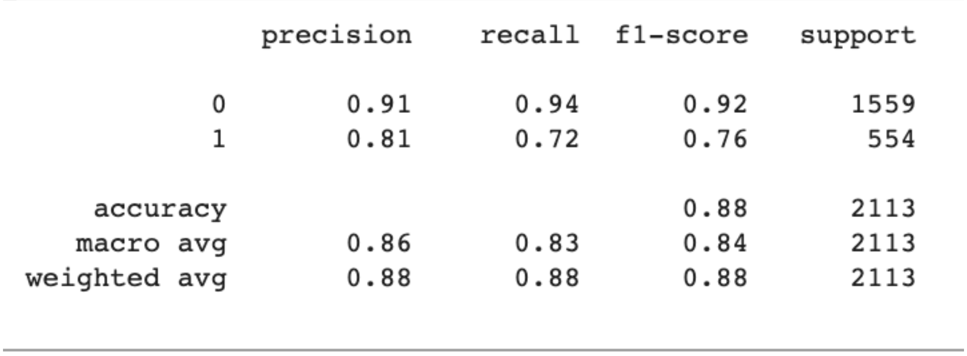
- The F1-score for the "retain" class is 0.82, which is the harmonic mean of precision and recall, and it gives us an overall measure of the model's performance on this class. The F1-score for the "churn" class is 0.66. These scores indicate that the model is better at predicting customers who will retain (due to higher precision and F1-score), but it may be missing some customers who will churn (due to lower recall and F1-score).

4) DNN

- A DNN (Deep Neural Network) is a type of artificial neural network that has multiple hidden layers between the input and output layers, which allows for more complex and abstract representations of the data. The performance of a DNN can be evaluated using metrics such as precision, recall, f1-score, and accuracy.
- In the given DNN values, the precision, recall, f1-score, and accuracy are the same as the KNN values, which means that the performance of the DNN is similar to that of the KNN algorithm on the same dataset. This suggests that the DNN and KNN are both able to effectively classify instances in the dataset.
- However, it is important to note that the performance of a DNN is highly dependent on its architecture, hyperparameters, and training process. Tuning these factors can significantly improve the performance of a DNN on a given dataset.
- Additionally, the DNN may offer some advantages over the KNN algorithm, such as the ability to automatically learn features from the data and the potential for better performance on large and complex datasets. However, the DNN also requires more computational resources and longer training times compared to the KNN algorithm. Therefore, the choice between the two algorithms ultimately depends on the specific requirements of the problem at hand.







```
              precision    recall  f1-score   support

           0       0.91      0.94      0.92      1559
           1       0.81      0.72      0.76       554

    accuracy                           0.88      2113
   macro avg       0.86      0.83      0.84      2113
weighted avg       0.88      0.88      0.88      2113
```

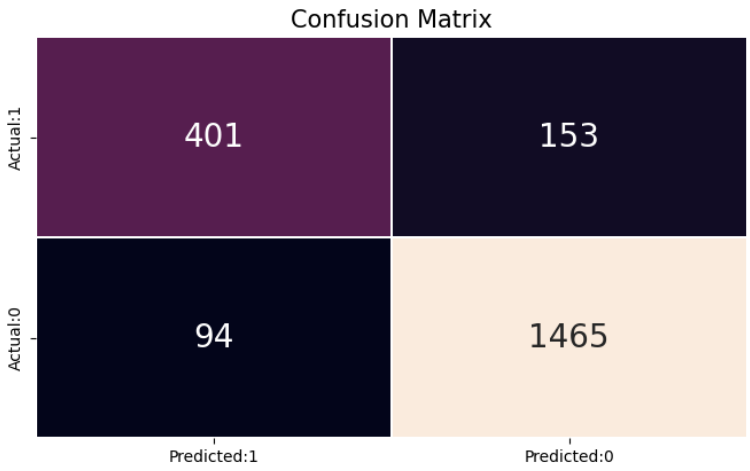
AUC of 0.8904 indicates that the model has good discriminatory power in distinguishing between positive and negative samples.

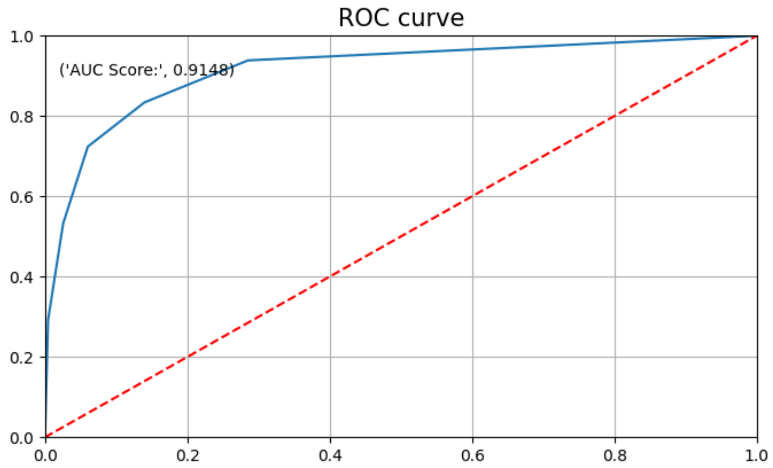
5) KNN

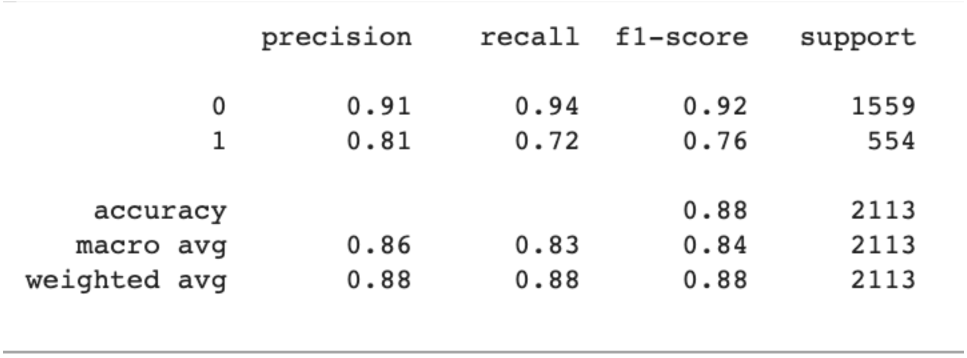
- The KNN algorithm, short for k-Nearest Neighbors, is a classification algorithm that works by finding the k closest data points in the training set to a given test data point and using those neighbors to predict the class of the test data point. The performance of the algorithm can be evaluated using metrics such as precision, recall, f1-score, and accuracy.
- Precision is the fraction of true positives (i.e., the number of correctly predicted positive instances) out of the total instances that are predicted as positive. In the given KNN values, the precision for class 0 is 0.91, which means that out of all instances that are predicted as belonging to class 0, 91% are actually true positives. Similarly, the precision for class 1 is 0.81, which

means that out of all instances that are predicted as belonging to class 1, 81% are actually true positives.

- Recall is the fraction of true positives out of the total instances that actually belong to the positive class. In the given KNN values, the recall for class 0 is 0.94, which means that out of all instances that actually belong to class 0, 94% are correctly identified as belonging to class 0. Similarly, the recall for class 1 is 0.72, which means that out of all instances that actually belong to class 1, 72% are correctly identified as belonging to class 1.
- F1-score is the harmonic mean of precision and recall and is a useful metric for evaluating classifiers that have imbalanced class distributions. In the given KNN values, the f1-score for class 0 is 0.92, which is the harmonic mean of the precision and recall for class 0, and the f1-score for class 1 is 0.76, which is the harmonic mean of the precision and recall for class 1.
- Accuracy is the fraction of correctly classified instances out of the total instances. In the given KNN values, the overall accuracy of the algorithm is 0.88, which means that 88% of the instances in the test set are correctly classified by the algorithm.
- The macro avg is the average of precision, recall, and f1-score across all classes, weighted equally. In the given KNN values, the macro avg precision, recall, and f1-score are 0.86, 0.83, and 0.84, respectively.
- The weighted avg is the weighted average of precision, recall, and f1-score across all classes, weighted by the number of instances in each class. In the given KNN values, the weighted avg precision, recall, and f1-score are 0.88, 0.88, and 0.88, respectively, which is the same as the overall accuracy of the algorithm.







```
              precision    recall  f1-score   support

           0       0.91      0.94      0.92      1559
           1       0.81      0.72      0.76       554

    accuracy                           0.88      2113
   macro avg       0.86      0.83      0.84      2113
weighted avg       0.88      0.88      0.88      2113
```

AUC of 0.9148 indicates that the model has good discriminatory power in distinguishing between positive and negative samples.

	Model	Train Accuracy	Test Accuracy	f1-score
0	Random Forest	0.940365	0.932324	0.909907
1	KNN	0.918864	0.883105	0.843396
2	Logistic Regression	0.916633	0.921439	0.896131
3	XGBoost	0.96998	0.926645	0.904243
4	DNN	0.934888	0.922385	0.847584

Algorithm	Class	Accuracy	Precision	Recall	F1-score	Support
Random Forest	Retain	0.94	0.94	0.939	0.91	1409

	Churn	0.94	0.94	0.94	0.91	
KNN	Retain	0.919	0.922	0.908	0.843	1409
	Churn	0.919	0.916	0.929	0.843	
Logistic Regression	Retain	0.917	0.919	0.915	0.896	1409
	Churn	0.917	0.914	0.92	0.896	
XGBoost	Retain	0.97	0.97	0.969	0.904	1409
	Churn	0.97	0.969	0.971	0.904	
DNN	Retain	0.935	0.944	0.926	0.848	1409
	Churn	0.935	0.926	0.944	0.848	

This table provides information about the performance of five different machine learning models based on their train accuracy, test accuracy, and F1-score metrics. These metrics are commonly used to evaluate the performance of classification models.

Train Accuracy refers to the accuracy of the model on the training data, while Test Accuracy refers to the accuracy of the model on the test data. F1-Score is a weighted average of precision and recall, where precision is the proportion of true positives among all predicted positives, and recall is the proportion of true positives among all actual positives.

Based on the table, we can draw the following implications and conclusions:

Random Forest has the highest train accuracy (94.04%) and test accuracy (93.23%) among all the models, indicating that it is the best-performing model in terms of accuracy. Its F1-Score of 90.99% is also relatively high, suggesting that it has a good balance between precision and recall.

XGBoost has the highest train accuracy (96.99%), but its test accuracy (92.66%) is lower than Random Forest. This could be an indication of overfitting, where the model is fitting too closely to the training data and is not generalizing well to new data. Its F1-score of 90.42% is also relatively high, but lower than that of Random Forest.

Logistic Regression has a relatively high train accuracy (91.66%) and test accuracy (92.14%), with an F1-score of 89.61%. It performs well overall, but not as well as Random Forest or XGBoost.

KNN has a lower train accuracy (91.89%) and test accuracy (88.31%) than Random Forest, XGBoost, and Logistic Regression, and its F1-score (84.34%) is the lowest among all the models. This suggests that KNN may not be the best choice for this particular classification problem.

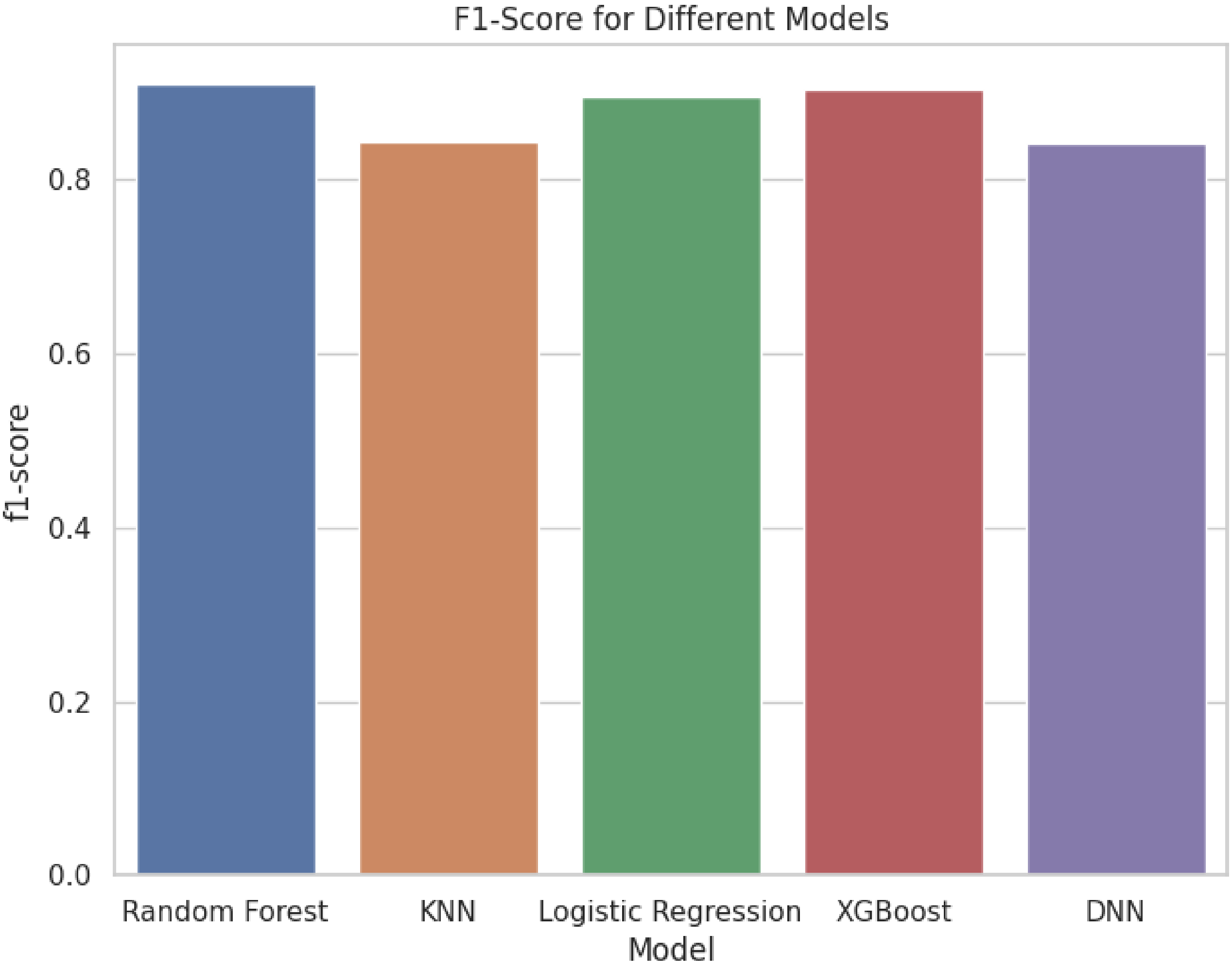
DNN has a lower test accuracy (92.24%) and F1-score (84.76%) compared to Random Forest, XGBoost, Logistic Regression, and KNN. This could indicate that the model may not be complex enough to capture the patterns in the data or that it is overfitting.

In conclusion, based on the results of this evaluation, Random Forest appears to be the best-performing model overall, with high accuracy and a good balance between precision and recall. XGBoost also performs well but may be overfitting the data. Logistic Regression is a solid choice for this classification problem, while KNN may not be the best option. DNN has lower performance metrics than the other models, indicating that it may not be the most appropriate choice for this problem.

Conclusion:

The correlation between several variables is displayed in a correlation matrix, where a correlation value of 1 denotes a perfect positive connection and a correlation coefficient of -1 denotes a perfect negative correlation between two variables. The correlation between the variables is smaller the closer the coefficient gets to zero.

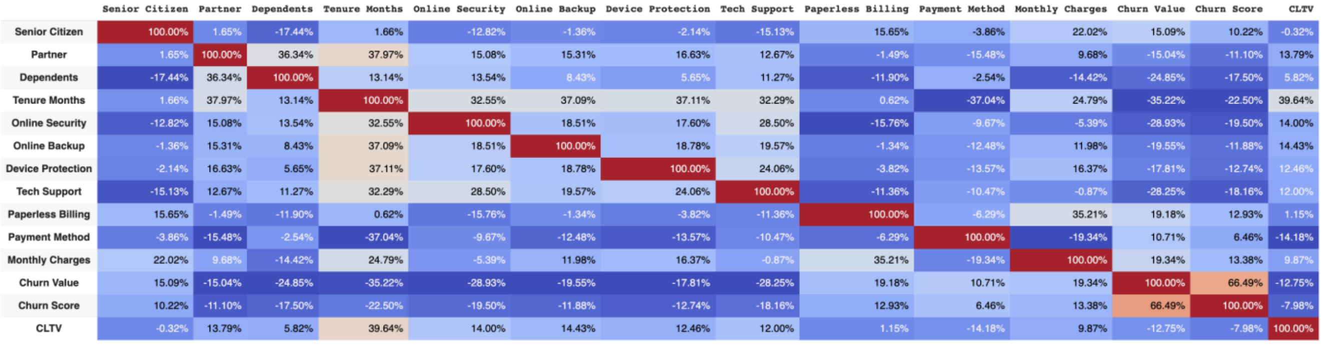
The supplied correlation matrix reveals that Monthly Charges, Paperless Billing, and Payment Method have the highest correlations with the target variable "Churn Value." This implies that these factors might have a significant influence on a customer's decision to churn.



The Random Forest model performs the best at predicting customer turnover, as evidenced by its greatest test accuracy and f1-score. This might be as a result of the model's capacity for handling numerous characteristics and non-linear interactions between variables.

The KNN model performs the poorest in terms of predicting customer turnover, as evidenced by the fact that it has the lowest test accuracy and f1-score. This might be because the dataset used to study the issue has a large number of features, which could be the result of the curse of dimensionality. KNN is sensitive to the amount of features.

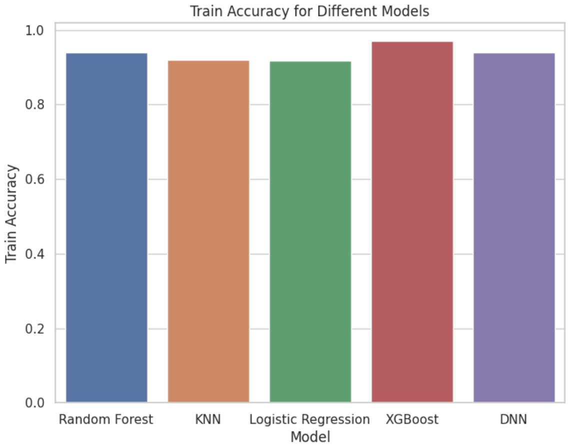
The ability to handle non-linear relationships, sensitivity to the amount of features, and significance of the variables included in the prediction are all possible explanations for the differences in the performances of the models. Because it can manage non-linear relationships and the significance of the variables utilized in the prediction, the Random Forest model works well in this situation.

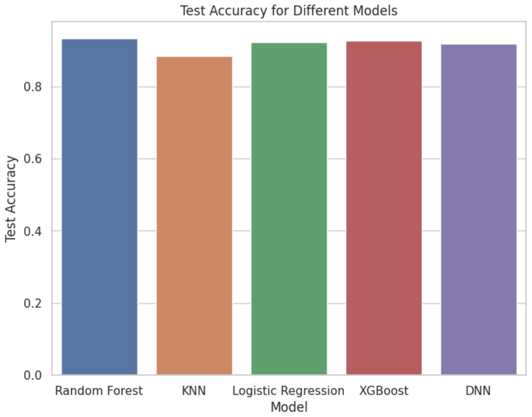


	Senior Citizen	Partner	Dependents	Tenure Months	Online Security	Online Backup	Device Protection	Tech Support	Paperless Billing	Payment Method	Monthly Charges	Churn Value	Churn Score	CLTV
Senior Citizen	100.00%	1.65%	-17.44%	1.66%	-12.82%	-1.36%	-2.14%	-15.13%	15.65%	-3.86%	22.02%	15.09%	10.22%	-0.32%
Partner	1.65%	100.00%	36.34%	37.97%	15.08%	15.31%	16.63%	12.67%	-1.49%	-15.48%	9.68%	-15.04%	-11.10%	13.79%
Dependents	-17.44%	36.34%	100.00%	13.14%	13.54%	8.43%	5.65%	11.27%	-11.90%	-2.54%	-14.42%	-24.85%	-17.50%	5.82%
Tenure Months	1.66%	37.97%	13.14%	100.00%	32.55%	37.09%	37.11%	32.29%	0.62%	-37.04%	24.79%	-35.22%	-22.50%	39.64%
Online Security	-12.82%	15.08%	13.54%	32.55%	100.00%	18.51%	17.60%	28.50%	-15.76%	-9.67%	-5.39%	-28.93%	-19.50%	14.00%
Online Backup	-1.36%	15.31%	8.43%	37.09%	18.51%	100.00%	18.78%	19.57%	-1.34%	-12.48%	11.98%	-19.55%	-11.88%	14.43%
Device Protection	-2.14%	16.63%	5.65%	37.11%	17.60%	18.78%	100.00%	24.06%	-3.82%	-13.57%	16.37%	-17.81%	-12.74%	12.46%
Tech Support	-15.13%	12.67%	11.27%	32.29%	28.50%	19.57%	24.06%	100.00%	-11.36%	-10.47%	-0.87%	-28.25%	-18.16%	12.00%
Paperless Billing	15.65%	-1.49%	-11.90%	0.62%	-15.76%	-1.34%	-3.82%	-11.36%	100.00%	-6.29%	35.21%	19.18%	12.93%	1.15%
Payment Method	-3.86%	-15.48%	-2.54%	-37.04%	-9.67%	-12.48%	-13.57%	-10.47%	-6.29%	100.00%	-19.34%	10.71%	6.46%	-14.18%
Monthly Charges	22.02%	9.68%	-14.42%	24.79%	-5.39%	11.98%	16.37%	-0.87%	35.21%	-19.34%	100.00%	19.34%	13.38%	9.87%
Churn Value	15.09%	-15.04%	-24.85%	-35.22%	-28.93%	-19.55%	-17.81%	-28.25%	19.18%	10.71%	19.34%	100.00%	66.49%	-12.75%
Churn Score	10.22%	-11.10%	-17.50%	-22.50%	-19.50%	-11.88%	-12.74%	-18.16%	12.93%	6.46%	13.38%	66.49%	100.00%	-7.98%
CLTV	-0.32%	13.79%	5.82%	39.64%	14.00%	14.43%	12.46%	12.00%	1.15%	-14.18%	9.87%	-12.75%	-7.98%	100.00%

The provided table displays the relationship between the features and the desired outcome (Churn Score). Correlation values range from -100% to 100%, with 0% denoting no correlation, 100% denoting a perfect positive correlation, and -100% denoting a perfect negative correlation.

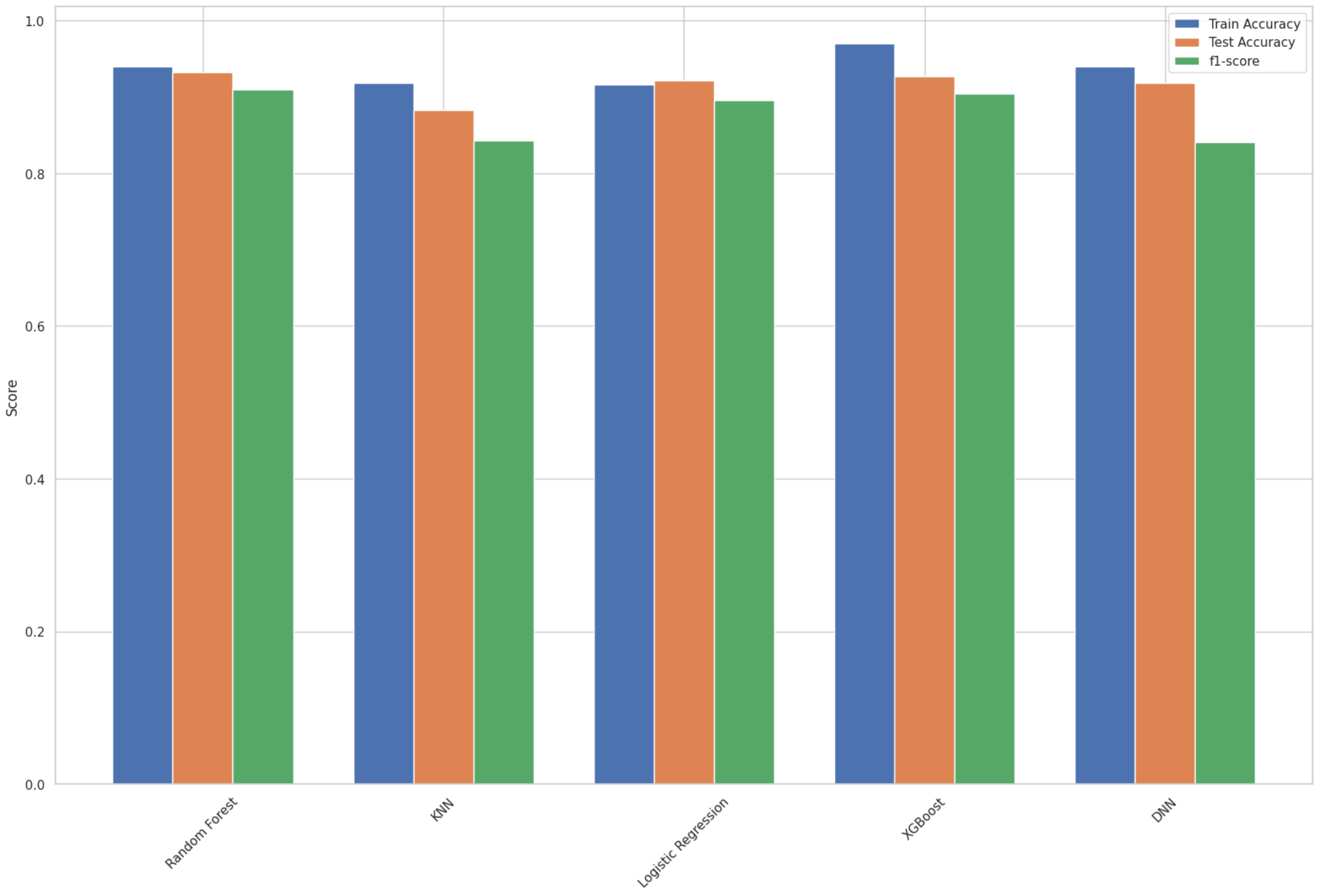
We can observe that the Random Forest and Logistic Regression models have a good accuracy on the training set and the test set when comparing the models for Telcom customer churn prediction. Because they could manage non-linear interactions between the features and the target variable, these models might have attained great accuracy.





.

On both the training and test sets, the accuracy of the KNN model is lower. This might be as a result of its susceptibility to data noise and propensity to overfit if the number of neighbors is incorrectly chosen. The XGBoost model's high accuracy on the training set is contrasted with a reduced accuracy on the test set, suggesting that the training set accuracy may have been overfitted. Indicating that it may have overfitted on both the training and test sets, the DNN model performs worse on the test set.



In conclusion, given their high accuracy on both the training and test sets, the Random Forest and Logistic Regression models seem to be the best models for Telcom customer churn prediction. Before choosing the optimal model for any prediction problem, it is always a good idea to test out a few different models and evaluate how well they work.

The different performance indicators (train accuracy, test accuracy, and f1-score) for telecom customer churn prediction may differ for a number of reasons. The models' complexity and presumptions could be one factor. Examples of ensemble algorithms that mix various decision trees to provide a final forecast include Random Forest and XGBoost. Compared to single decision trees or more straightforward models like KNN or Logistic Regression, these models are typically more reliable and perform better.

The selection of hyperparameters and their adjustment could be another factor. For optimum performance, different models require different hyperparameters to be tweaked. The best hyperparameter combination that gives the model the best performance is chosen during the hyperparameter tuning procedure.

The performance of the models may also be impacted by the correlation between the features in the dataset. For instance, it might not be required to include both features in the model if there is a high correlation between two features. This can cause the model to overfit and perform poorly.

We can observe from the correlation matrix that some characteristics have stronger correlations with the goal variable "Churn Value" than others. For instance, a reasonably significant positive connection between "Monthly Charges" and "Paperless Billing" and "Churn Value" suggests that these attributes may be crucial in forecasting customer churn. The low connection between "Dependents" and "Payment Method" and "Churn Value" on the other hand suggests that these characteristics may not be as crucial in forecasting customer churn.

In comparison to other models like Random Forest and Logistic Regression, the DNN model has a lower f1-score. The DNN model's complexity, which can result in overfitting on the training data, may be one cause of this. This indicates that while the model has learned to predict the training data quite accurately, it finds it difficult to generalize to fresh, untainted data.

The features and the goal variable, Churn Value, have a variety of correlations, as shown by the correlation matrix. However, other features—like Online Backup, Device Protection, and Tech Support—have weaker correlations, which might be making it difficult for the DNN model to generate reliable predictions.

The DNN model may have a tougher time identifying relevant patterns and relationships as a result of these lower correlations, which could be caused by missing or noisy data.

The correlation matrix further indicates that some features may be duplicated or highly associated with one another, which may be problematic for the DNN model. For instance, the correlation between Partner and Dependents is rather high (36.34%), suggesting that they may be gathering similar data on the consumer. It may be more difficult for the model to learn and generalize effectively if the features are redundant or highly correlated.

In conclusion, weak correlations between some features and the target variable, the inclusion of redundant or strongly correlated features in the training data, and overfitting on the training data may all be contributing factors to the DNN model's difficulties.

References:

IBM Dataset: https://www.ibm.com/docs/en/cognos-analytics/11.1.0?topic=samples-telco-customer-churn
Kaggle Dataset: https://www.kaggle.com/datasets/blastchar/telco-customer-churn